

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG



ĐỒ ÁN TỐT NGHIỆP

NGÀNH : CÔNG NGHỆ THÔNG TIN

Sinh viên : Bùi Đình Tuấn Đạt

Giảng viên hướng dẫn: TS. Lê Văn Phùng

HẢI PHÒNG – 2021

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG**

**MÔ HÌNH THIẾT KẾ CSDL QUAN HỆ MỨC LOGIC
DỰA TRÊN PHƯƠNG PHÁP “BLANPRE”
VÀ ỨNG DỤNG**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY
NGÀNH: CÔNG NGHỆ THÔNG TIN**

Sinh viên : Bùi Đình Tuấn Đạt

Giảng viên hướng dẫn: TS. Lê Văn Phùng

HẢI PHÒNG – 2021

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG

NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP

Sinh viên: Bùi Đình Tuấn Đạt

Mã SV: 1612111010

Lớp : CT2001C

Ngành : Công nghệ thông tin

Tên đề tài: Mô hình thiết kế CSDL quan hệ mức logic dựa trên phương pháp
“Blanpre” và ứng dụng

NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

- Tìm hiểu về Mô hình thiết kế CSDL quan hệ mức logic
- Tìm hiểu về phương pháp “Blanpre”
- Xây dựng CSDL áp dụng phương pháp “Blanpre” và ứng dụng quản lý các cung đường bộ trên địa bàn TP. Hải Phòng

2. Các tài liệu, số liệu cần thiết

- Tài liệu tham khảo về CSDL
- Tài liệu tham khảo về phương pháp phân tích và thiết kế hệ thống thông tin
- Tài liệu tham khảo về quản lý thông tin các cung đường

3. Địa điểm thực tập tốt nghiệp

- Công ty Cổ Phần Thiết Bị Điện, Điện Tử - Bách Khoa

CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Họ và tên : Lê Văn Phùng

Học hàm, học vị : Tiến sĩ

Cơ quan công tác : Viện Công nghệ Thông tin,
Viện Hàn lâm Khoa học và Công nghệ Việt Nam.

Nội dung hướng dẫn:

- Tìm hiểu về Mô hình thiết kế cơ sở dữ liệu quan hệ mức logic
- Tìm hiểu về phương pháp “Blanpre” để thiết kế cơ sở dữ liệu quan hệ cho một ứng dụng.
- Xây dựng cơ sở dữ liệu theo phương pháp “Blanpre” và viết chương trình thử nghiệm ứng dụng quản lý các cung đường bộ trên địa bàn TP. Hải Phòng.

Đề tài tốt nghiệp được giao ngày 18 tháng 10 năm 2021

Yêu cầu phải hoàn thành xong trước ngày 30 tháng 12 năm 2021

Đã nhận nhiệm vụ ĐTTN

Sinh viên

Đã giao nhiệm vụ ĐTTN

Giảng viên hướng dẫn

TS.Lê Văn Phùng

Hải Phòng, ngày tháng năm 2021

TRƯỞNG KHOA

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIÁNG VIÊN HƯỚNG DẪN TỐT NGHIỆP

Họ và tên giảng viên: Lê Văn Phùng

Đơn vị công tác: Viện Công nghệ Thông tin,
Viện Hàn lâm Khoa học và Công nghệ Việt Nam.

Họ và tên sinh viên : **Bùi Đình Tuấn Đạt**

Ngành: Công nghệ Thông tin

Nội dung hướng dẫn:

- Tìm hiểu về Mô hình thiết kế cơ sở dữ liệu quan hệ mức logic
- Tìm hiểu về phương pháp “Blanpre” để thiết kế cơ sở dữ liệu quan hệ cho một ứng dụng.
- Xây dựng cơ sở dữ liệu theo phương pháp “Blanpre” và viết chương trình thử nghiệm ứng dụng quản lý các cung đường bộ trên địa bàn TP. Hải Phòng.

1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp

- Học sinh có tinh thần cố gắng cao trong quá trình làm đề án tốt nghiệp, từ sưu tập tài liệu, tìm hiểu tài liệu, tổng hợp tư liệu, phân tích số liệu thực tế tại nơi ứng dụng.
- Đảm bảo đúng tiến độ thực hiện đề án theo quy định của nhà trường và hướng dẫn của giáo viên hướng dẫn.

2. Đánh giá chất lượng của đề án/khóa luận (so với nội dung yêu cầu đó đề ra trong nhiệm vụ Đ.T. T.N trên các mặt lý luận, thực tiễn, tính toán số liệu...)

- Đề án tốt nghiệp của sinh viên đã đáp ứng đầy đủ những vấn đề cốt yếu nhất của nội dung đề tài theo yêu cầu đề cương đề án tốt nghiệp đã đặt ra.
- Phân lý thuyết đã cơ bản đáp ứng được yêu cầu tổng quan kiến thức chung và tìm hiểu sâu về kiến thức hẹp để áp dụng thực tế.
- Phần thực hành thử nghiệm lập trình tuy còn đơn giản nhưng đã thể hiện được khả năng vận dụng những kiến thức học được vào giải quyết bài toán thực tế.

3. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Đạt Không đạt Điểm:.....

Hải Phòng, ngày 22 tháng 12 năm 2021

Giảng viên hướng dẫn

TS. Lê Văn Phụng

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN CHĂM PHẢN BIỆN

Họ và tên giảng viên:

.....

Đơn vị công tác:

.....

Họ và tên sinh viên: Ngành:

.....

Đề tài tốt

nghệ:.....

.....

.....

1. Phần nhận xét của giảng viên chăm phản biện

.....

....

.....

....

.....

....

.....

....

.....

....

.....

....

2. Những mặt còn hạn chế

.....

....

.....
....
.....
....
.....
....
.....
....
.....
....
.....
....

3. Ý kiến của giảng viên chấm phản biện

Đạt Không đạt Điểm:.....

Hải Phòng, ngày ... tháng ... năm 2021
Giảng viên chấm phản biện

LỜI CẢM ƠN

Em xin cảm ơn chân thành đến toàn thể thầy cô trong trường Đại Học Quản lý và Công nghệ nói chung và các thầy cô trong khoa Công nghệ thông tin nói riêng, những người đã tận tình hướng dẫn, dạy dỗ và trang bị cho em những kiến thức bổ ích trong năm vừa qua.

Em xin chân thành gửi lời cảm ơn sâu sắc đến các thầy cô khoa công nghệ thông tin đặc biệt là thầy giáo Ts Lê Văn Phùng, người đã tận tình hướng dẫn, trực tiếp chỉ bảo và tạo mọi điều kiện giúp đỡ em trong suốt quá trình làm đề án tốt nghiệp.

Sau cùng em xin gửi lời cảm ơn chân thành tới những người bạn đã đồng viên, cổ vũ và đóng góp ý kiến trong quá trình học tập, nghiên cứu cũng như quá trình làm đề án tốt nghiệp.

Bên cạnh đó, do còn nhiều hạn chế về kiến thức và kinh nghiệm nên trong đề án không tránh khỏi những thiếu sót, kính mong quý thầy cô, anh chị, bạn bè chỉ bảo thêm.

Em xin chân thành cảm ơn!

Hải Phòng, ngày tháng năm 2021

Sinh viên

Bùi Đình Tuấn Đạt

MỤC LỤC

LỜI CẢM ƠN	1
CHƯƠNG 1	6
TỔNG QUAN VỀ LÝ THUYẾT CƠ SỞ DỮ LIỆU QUAN HỆ	6
1.1. Cơ sở dữ liệu	6
1.1.1. Các khái niệm chung	6
1.1.2. Mô hình dữ liệu và mô hình dữ liệu quan hệ	6
1.2. Phụ thuộc hàm và thiết kế logic cơ sở dữ liệu quan hệ	8
1.2.1. Khái niệm về phụ thuộc hàm.....	8
1.2.2. Các thuật toán xác định bao đóng và khóa trong sơ đồ quan hệ $s=\langle R,F \rangle$	9
1.2.3. Các dạng chuẩn và các thuật toán liên quan.....	12
1.2.4. Chiến lược thiết kế logic cơ sở dữ liệu quan hệ.....	14
CHƯƠNG 2	20
MÔ HÌNH THIẾT KẾ CSDL QUAN HỆ MỨC LOGIC DỰA TRÊN PHƯƠNG PHÁP BLANPRE	20
2.1. Ý nghĩa của thiết kế CSDL mức logic	20
2.2. Khuôn cảnh chung các bước thiết kế CSDL mức logic	20
2.3. Xây dựng mô hình khái niệm dữ liệu bằng phương pháp Blaupre	22
2.3.1. Ý tưởng của mô hình	22
2.3.2. Quy trình thiết kế.....	22
2.3.3. Ví dụ	23
2.4. Chuyển mô hình khái niệm dữ liệu sang mô hình dữ liệu mức logic	31
2.4.1. Kỹ thuật chuyển mô hình khái niệm dữ liệu về hệ lược đồ quan hệ.....	31
2.4.2. Kỹ thuật chuẩn hóa.....	31
2.4.3. Kỹ thuật chuyển từ hệ lược đồ quan hệ sang sơ đồ E_R (ERD - mô hình dữ liệu mức logic).....	33
CHƯƠNG 3	35
ỨNG DỤNG THIẾT KẾ CSDL VỀ THÔNG TIN CÁC CUNG ĐƯỜNG BỘ TRÊN ĐỊA BÀN TP. HẢI PHÒNG	35
3.1. Bài toán quản lý thông tin các cung đường bộ trên địa bàn TP. Hải Phòng	35
3.2. Những phần mềm hỗ trợ	35

3.3. Thuật toán sử dụng và xác định dữ liệu đầu vào	36
3.3.1. Thuật toán sử dụng	36
3.3.2. Dữ liệu đầu vào.....	36
3.4. Nội dung và kết quả thử nghiệm.....	38
3.4.1. Nội dung thiết kế cơ sở dữ liệu các cung đường TP. Hải Phòng	38
3.4.2. Giới thiệu chương trình	55
<i>KẾT LUẬN</i>.....	62
1. <i>Kết quả đạt được của đồ án</i>	62
2. <i>Những hạn chế</i>	62
3. <i>Hướng phát triển</i>	62
<i>TÀI LIỆU THAM KHẢO</i>	63

DANH SÁCH HÌNH VẼ

<i>Hình 1.1. Phân lớp các dạng chuẩn</i>	14
<i>Hình 2.1. Sơ đồ khuôn cảnh chung các bước thiết kế CSDL mức logic [3]</i>	22
<i>Hình 2.2. Mô hình khái niệm dữ liệu</i>	30
<i>Hình 3.1. Mô hình khái niệm dữ liệu bài toán quản lý cung đường giao thông</i>	47
<i>Hình 3.2. Mô hình Thực thể- Mối quan hệ (Mô hình E_R) bài toán quản lý cung đường giao thông:</i>	50
<i>Hình 3.3. Giao diện form đăng nhập</i>	57
<i>Hình 3.4. Giao diện form quản lý thông tin cung đường</i>	58
<i>Hình 3.5. Giao diện form quản lý thông tin bảo trì</i>	59
<i>Hình 3.6. Giao diện form quản lý thông tin người dùng</i>	60
<i>Hình 3.7. Giao diện form tìm kiếm thông tin cung đường</i>	60

DANH SÁCH BẢNG

<i>Bảng 1.1. Quan hệ THISINH</i>	8
<i>Bảng 1.2. Quan hệ trình độ ngoại ngữ</i>	12
<i>Bảng 2.1. Ma trận Blanpre</i>	24
<i>Bảng 2.2. Ma trận Blanpre rút gọn</i>	25
<i>Bảng 2.3. Ma trận phụ thuộc hàm Blanpre</i>	27
<i>Bảng 3.1. Cấu trúc bảng QUAN</i>	52
<i>Bảng 3.2. Cấu trúc bảng DUONG</i>	52
<i>Bảng 3.3. Cấu trúc bảng LOAIMADUONG</i>	53
<i>Bảng 3.4. Cấu trúc bảng KIEUDUONG</i>	53
<i>Bảng 3.5. Cấu trúc bảng TOCHUCGIAOTHONG</i>	53
<i>Bảng 3.6. Cấu trúc bảng MUCDOHUHONG</i>	54
<i>Bảng 3.7. Cấu trúc bảng LOAIBAOTRI</i>	54
<i>Bảng 3.8. Cấu trúc bảng DONVITHICONG</i>	54
<i>Bảng 3.9. Cấu trúc bảng THONGTINBAOTRI</i>	55

CHƯƠNG 1

TỔNG QUAN VỀ LÝ THUYẾT CƠ SỞ DỮ LIỆU QUAN HỆ

1.1. Cơ sở dữ liệu

1.1.1. Các khái niệm chung

Dữ liệu bao gồm số, kí tự, văn bản, hình ảnh, đồ họa, âm thanh, đoạn phim,... có một giá trị nào đó đối với người sử dụng chúng và được lưu trữ, xử lý trong máy tính [5].

Cơ sở dữ liệu được xác định như một bộ sưu tập các dữ liệu có liên quan logic với nhau; nó được tổ chức sắp xếp theo một cách nào đó và được các hệ ứng dụng của một đơn vị/cơ quan cụ thể nào đó sử dụng.

1.1.2. Mô hình dữ liệu và mô hình dữ liệu quan hệ

1.1.2.1. Mô hình dữ liệu

Mô hình dữ liệu là cách biểu diễn các cấu trúc dữ liệu cho một cơ sở dữ liệu dưới dạng các khái niệm. Các cấu trúc dữ liệu bao gồm các đối tượng dữ liệu, mối liên hệ giữa các dữ liệu, ngữ nghĩa của dữ liệu và các ràng buộc trên đối tượng dữ liệu đó [5].

Có 3 loại mô hình cơ sở dữ liệu:

1. Mô hình cơ sở dữ liệu mức quan niệm

- Là mô hình mô tả dữ liệu của thế giới thực gắn với hoạt động nghiệp vụ của tổ chức sử dụng nó.
- Mô tả các cấu trúc và mối liên hệ giữa các đơn vị thông tin cơ bản.
- Là phương tiện để giao tiếp với người sử dụng nhằm xác định đúng đắn và đầy đủ các yêu cầu thông tin của hệ thống.
- Hoàn toàn độc lập với mọi hệ quản trị dữ liệu và cách thức sử dụng nó.
- Cung cấp các khái niệm gắn liền với cách cảm nhận dữ liệu của người sử dụng. Nó tập trung vào bản chất logic của biểu diễn dữ liệu, quan tâm đến cái được biểu diễn, chứ không quan tâm đến cách biểu diễn.

- Mô hình khái niệm cơ bản như mô hình E_R. Mô hình E_R dùng để mô tả cấu trúc logic tổng thể (lược đồ) của một cơ sở dữ liệu bằng hình ảnh (đặc tả). Người ta quan niệm thế giới thực bao gồm tập các E và R. Trong đó, E là “sự vật”/ “đối tượng” tức là thực thể trong thế giới thực và phải phân biệt được, còn R là mối quan hệ (relationship) giữa một nhóm thực thể [6].

2. Mô hình cơ sở dữ liệu mức logic:

- Cung cấp khái niệm cho người sử dụng có thể được và không xa so với cách tổ chức dữ liệu trong máy tính. Chúng che dấu một số chi tiết về việc lưu trữ dữ liệu nhưng có thể cài đặt trực tiếp trên hệ thống máy tính. Mô hình dữ liệu logic cho một hệ quản trị cơ sở dữ liệu:

- Mô tả các dữ liệu bằng cách sử dụng các ký hiệu tương ứng với mô hình dữ liệu mà 1 hệ quản trị cơ sở dữ liệu xây dựng trên nó.

- Có 4 loại mô hình dữ liệu logic: mô hình dữ liệu phân cấp, mạng, quan hệ, hướng đối tượng.

- Hiện nay, được tổ chức theo mô hình dữ liệu quan hệ là chủ yếu.

3. Mô hình cơ sở dữ liệu mức vật lý:

- Cung cấp các khái niệm mô tả chi tiết về việc các dữ liệu được lưu trữ trong máy như thế nào.

1.1.2.2. Mô hình dữ liệu quan hệ

Mô hình dữ liệu quan hệ được Cold đề xuất năm 1970. Nó đã tạo ra một cuộc cách mạng mới trong lĩnh vực cơ sở dữ liệu và nhanh chóng thay thế các mô hình dữ liệu trước đó [5].

Mô hình dữ liệu quan hệ tương đối đơn giản và dễ hiểu. Mô hình dữ liệu quan hệ là mô hình dữ liệu mà cốt lõi của nó là cơ sở dữ liệu quan hệ. Một cơ sở dữ liệu quan hệ là một tập của một hoặc nhiều quan hệ, trong đó mỗi một quan hệ là một bảng. Mô hình quan hệ sử dụng một tập các bảng để biểu diễn cả dữ liệu và các mối liên hệ giữa những dữ liệu này. Bảng có n cột và mỗi cột có một tên duy nhất.

Những cơ sở dữ liệu quan hệ thông dụng nhất đều có thể sử dụng ngôn ngữ SQL (Structured Query Language)

1.2. Phụ thuộc hàm và thiết kế logic cơ sở dữ liệu quan hệ

1.2.1. Khái niệm về phụ thuộc hàm

Khái niệm về phụ thuộc hàm trong một quan hệ là rất quan trọng trong việc thiết kế mô hình dữ liệu. Năm 1970, E.F Cold đã mô tả phụ thuộc hàm trong mô hình dữ liệu quan hệ, nhằm giải quyết việc phân rã không mất thông tin [1,5].

Cho $R = \{a_1, a_2, \dots, a_n\}$ là tập thuộc tính, $r = \{h_1, h_2, \dots, h_m\}$ là một quan hệ trên R , và $A, B \subseteq R$ (A, B là tập cột hay tập thuộc tính). Khi đó ta nói A xác định hàm cho B hay B phụ thuộc hàm vào A trong r (ký pháp $A \xrightarrow[r]{f} B$) nếu: $(\forall h_i, h_j \in r) ((\forall a \in A) (h_i(a) = h_j(a)) \Rightarrow (\forall b \in B) (h_i(b) = h_j(b)))$ nghĩa là đối số trùng nhau thì hàm có cùng giá trị. Đặt $F_r = \{(A, B) : A, B \subseteq R, A \xrightarrow[r]{f} B\}$. Lúc đó F_r được gọi là họ đầy đủ các phụ thuộc hàm của r .

Nhận xét :

Ta có thể thấy rằng B mà phụ thuộc hàm vào A , nếu hai dòng bất kì mà các giá trị của tập thuộc tính A mà bằng nhau từng cặp một, thì kéo theo các giá trị trên tập thuộc tính B cũng phải bằng nhau từng cặp một.

Ví dụ: Xét quan hệ [4]:

Bảng 1.1. Quan hệ THISINH

SBD	Hoten	Diachi	Tinh	Khuvuc
PD711001	Nguyễn Thái Bình	12 Bản Nhàn	Lạng Sơn	0
PD711002	Trần Nam Ninh	3 Kim mã	Hà Nội	3
PD711003	Lê Thanh Hoa	53 Hai Bà Trưng	Hà Nội	3
PD711004	Vũ Thúy Hồng	89 Đồng Đăng	Lạng Sơn	0
PD711005	Phạm Như Thúy	40 Trần Hưng Đạo	Hải Dương	2

Trong quan hệ THISINH, dựa vào định nghĩa phụ thuộc hàm của quan hệ ta có: {tinh} -> {khuvuc}; {sbd} -> {hoten, diachi, tinh, khuvuc}

Ý nghĩa: Khái niệm phụ thuộc hàm miêu tả một loại ràng buộc (phụ thuộc dữ liệu) xảy ra tự nhiên nhất giữa các tập thuộc tính.

1.2.2. Các thuật toán xác định bao đóng và khóa trong sơ đồ quan hệ $s = \langle R, F \rangle$

1. Một số thuật toán liên quan đến bao đóng

Một vấn đề thường xuyên xảy ra đối với một sơ đồ quan hệ cho trước ($s = \langle R, F \rangle$), và một phụ thuộc hàm $A \rightarrow B$, chúng ta muốn biết $A \rightarrow B$ có là phân tử của F^+ hay không. Để trả lời câu hỏi này chúng ta cần tính bao đóng F^+ của tập các phụ thuộc hàm F .

Tuy nhiên tính F^+ trong trường hợp tổng quát là rất khác nhau và tốn kém thời gian vì các tập phụ thuộc hàm F^+ là rất lớn cho dù F có thể là nhỏ. Chẳng hạn, $F = \{A \rightarrow B_1, A \rightarrow B_2, \dots, A \rightarrow B_n\}$, khi đó F^+ bao gồm cả những phụ thuộc hàm $A \rightarrow Y$ với $Y \subseteq \{B_1 \cup B_2 \cup \dots \cup B_n\}$, như vậy ta sẽ có 2^n tập con Y . Trong khi đó việc tính bao đóng của tập thuộc tính A lại không khó. Theo kết quả đã trình bày ở trên thì việc kiểm tra $A \rightarrow B \in F^+$ sẽ được thể bởi tính A^+ [1,5,6].

Thuật toán Tính bao đóng của một tập các thuộc tính đối với tập các phụ thuộc hàm trên sơ đồ quan hệ

Vào: $s = \langle R, F \rangle$ là một sơ đồ quan hệ

Trong đó:

$R = (a_1, a_2, \dots, a_n)$ là tập hữu hạn các thuộc tính.

F là tập các phụ thuộc hàm và $A \subseteq R$

Ra: A^+ là bao đóng của A đối với F .

Nhớ rằng $A^+ = \{a: A \rightarrow \{a\} \in F^+\}$.

$A \rightarrow B \in F^+$ nếu và chỉ nếu $B \subseteq A^+$.

Phương pháp :

Lần lượt tính các tập thuộc tính A_0, A_1 như sau:

1. $A_0 = A$

2. $A_i = A_{i-1} \cup \{a\}$ nếu $\exists (C \rightarrow D) \in F, \{a\} \in D$ và $C \subseteq A_{i-1}$

3. Rõ ràng $A = A_0 \subseteq A_1 \subseteq \dots \subseteq A_i$ và R hữu hạn nên tồn tại i sao cho:

$A_i = A_{i+1}$. Khi ấy thuật toán dừng và A_i chính là A^+

Ví dụ:

Xét sơ đồ quan hệ $s = \langle R, F \rangle$

Trong đó :

$$F = \begin{cases} \{c\} \rightarrow \{t\} \\ \{h,r\} \rightarrow \{c\} \\ \{h,t\} \rightarrow \{r\} \\ \{c,s\} \rightarrow \{g\} \\ \{h,s\} \rightarrow \{r\} \end{cases} \quad R = \{c, t, h, r, s, g\}$$

Tính $\{h, r\}^+$?

$$A_0 = \{h, r\}$$

$$A_1 = \{h, r, c\} \text{ do } \{h, r\} \rightarrow \{c\} \in F$$

$$A_2 = \{h, r, c, t\} \text{ do } \{c\} \rightarrow \{t\} \in F$$

$$A_3 = \{h, r, c, t\} = A_2$$

$$\text{Vậy } \{h, r\}^+ = \{h, r, c, t\}$$

2. Một số thuật toán liên quan đến khóa

Khi giải quyết các bài toán thông tin quản lý, người ta thường sử dụng các hệ quản trị cơ sở dữ liệu mà trong đó chứa cơ sở dữ liệu quan hệ. Các phép xử lý đối với bài toán này thường là tìm kiếm bản ghi sau đó thêm bản ghi mới, thay đổi nội dung bản ghi hoặc xóa bản ghi. Trong các thao tác trên, việc tìm kiếm bản ghi là rất quan trọng. Muốn tìm được bản ghi trong file dữ liệu thì chúng ta phải xây dựng khóa của file dữ liệu đó. Việc tìm khóa ở đây chính là tìm khóa tối thiểu.

Thuật toán tìm khóa tối thiểu cho một sơ đồ quan hệ [3]:

Input: Sơ đồ quan hệ $s = \langle R, F \rangle$

Trong đó : F là tập các phụ thuộc hàm

$$R = \{a_1, \dots, a_n\} \text{ là tập các thuộc tính}$$

Output: K là tối thiểu của s

Phương pháp:

Tìm liên tiếp các tập thuộc tính K_0, K_1, \dots, K_n như sau:

$$K_0 = R = \{a_1, \dots, a_n\}$$

$$K_i = \begin{cases} K_{i-1} & K_{i-1} \text{ nếu } K_{i-1} - \{a_i\} \rightarrow R \notin F^+ \\ K_{i-1} - \{a_i\} & K_{i-1} - \{a_i\} \text{ nếu ngược lại} \end{cases}$$

$K = K_n$ là khóa tối thiểu

Ta có thể dùng công thức tương đương:

$$K_i = \begin{cases} K_{i-1} - \{a_i\} & \text{Nếu } \{K_{i-1} - a_i\}^+ = R \\ K_{i-1} & \text{Nếu ngược lại.} \end{cases}$$

Nhận xét:

- Thay đổi thứ tự các thuộc tính của R bằng thuật toán trên chúng ta có thể tìm được một khóa tối thiểu khác.

- Nếu như đã biết A là một khóa nào đó thì có thể đặt $K_0 = A$, ta vẫn tìm ra được khóa tối thiểu và thời gian tìm nhanh hơn.

Ví dụ : Giả sử $s = \langle F, R \rangle$ là một lược đồ quan hệ trong đó:

$$R = \{a, b, c, d\}$$

$$F = \{\{a, b\} \rightarrow \{d\}, \{c\} \rightarrow \{b\}\}$$

Tìm khóa tối thiểu của sơ đồ quan hệ.

Áp dụng thuật toán trên ta có:

$$+ K_0 = R = \{a, b, c, d\}$$

+ Tính K_1

$$\text{Xét } K_1 = K_0 - \{a\} = \{b, c, d\}$$

$$\{b, c, d\}^+ = \{b, c, d\} \neq R$$

$$\text{Vậy } K_1 = \{a, b, c, d\}. (K_1 = K_0)$$

+ Tính K_2

$$\text{Xét } K_2 = K_1 - \{b\} = \{a, c, d\}$$

$$\{a, c, d\}^+ = \{a, b, c, d\} = R$$

$$\text{Vậy } K_2 = \{a, c, d\}$$

+ Tính K_3

$$\text{Xét } K_3 = K_2 - \{c\} = \{a, d\}$$

$$\{a, d\}^+ = \{a, d\} \neq R$$

Vậy $K_3 = \{a, c, d\}$ ($K_3 = K_2$)

+ Tính K_4

Xét $K_4 = K_3 - \{d\} = \{a, c\}$

$$\{a, c\}^+ = \{a, b, c, d\} = R$$

Vậy $K_4 = \{a, c\}$

Vậy khóa tối thiểu là $\{a, c\}$

1.2.3. Các dạng chuẩn và các thuật toán liên quan

*Các định nghĩa chuẩn hóa [3]

Định nghĩa 1 (Dạng chuẩn 1 – 1NF)

Giả sử $r = \{h_1, h_2, \dots, h_m\}$ là file dữ liệu trên tập cột $R = \{a_1, a_2, \dots, a_n\}$

Khi đó r là 1NF nếu các giá trị $h_i(a_j)$ là sơ cấp với mọi i, j

Khái niệm sơ cấp hiểu ở đây là giá trị $h_i(a_j)$ ($i = 1, \dots, m; j = 1, \dots, n$) không phân chia được nữa

Ví dụ: Xét quan hệ - Trình độ ngoại ngữ

Bảng 1.2. Quan hệ trình độ ngoại ngữ

MNS	Hoten	Ngoaingu
A199001	Hoàng Diệu	Anh, Nhật
A199002	Quốc Trung	Anh, Trung
A199003	Chí Hiếu	Anh, Đức
A199004	Lan Anh	Anh
A199005	Phương Lan	Anh
A199006	Bảo Quỳnh	Pháp, Nga, Anh

Có thể thấy thuộc tính NGOAINGU còn có thể được phân chia nhỏ hơn ra thành từng ngoại ngữ một và sau đó còn có thể phân thành hai bộ phận là tên ngoại ngữ và trình độ ngoại ngữ. Do vậy quan hệ ngoại ngữ chưa ở dạng chuẩn 1.

Định nghĩa 2 (Dạng chuẩn 2 – 2NF)

Quan hệ r được gọi là dạng chuẩn 2 nếu:

- Quan hệ r là dạng chuẩn 1

- Với mọi khóa tối thiểu K :

$A \rightarrow \{a\} \notin F$, với $A \subset K$ và a là thuộc tính thứ cấp (thuộc tính không phải là một phần tử của 1 khóa tối thiểu nào của r).

Định nghĩa 3 (dạng chuẩn 3 – 3NF)

Quan hệ r là dạng chuẩn 3 nếu:

$A \rightarrow \{a\} \notin F$, đối với A mà $A^+ \neq R$, $a \notin A$, $a \notin \cup K$

Có nghĩa là:

- K là một khóa tối thiểu
- A là thuộc tính thứ cấp
- A không là khóa
- $A \rightarrow \{a\}$ không đúng trong r

Định nghĩa 4 (Dạng chuẩn Boyce – codd – BCNF)

Quan hệ $r = \{h_1, h_2, \dots, h_m\}$ được gọi là dạng chuẩn Boyce – codd nếu $A \rightarrow \{a\} \notin F_r$, đối với những tập thuộc tính A mà $A^+ \neq R$, $a \notin A$.

Nhận xét:

Qua định nghĩa có thể xây dựng dạng chuẩn BCNF là 3NF và 3NF là 2NF. Chúng ta có thể đưa ra các ví dụ chứng tỏ quan hệ là 2NF nhưng không là 3NF và có quan hệ là 3NF nhưng không là BCNF.

Nói cách khác là lớp các quan hệ BCNF là lớp con thực sự của lớp các quan hệ 3NF và lớp các quan hệ 3NF và lớp các quan hệ 3NF này lại là lớp con thực sự của lớp các quan hệ 2NF

Đối với $s = \langle R, F \rangle$ thì các dạng 2NF, 3NF, BCNF trong đó ta thay $F_+ = F^+$

Chú ý: Đối với sơ đồ quan hệ chúng ta không có dạng chuẩn 1NF.

Ví dụ 1:

Cho $s = \langle R, F \rangle$ là sơ đồ quan hệ, với $R = (a_1, a_2, a_3, a_4)$

$F = \{ \{a_1\} \rightarrow \{a_2\}, \{a_3\} \rightarrow \{a_4\}, \{a_2\} \rightarrow \{a_1, a_2, a_4\} \}$

Để thấy a_1, a_2 là các khóa tối thiểu của s , a_3, a_4 là thuộc tính thứ cấp. Do đó, s là 2NF, nhưng không là 3NF.

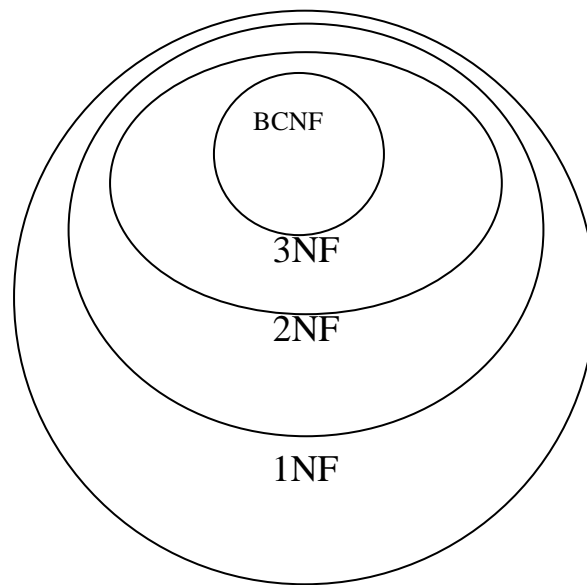
Ví dụ 2:

Cho $t = \langle R, F \rangle$ là sơ đồ quan hệ, với $R = (a_1, a_2, a_3)$

$F = \{ \{a_1, a_2\} \rightarrow \{a_3\}, \{a_3\} \rightarrow \{a_2\} \}$

Ta nhận thấy $\{a_1, a_2\}$ là một khóa tối thiểu của t . Hiển nhiên t là 3NF. Vì có : $\{a_3\} \rightarrow \{a_2\}$, a_3 không thuộc khóa, a_2 cơ bản, nên t không là BCNF.

Như vậy việc phân lớp các dạng chuẩn có thể được thể hiện



Hình 1.1. Phân lớp các dạng chuẩn

1.2.4. Chiến lược thiết kế logic cơ sở dữ liệu quan hệ

1.2.4.1. Chiến lược thiết kế CSDL

Có nhiều chiến lược thiết kế CSDL. Mỗi chiến lược dựa trên những cách tiếp cận và quan điểm riêng biệt tùy theo từng loại bài toán. Có thể kể ra một số chiến lược thiết kế quan trọng như thiết kế CSDL quan hệ, thiết kế CSDL suy diễn, thiết kế CSDL liệu mờ, thiết kế CSDL phân tán, thiết kế CSDL đa phương tiện,.... [3]

Nội dung đồ án sẽ tập trung quan tâm đến CSDL quan hệ.

1.2.4.2. Quy trình chung thiết kế cơ sở dữ liệu quan hệ

Quy trình chung thiết kế cơ sở dữ liệu gồm những giai đoạn cơ bản sau:

1. Phân tích yêu cầu

2. Phân tích dữ liệu và xây dựng mô hình khái niệm dữ liệu
3. Thiết kế CSDL mức logic
4. Thiết kế CSDL mức vật lý
5. Thiết kế an toàn bảo mật cho CSDL

Trong các giai đoạn trên, 3 giai đoạn đầu là rất quan trọng. Vì vậy phần trình bày sau đây chỉ dành cho 3 giai đoạn này.

1.2.4.2.1. Phân tích yêu cầu

Mục đích của việc phân tích yêu cầu:

- Xác định yêu cầu dữ liệu của cơ sở dữ liệu: danh sách các thuộc tính cần quản lý và gộp chúng theo các đối tượng.
- Phân loại và mô tả thông tin về các đối tượng đó
- Xác định mối liên hệ giữa các đối tượng
- Xác định các giao dịch sẽ được thực hiện trên các đối tượng dữ liệu đó.
- Xác định các quy tắc nghiệp vụ (luật) đảm bảo tính toàn vẹn của dữ liệu

Việc phân tích yêu cầu được tiến hành thông qua phương pháp mô hình hóa với trừu tượng logic hay khái niệm. Kết quả của việc phân tích yêu cầu và phân tích dữ liệu là xây dựng được mô hình khái niệm dữ liệu. Trong quá trình xây dựng mô hình này, vai trò của người sử dụng rất quan trọng. Họ có thể bổ sung cho chúng ta những khiếm khuyết của hệ thống hiện tại. Nhờ vậy, chúng ta sẽ có cơ sở để hoàn thiện mô hình với các dữ liệu đầy đủ và chính xác hơn.

Việc phân tích yêu cầu về dữ liệu cho hệ thống gồm những nội dung sau:

1 -Xác định dữ liệu cần lưu giữ

Đối với hệ thống CSDL, mấu chốt vấn đề là xác định đúng và đủ danh sách thuộc tính cần quản lý. Danh sách này quyết định thành công của việc xây dựng CSDL.

Đối với mỗi dữ liệu trong danh sách, cần mô tả đầy đủ các chỉ mục dữ liệu:

- Tên dữ liệu (sát với thực tế)
- Định nghĩa (người dùng hiểu được)

- Kiểu dữ liệu (chỉ rõ kiểu của dữ liệu là ký tự, số, ngày tháng, logic,... Và kích cỡ của chúng)
- Loại dữ liệu (sơ cấp: không thể chia nhỏ được nữa, tích hợp/ kết nối: được gộp lại từ việc ghép nhiều dữ liệu với nhau, tính toán: được suy ra từ một số dữ liệu khác)
- Định lượng (số các giá trị khác nhau mà dữ liệu có thể nhận)
- Ghi chú thêm: cho phép chúng ta dùng hình thức diễn đạt tự do để có thể nêu thêm các dữ liệu bổ sung cho dữ liệu.

2-Xác định ứng dụng sẽ được cài đặt trên CSDL

Việc xác định ứng dụng sẽ được cài đặt trên CSDL là rất quan trọng vì đó là cơ sở xác định đúng và đủ của các dữ liệu và giúp cho việc nhìn nhận tổng quát hơn về cơ sở dữ liệu.

3-Xác định các thao tác thường xuyên thực hiện

Việc xác định thao tác thường xuyên thực hiện trên dữ liệu đóng vai trò quan trọng. Yếu tố này đôi khi lại quyết định tính thành công của một CSDL. Yếu tố này đôi khi quyết định tính thành công của một CSDL. Nhờ có yếu tố này mà việc thiết kế cơ sở dữ liệu mới được coi là đầy đủ.

Trên thực tế, khi lượng dữ liệu tăng lên thì thời gian thực hiện các thao tác trên dữ liệu cũng tăng lên. Do vậy, nếu trong khâu khảo sát, phân tích yêu cầu, chúng ta không xác định được các thao tác thường xuyên được thực hiện thì chúng ta không thể đưa ra được thiết kế hiệu quả, không thể tạo ra những chỉ mục phù hợp, cách phân chia dữ liệu hợp lý.

Đặc biệt, việc xác định các thao tác thường xuyên được thực hiện trên dữ liệu rất có ích trong quá trình tinh chỉnh sơ đồ CSDL mức logic cũng như thiết kế CSDL mức vật lý.

1.2.4.2.2. Phân tích dữ liệu và xây dựng mô hình khái niệm dữ liệu

Mô hình khái niệm dữ liệu:

- Là công cụ kết nối giữa phân tích, thiết kế và người sử dụng
- Là mô hình dữ liệu mức khái niệm, mô tả cấu trúc và các ràng buộc của dữ liệu trong cơ sở dữ liệu.
- Độc lập với hệ quản trị cơ sở dữ liệu và mô hình dữ liệu sẽ được sử dụng để cài đặt.
- Được biểu diễn dạng sơ đồ với 3 thành phần chính là kiểu thực thể, thuộc tính, liên kết (các mối quan hệ)

Quá trình xây dựng mô hình khái niệm dữ liệu có thể được chia làm các giai đoạn sau đây [5]:

- Khảo sát thực tế
- Thiết lập mô hình dữ liệu
- Kiểm soát và chuẩn hóa mô hình

1.2.4.2.3. Thiết kế CSDL mức logic

Thiết kế cơ sở dữ liệu logic là vô cùng quan trọng trong khâu thiết kế CSDL. Kết quả thiết kế cơ sở dữ liệu logic cho cả người phân tích thiết kế hệ thống, người sử dụng và người quản lý hình dung được CSDL của hệ thống thông tin trong một tổ chức bao gồm bao nhiêu tệp, tên từng tệp, danh sách các trường mỗi tệp, nhóm thuộc tính khóa chính trong mỗi tệp cũng như tổng thể các mối quan hệ logic giữa các tệp (kết nối thông qua khóa ngoại). Các kỹ thuật được sử dụng ở đây thể hiện tính công nghệ rõ nét.

Các kỹ thuật sử dụng trong thiết kế cơ sở dữ liệu mức logic bao gồm [4]:

1. Kỹ thuật xác định các thực thể

Một thực thể được xác định nếu xác định được 3 thành phần: tên của thực thể, danh sách thuộc tính (ít nhất là một) và nhóm thuộc tính định danh (ít nhất là một thuộc tính).

2. Kỹ thuật đặc tả

Chúng ta đã biết kỹ thuật đặc tả mối quan hệ giữa 2 thực thể dựa vào mô tả bằng ngôn ngữ tự nhiên. Ngoài cách này chúng ta còn có thể đặc tả mối quan hệ giữa 2 thực thể dựa trên những kỹ thuật sau đây:

- Dựa vào quy tắc quản lý hoặc những quy tắc toàn vẹn
- Dựa vào khóa của các lược đồ quan hệ (Xác định qua khóa liên kết)

3. Kỹ thuật chuyển mô hình khái niệm dữ liệu về hệ lược đồ quan hệ

4. Kỹ thuật chuẩn hóa

Chuẩn hóa thường được hoàn thành sau một số bước, mỗi bước nhận được các quan hệ tương ứng với một dạng chuẩn. Dạng chuẩn được hiểu là một trạng thái của quan hệ có thể được xác định nhờ áp dụng các quy tắc để phát hiện sự phụ thuộc hàm (mối quan hệ) giữa các thuộc tính của quan hệ.

Chuẩn hóa dựa trên cơ sở phân tích các phụ thuộc hàm. Phụ thuộc hàm như đã nói ở trên là một mối quan hệ cụ thể giữa hai thuộc tính (hay nhóm thuộc tính) trong một quan hệ.

Các dữ liệu trong quan hệ không dễ cho thấy về sự tồn tại của các phụ thuộc hàm. Chỉ có sự hiểu biết và tri thức của con người về các đối tượng mà chính quan hệ đó mô tả mới cho phép xác định các phụ thuộc hàm vốn tồn tại trong nó. Tuy có những thuật toán (thường phức tạp) cho phép phát hiện ra có hay không các sự phụ thuộc hàm giữa hai hay nhiều thuộc tính của nó nhưng thường ít dùng.

5. Kỹ thuật chuyển từ hệ lược đồ quan hệ sang sơ đồ E-R (ERD-mô hình dữ liệu mức logic)

Để dễ nhận thức và trao đổi, mô hình E-R thường được biểu diễn dưới dạng một đồ thị, trong đó các nút là các thực thể, còn các cung là các mối quan hệ (các kiểu liên kết các thực thể).

Mô hình được lập như sau:

Mỗi thực thể được biểu diễn bằng một hình chữ nhật có 2 phần: phần trên là tên thực thể (viết in), phần dưới chứa danh sách các thuộc tính, trong đó thuộc tính

khóa được đánh dấu (mỗi thực thể chỉ xác định một khóa tối thiểu). Tên thực thể thường là danh từ (chỉ đối tượng).

Một mối quan hệ được biểu diễn thường bằng hình thoi/elip, được nối bằng nét liền tới các thực thể tham gia vào mối quan hệ đó. Trong hình thoi là tên của mối quan hệ cũng được viết in, danh sách các thuộc tính của nó thì được viết thường. Tên của mối quan hệ thường là động từ chủ động hay bị động. Trong phương pháp MERISE, mối quan hệ thường được biểu diễn bằng hình elip. Mô hình E_R cuối cùng thường mối quan hệ không còn loại N-N. Trong mối quan hệ nhị nguyên thì ở hai đầu mút các đường nối, sát với thực thể, người ta vẽ đường ba chẽ (còn gọi là đường chân gà) về phía có khóa ngoại (khóa liên kết) thể hiện nhiều, còn phía kia thể hiện một. Bản số trong mỗi đặc tả mối quan hệ giữa 2 thực thể là cặp max của hai bản số xác định trong đặc tả và được gọi là bản số trực tiếp [1,6].

CHƯƠNG 2

MÔ HÌNH THIẾT KẾ CSDL QUAN HỆ MỨC LOGIC DỰA TRÊN PHƯƠNG PHÁP BLANPRE

2.1. Ý nghĩa của thiết kế CSDL mức logic

Thiết kế cơ sở dữ liệu mức logic là vô cùng quan trọng trong khâu thiết kế CSDL. Kết quả thiết kế cơ sở dữ liệu logic cho cả người phân tích thiết kế hệ thống, người sử dụng và người quản lý hình dung được CSDL của hệ thống thông tin trong một tổ chức bao gồm bao nhiêu tệp, tên từng tệp, danh sách các trường mỗi tệp, nhóm thuộc tính khóa chính trong mỗi tệp cũng như tổng thể các mối quan hệ logic giữa các tệp (kết nối thông qua khóa ngoại). Các kỹ thuật được sử dụng ở đây thể hiện tính công nghệ rõ nét [4].

2.2. Khuôn cảnh chung các bước thiết kế CSDL mức logic

Vấn đề đặt ra là thế nào là một thiết kế cơ sở dữ liệu tốt?

Để làm rõ vấn đề này chúng ta xét ví dụ sau.

Cho một lược đồ:

THÔNG TIN VỀ NHÀ CUNG CẤP (tên NCC, địa chỉ NCC, mặt hàng, giá)

Lược đồ này chứa một số các thông tin về nhà cung cấp. Nhiều vấn đề có thể nảy sinh trong đó như:

1. Dư thừa (redundancy). Địa chỉ của nhà cung cấp được lập lại mỗi lần cho mỗi mặt hàng được cung cấp.

2. Mâu thuẫn tiềm ẩn (potential inconsistency) hay bất thường khi cập nhật. Do hậu quả của dư thừa, chúng ta có thể cập nhật địa chỉ của một nhà cung cấp trong một bản ghi nhưng vẫn để lại địa chỉ cũ trong một bản ghi khác. Vì vậy chúng ta có thể không có một địa chỉ duy nhất đối với mỗi nhà cung cấp.

3. Bất thường khi chèn (insertion anomaly). Chúng ta không thể biết địa chỉ một nhà cung cấp nếu hiện tại họ chưa cung cấp ít một mặt hàng. Chúng ta có thể đặt giá trị null trong các thành phần mặt hàng và giá của một bản ghi cho người đó, nhưng khi chúng ta nhập mặt hàng cho nhà cung cấp đó, chúng ta có nhớ xóa đi bản

ghi mang giá trị null hay không? Điều tệ hại là mặt hàng và tên NCC cùng tạo ra một khóa cho quan hệ đó, và có lẽ không thể tìm ra các bản ghi nhờ chỉ mục sơ cấp được, nếu có những giá trị null trong trường khóa mặt hàng.

4. Bất thường khi xoá (deletion anomaly). Ngược lại với vấn đề (3) là vấn đề chúng ta có thể xoá tất cả các mặt hàng được cung cấp bởi một người, vô ý làm mất dấu vết để tìm ra địa chỉ của nhà cung cấp này.

Trong ví dụ này, tất cả các vấn đề nảy sinh trên sẽ được giải quyết triệt để khi thay nó bằng hai lược đồ quan hệ mới:

NHACC (tên NCC, địa chỉ NCC)

GIA_NCC (tên NCC, mặt hàng, giá)

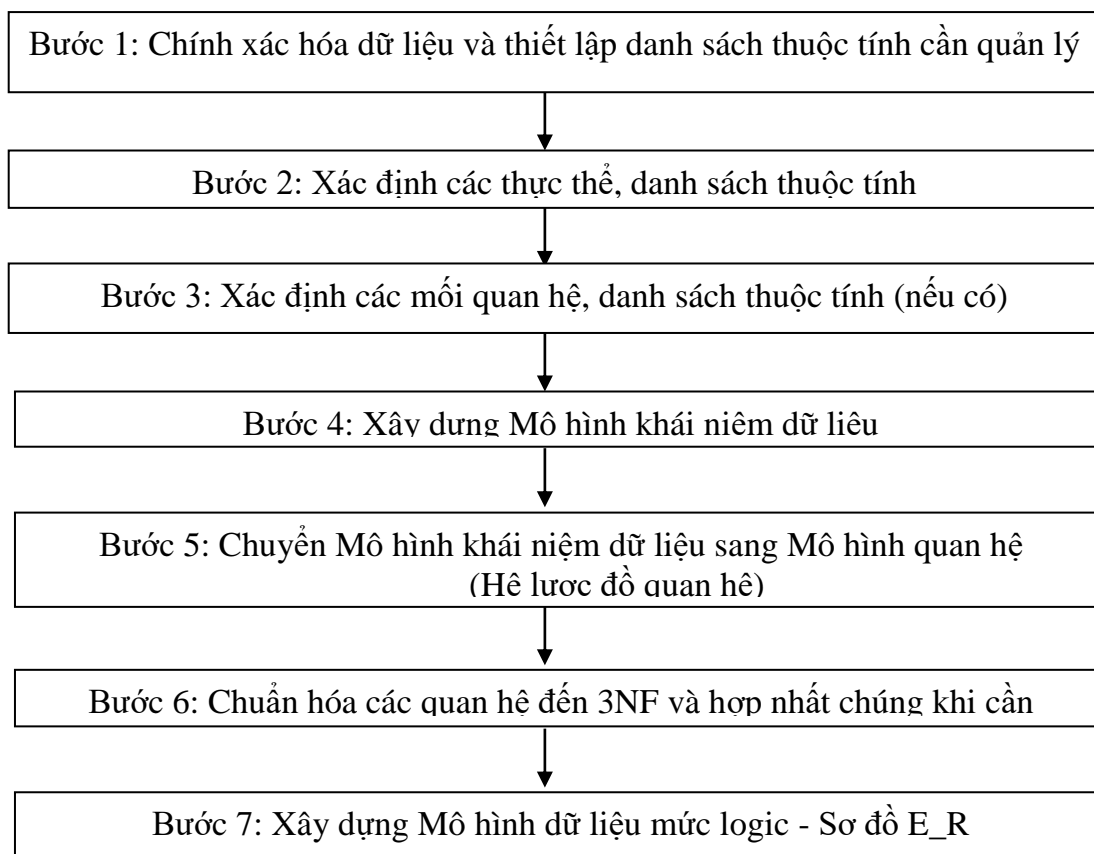
Ở đây quan hệ NHACC cung cấp địa chỉ của mỗi NCC đúng một lần, do vậy không có dư thừa.

Ngoài ra, chúng ta cũng có thể nhập địa chỉ của nhà cung cấp dù hiện tại họ chưa cung cấp một loại mặt hàng nào. Như vậy tư tưởng nảy sinh tự nhiên là làm thế nào để có một CSDL tốt ?

Để đảm bảo quá trình thiết kế dẫn đến một CSDL tốt (theo nghĩa tránh được những nhược điểm trên) người ta thường thực hiện đầy đủ các bước theo một trình tự công nghệ được trình bày ở hình dưới đây.

Đối với phương pháp Blanpre, bước 2 và bước 3 dựa vào ma trận phụ thuộc hàm.

Đối với phương pháp trực giác, việc xác định các mối quan hệ không dựa trên ràng buộc phụ thuộc hàm, hay quy tắc quản lý mà bằng trực giác. Trong đó, các hồ sơ dữ liệu thường được xét tới như một môi quan hệ n ngôi [5].



Hình 2.1. Sơ đồ khuôn cảnh chung các bước thiết kế CSDL mức logic [3]

2.3. Xây dựng mô hình khái niệm dữ liệu bằng phương pháp Blaupre

2.3.1. Ý tưởng của mô hình

Phương pháp Blaupre là phương pháp có độ tin cậy cao. Cơ sở của phương pháp này là đặc tả ràng buộc toàn vẹn bằng phụ thuộc hàm. Các bước thực hiện phương pháp Blaupre bao gồm các công việc sau đây [4]:

2.3.2. Quy trình thiết kế

1. Kiểm kê dữ liệu
2. Xác định các phụ thuộc hàm giữa các dữ liệu
3. Xây dựng mô hình khái niệm dữ liệu

2.3.3. Ví dụ

1. Kiểm kê dữ liệu

Danh sách này chủ yếu được rút tĩa từ những thông tin thu thập được trong giai đoạn khảo sát ban đầu: tài liệu thu thập được; nhu cầu, giải thích của người sử dụng. Danh sách này sẽ làm nền cho việc thiết lập một từ điển dữ liệu về sau.

Có thể phân biệt hai loại dữ liệu:

- Loại dữ liệu xuất hiện trực tiếp trên các tài liệu, màn hình, tệp in thu thập được .
- Loại dữ liệu không hề xuất hiện nhưng cần thiết để chứa các kết quả trung gian, các thông tin đang chờ được xử lý, hay để tính toán ra các dữ liệu thuộc loại thứ nhất.

Một công cụ thông dụng, hữu ích cho giai đoạn này là “Ma trận của Blanpre”, dùng để phân tích các tài liệu thu thập và liệt kê ra danh sách các dữ liệu. Trong ma trận này, ta trình bày mỗi cột là một tài liệu và mỗi hàng là một loại dữ liệu. Tại mỗi ô giao điểm, ta đánh dấu loại dữ liệu có xuất hiện trên tài liệu. Nên dùng hai loại dấu hiệu khác nhau để phân biệt loại dữ liệu trực tiếp (số 1 chẳng hạn) với loại được tính toán thành (số 2 chẳng hạn).

Khi xây dựng ma trận này, ta nên bắt đầu bằng những tài liệu cơ bản, quan trọng nhất và chỉ cần trình bày một loại tài liệu khi nó cho phép nhận dạng ít nhất một loại dữ liệu mới.

Hai chứng từ sau là cần thiết cho quản lý hoạt động kinh doanh:

a. ĐƠN ĐẶT HÀNG

Số:

Ngày:

Mã khách hàng:

Tên khách hàng:

Địa chỉ:

Số TT	Mã hàng	Tên hàng	Đơn vị	Số lượng

b. PHIẾU GIAO HÀNG

Số:

Ngày:

Mã khách hàng:

Nơi giao:

Số TT	Mã hàng	Đơn giá	Số lượng	Thành tiền

TONGTIEN

Bảng 2.1. Ma trận Blanpre

	Loại dữ liệu	Đơn đặt hàng	Phiếu giao hàng
1.	Số đơn	1	
2.	Ngày đặt	1	
3.	Mã khách	1	1
4.	Tên khách	1	
5.	Địa chỉ	1	
6.	STT	2	2
7.	Mã hàng	1	1
8.	Tên hàng	1	
9.	Đơn vị	1	
10.	SL đặt	1	
11.	Số phiếu		1

12.	Ngày giao		1
13.	Nơi giao		1
14.	Đơn giá giao		1
15.	SL giao		1
16.	Thành tiền		2
17.	Tổng tiền		2

Trong bảng 1-dữ liệu trực tiếp, 2-dữ liệu gián tiếp /trung gian (tính được thông qua các dữ liệu khác)

Từ danh sách này, người ta cần thanh lọc:

- Bỏ bớt các dữ liệu đồng nghĩa nhưng khác tên, chỉ giữ lại một

Ví dụ: Mã số sản phẩm =Danh mục đặt hàng

-Phân biệt các dữ liệu cùng tên nhưng khác nghĩa và tách thành nhiều loại dữ liệu khác nhau.

Ví dụ: giá bán của một cửa hiệu khác với giá bán của một công ty sản xuất

-Nhập chung các loại dữ liệu luôn xuất hiện đồng thời với nhau trên mọi loại tài liệu thành một dữ liệu sơ cấp.

Ví dụ: Số nhà và tên đường; ngày, tháng và năm sinh.

-Loại bỏ những loại dữ liệu có thể xác định được một cách duy nhất từ các dữ liệu khác, hoặc bằng công thức tính toán, do các quy luật của tổ chức.

Ví dụ: Tổng giá đơn đặt hàng = Số lượng giá* đơn giá

-Giả sử do quy luật tổ chức, mọi đề nghị mua hàng phải được giải quyết nội trong ngày, ta suy ra: Ngày đề nghị mua hàng = Ngày đặt hàng

Sau khi chính xác hóa dữ liệu (loại đồng danh, đồng nghĩa, thêm bớt thuộc tính, loại bỏ dữ liệu gián tiếp,...) ta có:

Bảng 2.2. Ma trận Blanpre rút gọn

Loại dữ liệu	Đơn đặt hàng	Phiếu giao
Số đơn	1	
Ngày đặt	1	

Mã khách	1	1
Tên khách	1	
địa chỉ	1	
Mã hàng	1	1
Tên hàng	1	
Đơn vị	1	
SL đặt	1	
Số phiếu		1
Ngày giao		1
Nơi giao		1
Đơn giá giao		1
SL giao		1

2. Xác định các phụ thuộc hàm giữa các dữ liệu

Từ danh sách các loại dữ liệu đã thanh lọc của hệ thống thông tin đạt được qua giai đoạn trên, ta phải xác định tất cả các phụ thuộc hàm hiện hữu giữa chúng.

Cu thể, ta phải tự đặt câu hỏi:

Mỗi giá trị của một loại dữ liệu A có tương ứng với một giá trị duy nhất của loại dữ liệu B không ?

Nếu “có” thì B phụ thuộc hàm vào A: $A \rightarrow B$

Ngoài các phụ thuộc hàm có vẻ trái A là một loại dữ liệu sơ cấp) gọi là phụ thuộc hàm sơ cấp), tương đối dễ xác định, ta còn phải nhận diện cả các hàm trong đó vẻ trái A là một tập hợp của nhiều loại dữ liệu (gọi là phụ thuộc hàm đa phần).

Ta tiếp tục đặt câu hỏi: *Cần ấn định giá trị của những loại dữ liệu nào để có thể suy ra một giá trị duy nhất của loại dữ liệu B ?*

Các hàm phụ thuộc sẽ được trình bày dưới dạng một bảng các phụ thuộc hàm như sau:

Bảng 2.3. Ma trận phụ thuộc hàm Blanpre

Loại dữ liệu	Phụ thuộc hàm sơ cấp	Phụ thuộc hàm đa phần
Số đơn		
Ngày đặt	↓ Nhận	
Mã khách		
Tên khách	↓ Có ↑	
Địa chỉ	↓	
Mã hàng		
Tên hàng	↓	
Đơn vị	↓	
<i>SL đặt</i>		↓ DATCACLOAI
Số phiếu		
Ngày giao	↓	
Nơi giao	↓	
Đơn giá giao		↓ ↓
<i>SL giao</i>		↓ ↓ GIAOCAC LOAI

3. Xây dựng mô hình khái niệm dữ liệu [4]

Giai đoạn này bao gồm 5 bước:

- Xác định tập hợp các khóa chính
- Nhận diện các thực thể
- Nhận diện các quan hệ
- Phân bố hết các thuộc tính
- Vẽ mô hình khái niệm dữ liệu

Xác định tập hợp các khóa chính

Tập hợp K của những khóa chính là tập hợp tất cả những loại dữ liệu có đóng vai trò nguồn (thuộc về trái) trong ít nhất một phụ thuộc hàm.

Trong ví dụ trên, $K = \{\text{Số đơn, mã khách, mã hàng, số phiếu}\}$

Nhận diện các thực thể

Mỗi phần tử của tập hợp K sẽ là khóa chính của một thực thể.

Trong ví dụ trên, ta nhận ra được 4 thực thể:

ĐƠN HÀNG (Số đơn,...)

KHÁCH (Mã khách,...)

HÀNG (Mã hàng,...)

PHIẾU GIAO (Số phiếu,.....)

Nhận diện các quan hệ

Có hai trường hợp:

A. Nếu gốc của một phụ thuộc hàm bao gồm ít nhất 2 phần tử thuộc tập hợp K thì nó tương ứng với một quan hệ N-N giữa các thực thể có khóa chính là các phần tử này.

Trong ví dụ trên, ta nhận ra được 2 quan hệ:

ĐẶT CÁC LOẠI (số lượng đặt)

GIAO CÁC LOẠI (số lượng giao, giá)

B. Sự hiện hữu của một phụ thuộc hàm giữa hai phần tử của tập hợp K xác định một quan hệ nhị nguyên kiểu 1-N giữa hai thực thể có khóa chính là các phần tử này.

Trong ví dụ trên, ta nhận ra được 2 quan hệ :

CÓ và NHẬN

Phân bổ các thuộc tính còn lại

Tất cả các loại dữ liệu không tập hợp K đều là những thuộc tính thông thường và có thể được phân bổ một cách dễ dàng vào các thực thể hay quan hệ, tùy theo các hàm phụ thuộc.

Phân bổ các thuộc tính còn lại cho các thực thể ta thấy vừa hết:

ĐƠN HÀNG (Số đơn, ngày đặt)

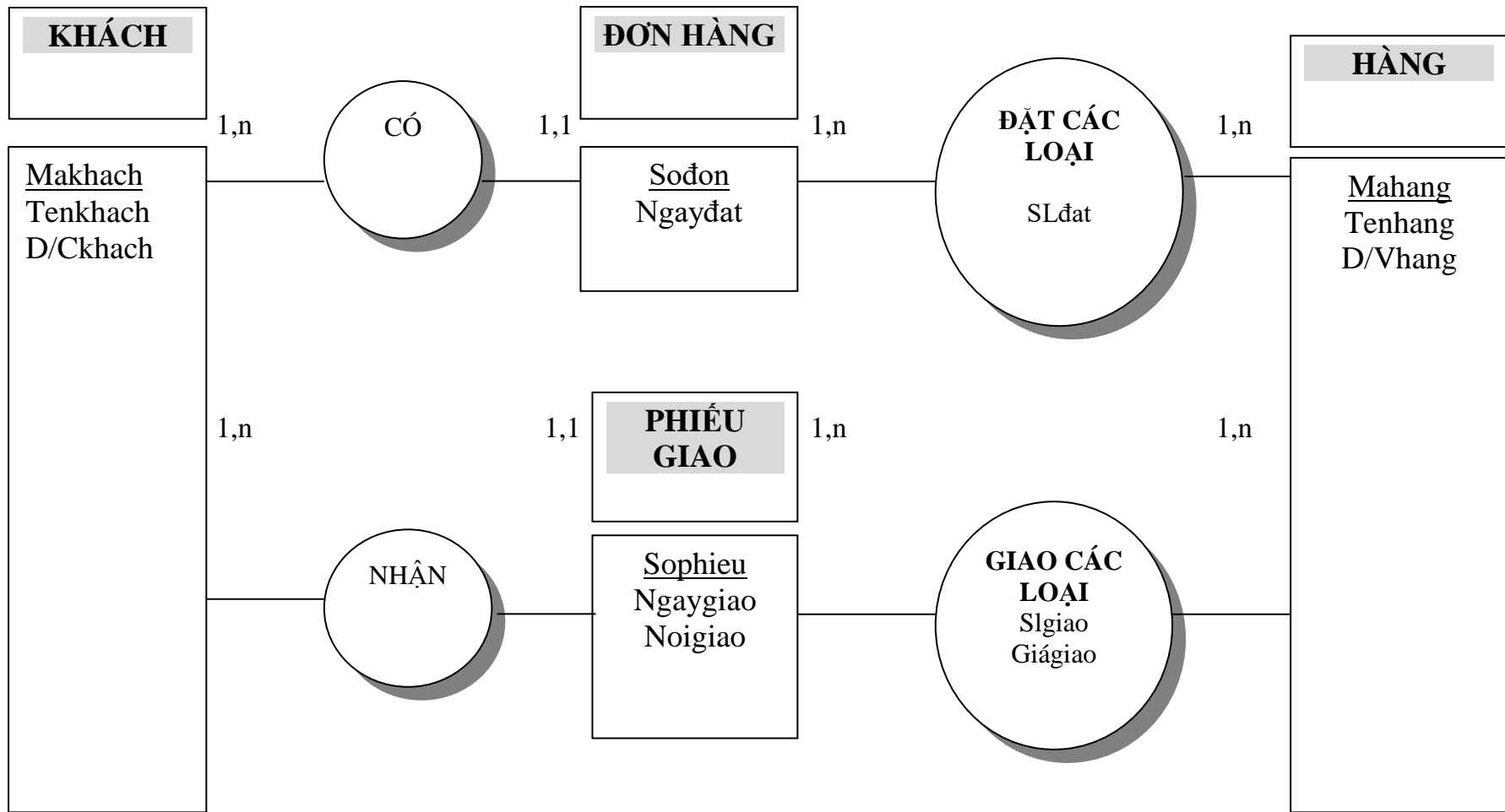
KHÁCH (Mã khách, tên khách, Đ/Ckhách)

HÀNG (Mã hàng, Tên hàng, Mô tả hàng, Đ/Vhàng)

PHIẾU GIAO (Số phiếu, ngày giao, nơi giao)


Vẽ mô hình khái niệm dữ liệu

Từ các thực thể và quan hệ đã nhận diện, ta có thể vẽ lên một mô hình khái niệm dữ liệu như sau:



Hình 2.2. Mô hình khái niệm dữ liệu

Hình chữ nhật biểu diễn thực thể

Hình  biểu diễn mối quan hệ

thuộc tính khóa được gạch chân

2.4. Chuyển mô hình khái niệm dữ liệu sang mô hình dữ liệu mức logic

2.4.1. Kỹ thuật chuyển mô hình khái niệm dữ liệu về hệ lược đồ quan hệ

📖 Có 3 quy tắc chuyển [3]:

Quy tắc 1 (quy tắc biến đổi cơ bản):

Mỗi **thực thể** được chuyển thành một quan hệ trong đó các thuộc tính của thực thể được chuyển thành thuộc tính của quan hệ, định danh của thực thể trở thành khóa của quan hệ.

Quy tắc 2:

Mỗi **mối quan hệ** 1-N mà không có thuộc tính riêng sẽ không được chuyển thành một quan hệ. Nhưng thực thể tham gia vào mối quan hệ về phía N (phía 1:1 trong mô hình) sẽ đổi mới bằng cách sau khi dùng phép biến đổi cơ bản sẽ nhận thêm khóa của thực thể tham gia vào mối quan hệ ở phía 1 làm khóa liên kết. Còn thực thể tham gia vào mối quan hệ ở phía 1 sẽ biến đổi theo quy tắc 1.

Quy tắc 3:

Mỗi **mối quan hệ** N-N hoặc mối quan hệ có thuộc tính riêng sẽ được chuyển thành một quan hệ mới. Quan hệ mới này có thuộc tính gồm định danh của tất cả các thực thể trong mối quan hệ và các thuộc tính riêng của nó. Khóa của quan hệ được xác định lại sau đó. Các thực thể tham gia vào mối quan hệ đều biến đổi theo quy tắc 1.

Sau khi chuyển mô hình khái niệm dữ liệu về hệ lược đồ quan hệ, chúng ta cần chuẩn hóa hệ lược đồ này đến 3NF.

2.4.2. Kỹ thuật chuẩn hóa

Chuẩn hóa là một quá trình chuyển một cấu trúc dữ liệu phức hợp thành các cấu trúc dữ liệu tốt và đơn giản hơn [3].

Chuẩn hóa thường được hoàn thành sau một số bước, mỗi bước nhận được các quan hệ tương ứng với một dạng chuẩn. Dạng chuẩn được hiểu là một trạng thái của quan hệ có thể được xác định nhờ áp dụng các quy tắc để phát hiện sự phụ thuộc hàm (mối quan hệ) giữa các thuộc tính của quan hệ.

Chuẩn hóa dựa trên cơ sở phân tích các phụ thuộc hàm. Phụ thuộc hàm như đã nói ở trên là một mối quan hệ cụ thể giữa hai thuộc tính (hay nhóm thuộc tính) trong một quan hệ.

Các dữ liệu trong quan hệ không dễ cho thấy về sự tồn tại của các phụ thuộc hàm. Chỉ có sự hiểu biết và tri thức của con người về các đối tượng mà chính quan hệ đó mô tả mới cho phép xác định các phụ thuộc hàm vốn tồn tại trong nó. Tuy có những thuật toán (thường phức tạp) cho phép phát hiện ra có hay không các sự phụ thuộc hàm giữa hai hay nhiều thuộc tính của nó nhưng thường ít dùng.

a. Các dạng chuẩn cơ bản

Có 3 dạng chuẩn cơ bản là:

Chuẩn 1 (first-normal-form: 1NF): Một quan hệ đạt chuẩn 1 nếu nó không chứa các thuộc tính lặp.

Chuẩn 2 (second-normal-form: 2NF) (Các thuộc tính ngoài khóa phụ thuộc hoàn toàn vào khóa)

Một quan hệ đạt chuẩn 2 nếu:

- Đạt chuẩn 1
- Không tồn tại thuộc tính ngoài khóa mà phụ thuộc vào một phần của khóa.

Chuẩn 3 (third-normal-form: 3NF) (Các thuộc tính ngoài khóa phụ thuộc trực tiếp vào khóa)

Một quan hệ đạt chuẩn 3 nếu:

- Đạt chuẩn 2
- Không tồn tại thuộc tính ngoài khóa mà phụ thuộc bắc cầu vào khóa (qua một thuộc tính gọi là *thuộc tính cầu* (cũng là thuộc tính ngoài khóa)).

Để chuẩn hóa một hệ lược đồ quan hệ, ta xét lần lượt từng quan hệ và kiểm tra tính chuẩn của nó. Muốn vậy, trước hết ta *xác định các phụ thuộc hàm và khóa chính* (khóa tối thiểu) của quan hệ. Sau đó tiến hành kiểm tra lần lượt các loại chuẩn đối với quan hệ.

Ở đây, chúng ta lưu ý một số điểm khi chuẩn hóa.

Xét chuẩn 1:

+ Nếu quan hệ còn các thuộc tính có dấu * (thuộc tính lặp) nghĩa là quan hệ đó chưa đạt chuẩn 1. Ta sử dụng quy tắc tách bình thường.

+ Trong trường hợp có nhiều nhóm thuộc tính lặp (xen kẽ với các thuộc tính không lặp) ta nên đồng thời tách theo từng nhóm lặp. Hay nói rõ hơn, khi có m nhóm lặp ta sẽ tách lược đồ đó theo quy tắc đã biết để thu được đồng thời m+1 lược đồ quan hệ mới đạt chuẩn 1.

Xét chuẩn 2:

+ Mọi quan hệ đã đạt chuẩn 1, chưa đạt chuẩn 2, đều dễ dàng tách thành các quan hệ đạt chuẩn 2 theo quy tắc đã biết.

+ Trường hợp nhóm thuộc tính khóa chỉ có một thuộc tính, điều này có nghĩa là phụ thuộc bộ phận không thể xảy ra, ta kết luận ngay là quan hệ đó đương nhiên đạt chuẩn 2.

+ Trong trường hợp có m nhóm phụ thuộc bộ phận, ta sẽ tách đồng thời thành m+1 quan hệ đạt chuẩn 2 dựa theo quy tắc đã biết.

Xét chuẩn 3:

Cũng giống như các chuẩn trên, khi có m thuộc tính cầu, ta sẽ tách được m+1 quan hệ mới đạt chuẩn 3.

+ Trường hợp nhóm thuộc tính ngoài khóa chỉ có 1 thuộc tính, điều này có nghĩa không thể tồn tại thuộc tính cầu, nên quan hệ đó đương nhiên đạt chuẩn 3.

2.4.3. Kỹ thuật chuyển từ hệ lược đồ quan hệ sang sơ đồ E_R (ERD - mô hình dữ liệu mức logic)

Để dễ nhận thức và trao đổi, mô hình E_R thường được biểu diễn dưới dạng một đồ thị, trong đó các nút là các thực thể, còn các cung là các mối quan hệ (các kiểu liên kết các thực thể).

Mô hình được lập như sau:

Mỗi thực thể được biểu diễn bằng một hình chữ nhật có 2 phần: phần trên là tên thực thể (viết in), phần dưới chứa danh sách các thuộc tính, trong đó thuộc tính khóa được đánh dấu (mỗi thực thể chỉ xác định một khóa tối thiểu). Tên thực thể thường là danh từ (chỉ đối tượng).

Một mối quan hệ được biểu diễn thường bằng hình thoi/elíp, được nối bằng nét liền tới các thực thể tham gia vào mối quan hệ đó. Trong hình thoi là tên của mối quan hệ cũng được viết in, danh sách các thuộc tính của nó thì được viết thường. Tên của mối quan hệ thường là động từ chủ động hay bị động. Trong phương pháp MERISE, mối quan hệ thường được biểu diễn bằng hình elíp. Mô hình E_R cuối cùng thường mối quan hệ không còn loại N - N. Trong mối quan hệ nhị nguyên thì ở hai đầu nút các đường nối, sát với thực thể, người ta vẽ đường ba chẽ (còn gọi là đường chân gà) về phía có khóa ngoại (khóa liên kết) thể hiện nhiều, còn phía kia thể hiện một. Bản số

trong mỗi đặc tả mối quan hệ giữa 2 thực thể là cặp Max của hai bản số xác định trong đặc tả và được gọi là bản số trực tiếp.

Chú ý:

+ Mỗi quan hệ có thể không có thuộc tính. Khi có, ta thường gọi là thuộc tính riêng và cũng được viết trong hình thoi song nhớ rằng chỉ viết chữ thường (phân biệt tên của mối quan hệ viết bằng chữ in)

+ Giữa 2 thực thể có thể có nhiều mối quan hệ và chúng cần vẽ riêng rẽ, không chập vào nhau.

Việc chuyển từ hệ lược đồ quan hệ sang sơ đồ E_R được thực hiện theo các bước sau:

Bước 1:

Từ mỗi lược đồ quan hệ, vẽ một hình chữ nhật bao gồm tên lược đồ cùng với các thuộc tính của nó:

Bước 2: Xác định quan hệ (thể hiện bằng đường nối) giữa 2 thực thể (thể hiện bằng hình chữ nhật) bất kỳ.

Bước 2.1: Xác định nhóm thuộc tính chung.

Nếu không có nhóm thuộc tính chung thì kết luận: không có quan hệ

Bước 2.2: Kiểm tra nhóm thuộc tính chung đó có phải là khóa chính của 1 trong 2 thực thể?

Nếu không có thì kết luận không có quan hệ: tên công ty là thuộc tính chung nhưng không phải là khóa chính của thực thể nào cả.

Bước 2.3: Kết luận tồn tại khóa ngoại: Nhóm thuộc tính chung ở bên thực thể không phải là khóa chính mà sẽ là khóa ngoại.

Bước 3: Vẽ đường nối giữa 2 hình chữ nhật và đường 3 chẽ gắn với thực thể chứa khóa ngoại.

CHƯƠNG 3

ỨNG DỤNG THIẾT KẾ CSDL VỀ THÔNG TIN CÁC CUNG ĐƯỜNG BỘ TRÊN ĐỊA BÀN TP. HẢI PHÒNG

3.1. Bài toán quản lý thông tin các cung đường bộ trên địa bàn TP. Hải Phòng

Hải Phòng có **15** đơn vị hành chính gồm 7 quận nội thành: *Đồ Sơn, Dương Kinh, Hải An, Hồng Bàng, Kiến An, Lê Chân, Ngô Quyền*; và 8 huyện: *An Dương, An Lão, Bạch Long Vỹ, Cát Hải, Kiến Thụy, Thủy Nguyên, Tiên Lãng, Vĩnh Bảo*.

Thành phố Hải Phòng có khoảng 600 tuyến đường phố, nằm trong 7 quận nội thành. Các con phố ở Hải Phòng đều rất sạch sẽ, nhỏ, hẹp, nhưng rất hiếm xảy ra ùn tắc. Có một vài con phố có mặt bằng rộng là Lạch Tray, Bạch Đằng, Lê Hồng Phong,...

Thông tin về giao thông trên địa bàn TP. Hải Phòng (ở đây gọi tắt là HP) được thu thập và lưu trữ cũng như xử lý ngày càng nhiều. Vấn đề đặt ra là cần tổ chức, khai thác thông tin về giao thông một cách khoa học nhằm giúp các nhà quản lý có cái nhìn chính xác về thực trạng giao thông và nhanh chóng đưa ra những quyết định đúng đắn, kịp thời và người dân có thể nắm bắt được thông tin các cung đường trước khi quyết định tham gia giao thông

Trong giới hạn chương 3 của khóa luận này, sinh viên chỉ tổ chức quản lý khai thác thông tin về đường bộ, cụ thể là các cung đường trên địa bàn 15 đơn vị hành chính của HP. Nội dung chính của phần này là ứng dụng kết quả nghiên cứu của phân lý thuyết trên để thiết kế CSDL về thông tin các cung đường bộ trên địa bàn. Theo cách làm truyền thống, để thiết kế một CSDL tập trung, chúng ta sử dụng các kỹ năng thiết kế CSDL như thiết kế CSDL kiểu “tờ điện”, kiểu “mô hình”, kiểu “Blanpre”, kiểu “trực giác”,... Ở đây, sinh viên sẽ thiết kế một CSDL tập trung dựa vào lý thuyết CSDL tập trung (dựa vào thuật toán thiết kế kiểu “Blanpre”).

3.2. Những phần mềm hỗ trợ

Hệ quản trị cơ sở dữ liệu: SQL Server 2019

Môi trường lập trình: Visual Studio 2019

3.3. Thuật toán sử dụng và xác định dữ liệu đầu vào

3.3.1. Thuật toán sử dụng

Như phần trên đã nói, thuật toán được sử dụng ở đây chính là quy trình thiết kế theo kiểu “Blanpre” đã trình bày trong chương 2.

3.3.2. Dữ liệu đầu vào

Để xác định dữ liệu đầu vào, trước hết, chúng ta sẽ mô tả chi tiết các thông tin cần thiết về các cung đường.

Thông tin về ĐƯỜNG gồm có Tên đường, thuộc một Quận/Huyện nào đó, Chiều dài đường, Loại mặt đường, Kiểu đường, Hệ thống chiếu sáng, Cách thức tổ chức giao thông, Mức độ hư hỏng, Giao cắt đường sắt, Hệ thống tín hiệu, Mật độ lưu thông, Mã thông tin bảo trì, Đơn vị bảo trì, Loại bảo trì, Nội dung bảo trì, Giá trị bảo trì, Thời gian bắt đầu đợt bảo trì nào đó, Thời gian kết thúc đợt bảo trì nào đó,...

- Loại mặt đường: bao gồm Bê tông nhựa, Bê tông xi măng (BTXM), Đá dăm trộn nhựa, Thảm nhập nhựa; láng nhựa 2, 3 lớp, Đá dăm tiêu chuẩn, cấp phối đá dăm, Cấp phối thiên nhiên)

- Kiểu đường: gồm Đường liên quận/huyện, Đường nội quận/huyện.
- Hệ thống chiếu sáng: xác định đường có hệ thống chiếu sáng hay không
- Cách thức tổ chức giao thông (tổ chức GT): Xác định cách tổ chức đường giao thông (đường 1 chiều, đường 2 chiều có giải phân cách cứng, đường 2 chiều không có giải phân cách)

- Mức độ hư hỏng: Xác định mức độ hư hỏng của đường:

Xếp loại 1 và 2: ít hoặc không có vết nứt hay biến dạng, tình trạng mặt đường tốt, không cần sửa chữa hoặc công tác sửa chữa có thể trì hoãn được.

Xếp loại 3 và 4: nứt nhưng không có hoặc ít biến dạng; tình trạng mặt đường không tốt cần tiến hành sửa chữa không cần nghiên cứu gì thêm.

Xếp loại 5, 6, 7: nứt và biến dạng trên diện rộng; tình trạng lớp mặt đường rất xấu và tồi tệ cần sửa chữa lớn hoặc tiến hành trải thảm.

- Giao cắt đường sắt (ĐS): xác định đường có giao cắt với đường sắt hay không
- Hệ thống tín hiệu: xác định đường có hệ thống tín hiệu đèn giao thông hay không
- Mật độ lưu thông: xác định mật độ xe lưu thông
- Loại bảo trì: Xác định loại bảo trì (Sửa chữa thường xuyên, sửa chữa định kỳ, sửa chữa đột xuất).
- Đơn vị bảo trì: Xác định đơn vị bảo trì như tên, địa chỉ, email, điện thoại,...
- Thông tin bảo trì: Xác định thông tin của việc bảo trì đường bộ (tên cung đường tiến hành bảo trì, loại bảo trì, Nội dung các công việc trong đợt bảo trì (Các hạng mục công việc bảo trì), Đơn vị bảo trì, Tổng giá trị của đợt bảo trì, Thời gian bắt đầu bảo trì, Thời gian kết thúc bảo trì).

Tiếp theo, chúng ta sẽ thiết lập mô hình dữ liệu bằng phương pháp Blaupre.

Các loại dữ liệu cần thiết:

Danh sách các thuộc tính:

R= {Tên đường, Tên quận/huyện, Kiểu đường, Tổ chức GT, Loại mặt đường, Mức độ hư hỏng, Chiều dài, Hệ thống chiếu sáng, Loại giao cắt, Hệ thống tín hiệu, Mật độ lưu thông, Mã thông tin bảo trì, Tên đơn vị bảo trì, Địa chỉ, Điện thoại, Email, Loại bảo trì, Nội dung bảo trì, Tổng giá trị, Thời gian bắt đầu, Thời gian kết thúc }

Xác định quan hệ giữa các dữ liệu

Qua khảo sát, chúng ta biết quan hệ phụ thuộc giữa các loại dữ liệu:

1-Mã đường → Tên đường, Mã quận/huyện, Mã Kiểu đường , Mã Tổ chức GT, Mã Loại mặt đường, Mã Mức độ hư hỏng, Chiều dài, Hệ thống chiếu sáng, Giao cắt ĐS, Hệ thống tín hiệu, Mật độ lưu thông.

2-Mã quận/huyện → Tên quận/huyện

3-Mã Kiểu đường → Kiểu đường

4-Mã Tổ chức GT → Tổ chức GT

5- Mã Loại mặt đường → Loại mặt đường

6-Mã Mức độ hư hỏng → Mức độ hư hỏng

7- Mã đơn vị bảo trì → Tên đơn vị bảo trì, Địa chỉ, Điện thoại, Email

8-Mã Loại bảo trì → Loại bảo trì

9-Mã thông tin bảo trì → Mã đơn vị bảo trì, mã đường, mã loại bảo trì, Nội dung bảo trì, Tổng giá trị, Thời gian bắt đầu, Thời gian kết thúc.

3.4. Nội dung và kết quả thử nghiệm

3.4.1. Nội dung thiết kế cơ sở dữ liệu các cung đường TP. Hải Phòng

+*Bổ sung các thuộc tính định danh:*

R= {Mã đường, Tên đường, Mã quận/huyện, Tên quận/huyện, Mã Kiểu đường, Kiểu đường, Mã Tổ chức GT, Tổ chức GT, Mã Loại mặt đường, Loại mặt đường, Mã Mức độ hư hỏng, Mức độ hư hỏng, Chiều dài, Hệ thống chiếu sáng, Giao cắt ĐS, Hệ thống tín hiệu, Mật độ lưu thông, Mã đơn vị bảo trì, Tên đơn vị bảo trì, Địa chỉ, Điện thoại, Email, Mã thông tin bảo trì, Mã Loại bảo trì, Loại bảo trì, Nội dung bảo trì, Tổng giá trị, Thời gian bắt đầu, Thời gian kết thúc}.

-*Mã hóa tập thuộc tính R:*

R= {a01-Mã đường,a02-Tên đường,a03-Mã quận/huyện, a04-Tên quận/huyện, a05-Mã Kiểu đường, a06-Kiểu đường, a07-Mã Tổ chức GT, a08-Tổ chức GT, a09-Mã Loại mặt đường, a10-Loại mặt đường, a11-Mã Mức độ hư hỏng, a12-Mức độ hư hỏng, a13-Chiều dài, a14-Hệ thống chiếu sáng, a15-Giao cắt ĐS , a16-Hệ thống tín hiệu, a17-Mật độ lưu thông, a18-Mã đơn vị bảo trì, a19-Tên đơn vị bảo trì, a20-Địa chỉ, a21-Điện thoại, a22-Email, a23-Mã thông tin bảo trì, a24-Mã Loại bảo trì, a25-Loại bảo trì, a26-Nội dung bảo trì, a27-Tổng giá trị, a28-Thời gian bắt đầu, a29-Thời gian kết thúc}

Xây dựng Ma trận Blanpre:

Loại dữ liệu	Hồ sơ quản lý
1. a01-Mã đường	1
2. a02-Tên đường	1
3. a03-Mã quận/huyện	1
4. a04-Tên quận/huyện	1
5. a05-Mã Kiểu đường	1

6. a06-Kiểu đường	1
7. a07-Mã Tổ chức GT	1
8. a08-Tổ chức GT	1
9. a09-Mã Loại mặt đường	1
10. a10-Loại mặt đường,	1
11. a11-Mã Mức độ hư hỏng	1
12. a12-Mức độ hư hỏng,	1
13. a13-Chiều dài	1
14. a14-Hệ thống chiếu sáng	1
15. a15-Giao cắt ĐS	1
16. a16-Hệ thống tín hiệu	1
17. a17-Mật độ lưu thông	1
18. a18-Mã đơn vị bảo trì	1
19. a19-Tên đơn vị bảo trì	1
20. a20-Địa chỉ,	1
21. a21-Điện thoại,	1
22. a22-Email	1
23. a23-Mã thông tin bảo trì	1
24. a24-Mã Loại bảo trì,	1
25. a25-Loại bảo trì,	1
26. a26-Nội dung bảo trì,	1
27. a27-Tổng giá trị	1
28. a28-Thời gian bắt đầu	1
29. a29-Thời gian kết thúc	1

Ma trận phụ thuộc hàm Blanpre:

Xác định phụ thuộc hàm giữa các dữ liệu:

Từ danh sách các loại dữ liệu đã thanh lọc của hệ thống thông tin đạt được qua giai đoạn trên, ta phải xác định tất cả các phụ thuộc hàm hiện hữu giữa chúng.

Cụ thể, ta phải tự đặt câu hỏi:

Mỗi giá trị của một loại dữ liệu A có tương ứng với một giá trị duy nhất của loại dữ liệu B không?

Nếu “có” thì B phụ thuộc hàm vào A: $A \rightarrow B$

Ngoài các phụ thuộc hàm có vẻ trái A là một loại dữ liệu sơ cấp (gọi là phụ thuộc hàm sơ cấp), tương đối dễ xác định, ta còn phải nhận diện cả các hàm trong đó vẻ trái A là một tập hợp của nhiều loại dữ liệu (gọi là phụ thuộc hàm đa phần).

Ta tiếp tục đặt câu hỏi:

Cần ấn định giá trị của những loại dữ liệu nào để có thể suy ra một giá trị duy nhất của loại dữ liệu B?

Kết quả cuối cùng chúng ta có:

1-Mã đường \rightarrow Tên đường, Mã phường/xã, Mã Kiểu đường, Mã Tổ chức GT, Mã Loại mặt đường, Mã Mức độ hư hỏng, Chiều dài, Hệ thống chiếu sáng, Giao cắt ĐS, Hệ thống tín hiệu, Mật độ lưu thông.

2-Mã quận/huyện \rightarrow Tên quận/huyện

3-Mã Kiểu đường \rightarrow Kiểu đường

4-Mã Tổ chức GT \rightarrow Tổ chức GT

5- Mã Loại mặt đường \rightarrow Loại mặt đường

6-Mã Mức độ hư hỏng \rightarrow Mức độ hư hỏng

7- Mã đơn vị bảo trì \rightarrow Tên đơn vị bảo trì, Địa chỉ, Điện thoại, Email

8-Mã Loại bảo trì \rightarrow Loại bảo trì

9-Mã thông tin bảo trì \rightarrow Mã đơn vị bảo trì, mã đường, mã loại bảo trì, Nội dung bảo trì, Tổng giá trị, Thời gian bắt đầu, Thời gian kết thúc.

Tập phụ thuộc hàm F được xác định:

a01 \rightarrow a02, a03, a05, a07, a09, a11, a13, a14, a15, a16, a17

a03 \rightarrow a04

a05 \rightarrow a06

a07 \rightarrow a08

a09 \rightarrow a10

a11 \rightarrow a12

a18 \rightarrow a19, a20, a21, a22

a24 \rightarrow a25

a23 → a18, a01, a24, a26, a27, a28, a29

Ma trận phụ thuộc hàm Blaupre dưới dạng bảng:

Loại dữ liệu	Phụ thuộc hàm sơ cấp	PTH đa phân
30.a01-Mã đường		
31.a02-Tên đường	↓	
32.a03-Mã quận/huyện	↓	
33.a04-Tên quận/huyện	↓	
34.a05-Mã Kiểu đường	↓	
35.a06-Kiểu đường	↓	
36.a07-Mã Tổ chức GT	↓	
37.a08-Tổ chức GT	↓	
38.a09-Mã Loại mặt đường	↓	
39.a10-Loại mặt đường,	↓	
40.a11-Mã Mức độ hư hỏng	↓	
41.a12-Mức độ hư hỏng,		
42.a13-Chiều dài	↓	
43.a14-Hệ thống chiếu dung	↓	
44.a15-Giao cắt ĐS	↓	
45.a16-Hệ thống tín hiệu	↓	
46.a17-Mật độ lưu thông	↓	

47.a18-Mã đơn vị bảo trì		
48.a19-Tên đơn vị bảo trì	↓	
49.a20-Địa chỉ,	↓	
50.a21-Điện thoại,	↓	
51.a22-Email	↓	
52.a23-Mã thông tin bảo trì	↑	
53.a24-Mã Loại bảo trì,	↓	
54.a25-Loại bảo trì,	↓	
55.a26-Nội dung bảo trì,	↓	
56.a27-Tổng giá trị	↓	
57.a28-Thời gian bắt đầu	↓	
58.a29-Thời gian kết thúc	↓	
1.a01-Mã đường	↓	

Xây dựng mô hình khái niệm dữ liệu:

Giai đoạn này bao gồm 5 bước:

- Xác định tập hợp các khóa chính
- Nhận diện các thực thể
- Nhận diện các quan hệ
- Phân bố hết các thuộc tính
- Vẽ mô hình khái niệm dữ liệu

1. Xác định tập hợp các khóa chính:

Tập hợp K của những khóa chính là tập hợp tất cả những loại dữ liệu có đóng vai trò nguồn (thuộc về trái) trong ít nhất một phụ thuộc hàm.

Trong hệ thống trên, ta có :

$K = \{ \text{Mã đường, Mã phường/xã, Mã Kiểu đường, Mã Tổ chức GT, Mã Loại mặt đường, Mã Mức độ hư hỏng, Mã đơn vị bảo trì, Mã Loại bảo trì, Mã thông tin bảo trì} \}$

2. Nhận diện các thực thể:

Mỗi phần tử của tập hợp K sẽ là khóa chính của một thực thể.

Trong hệ thống trên, ta có 9 thành phần trong K nên nhận ra được 9 thực thể:

R1-ĐƯỜNG (Mã đường, ...)

R2-QUẬN/HUYỆN (Mã quận/huyện, ...)

R3-KIỂUĐƯỜNG (Mã Kiểu đường, ...)

R4-TỔCHỨC GIAOTHÔNG (Mã Tổ chức GT, ...)

R5-LOẠI MẶT ĐƯỜNG (Mã Loại mặt đường, ...)

R6-MỨCĐỘHƯHỎNG (Mã Mức độ hư hỏng, ...)

R7-ĐƠN VỊ BẢO TRÌ (Mã đơn vị bảo trì,...)

R8-LOẠI BẢO TRÌ (Mã Loại bảo trì, ...)

R9-BẢO TRÌ (Mã thông tin bảo trì, ...)

3. Nhận diện các quan hệ:

Có hai trường hợp:

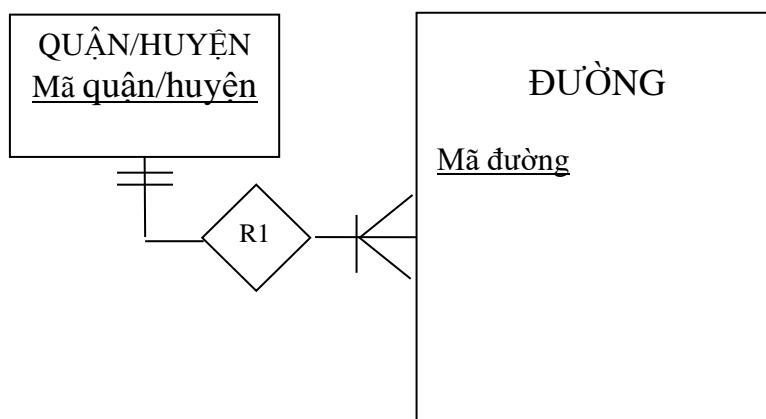
3.1. Nếu gốc của một phụ thuộc hàm bao gồm ít nhất 2 phần tử thuộc tập hợp K thì nó tương ứng với một quan hệ $N-N$ giữa các thực thể có khóa chính là các phần tử này.

Trong hệ thống trên, ta không nhận được quan hệ nào loại này.

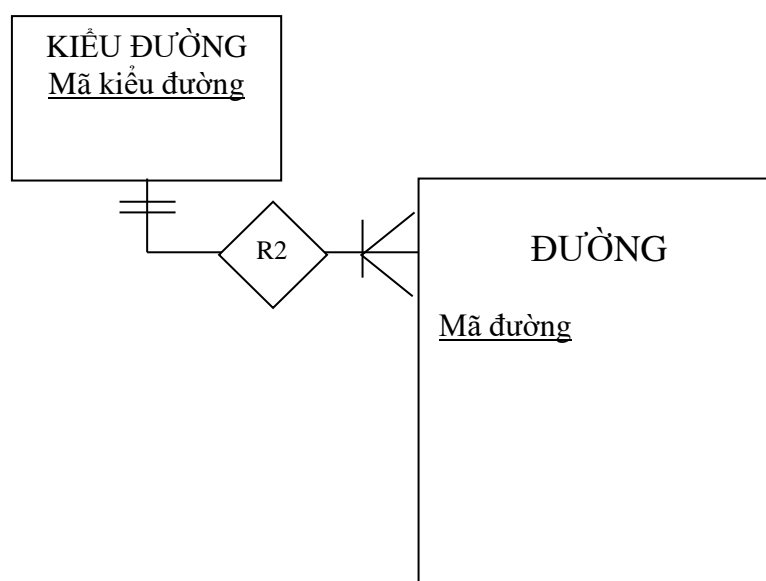
3.2. Sự hiện hữu của một phụ thuộc hàm giữa hai phần tử của tập hợp K xác định một quan hệ nhị nguyên kiểu $1-N$ giữa hai thực thể có khóa chính là các phần tử này.

Trong hệ thống trên, ta nhận ra được 8 quan hệ:

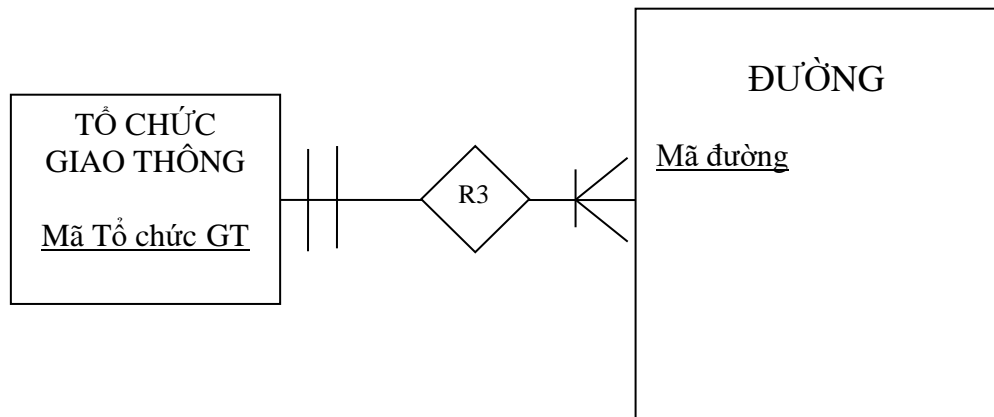
Vì mã đường, mã quận/huyện đều thuộc K , mã đường \rightarrow mã Quận/huyện nên:



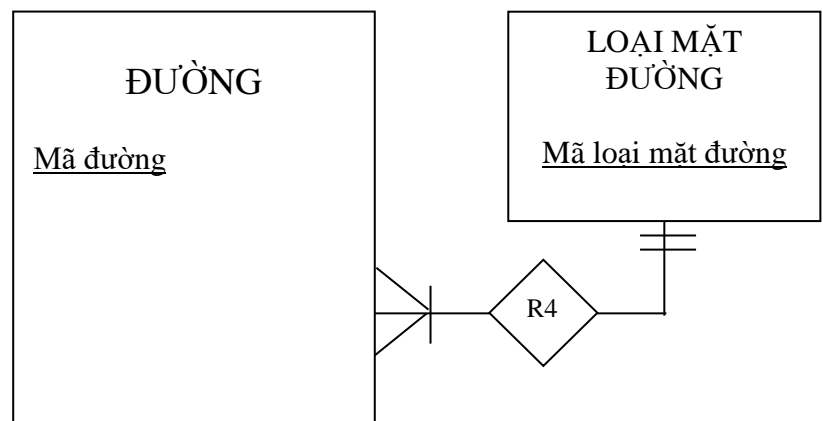
Vì mã đường, mã kiểu đường thuộc K , mã đường \rightarrow mã kiểu đường nên:



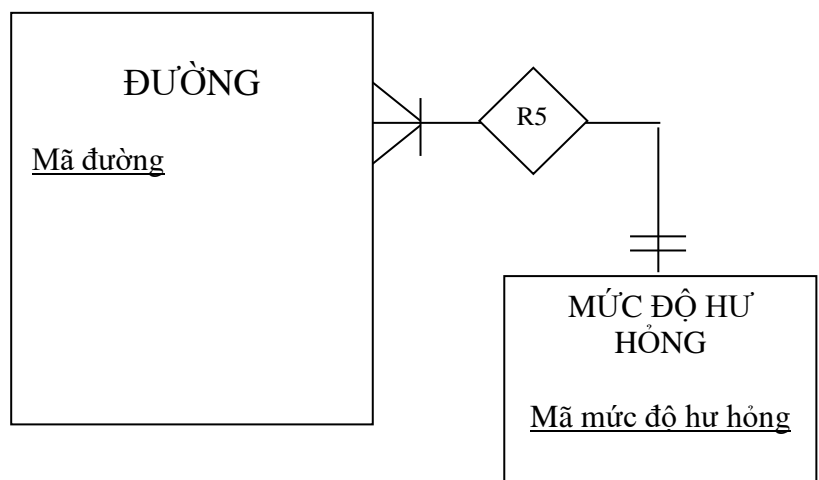
Vì mã đường, mã Tổ chức GT thuộc K, mã đường → mã Tổ chức GT nên:



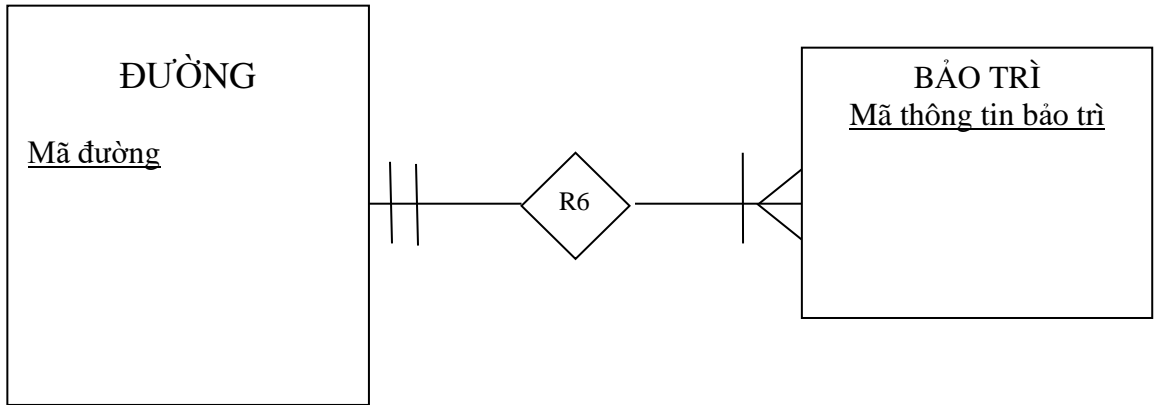
Vì mã đường, mã loại mặt đường thuộc K, mã đường → mã loại mặt đường nên:



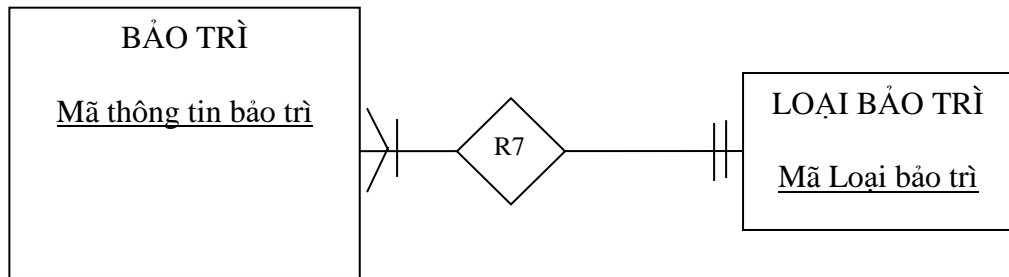
Vì mã đường, mã mức độ hư hỏng thuộc K, mã đường → mã mức độ hư hỏng nên:



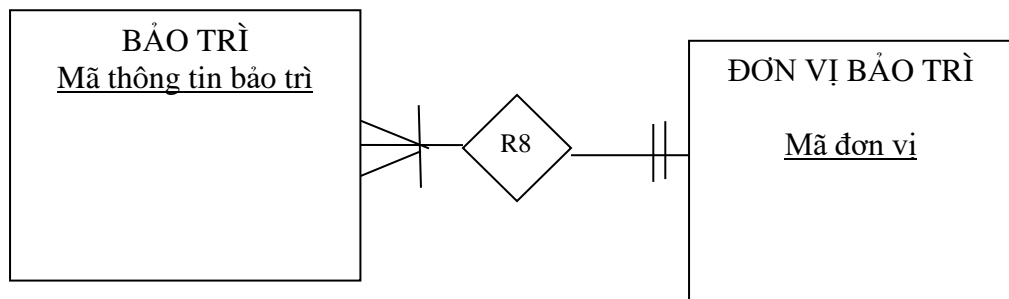
Vì mã thông tin bảo trì, mã đường thuộc K, mã thông tin bảo trì → mã đường
nên:



Vì mã thông tin bảo trì, mã loại bảo trì thuộc K, mã thông tin bảo trì → mã loại
bảo trì nên:



Vì mã thông tin bảo trì, mã đơn vị thuộc K, mã thông tin bảo trì → mã đơn vị
nên:



4. Phân bổ các thuộc tính còn lại:

Tất cả các loại dữ liệu không tập hợp K đều là những thuộc tính thông thường và có thể được phân bổ một cách dễ dàng vào các thực thể hay quan hệ, tùy theo các hàm phụ thuộc.

Phân bổ các thuộc tính còn lại cho các thực thể ta thấy vừa hết:

R1-ĐƯỜNG (Mã đường, Tên đường, Chiều dài, Hệ thống chiếu sáng, Giao cắt ĐS, Hệ thống tín hiệu, Mật độ lưu thông)

R2-QUẬN/HUYỆN (Mã quận/huyện, Tên quận/huyện)

R3-KIỂUĐƯỜNG (Mã Kiểu đường, Kiểu đường)

R4-TỔCHỨC GIAOTHÔNG (Mã Tổ chức GT, Tổ chức GT)

R5-LOẠI MẶT ĐƯỜNG (Mã Loại mặt đường, Loại mặt đường)

R6-MỨC ĐỘ HƯ HỎNG (Mã Mức độ hư hỏng, Mức độ hư hỏng)

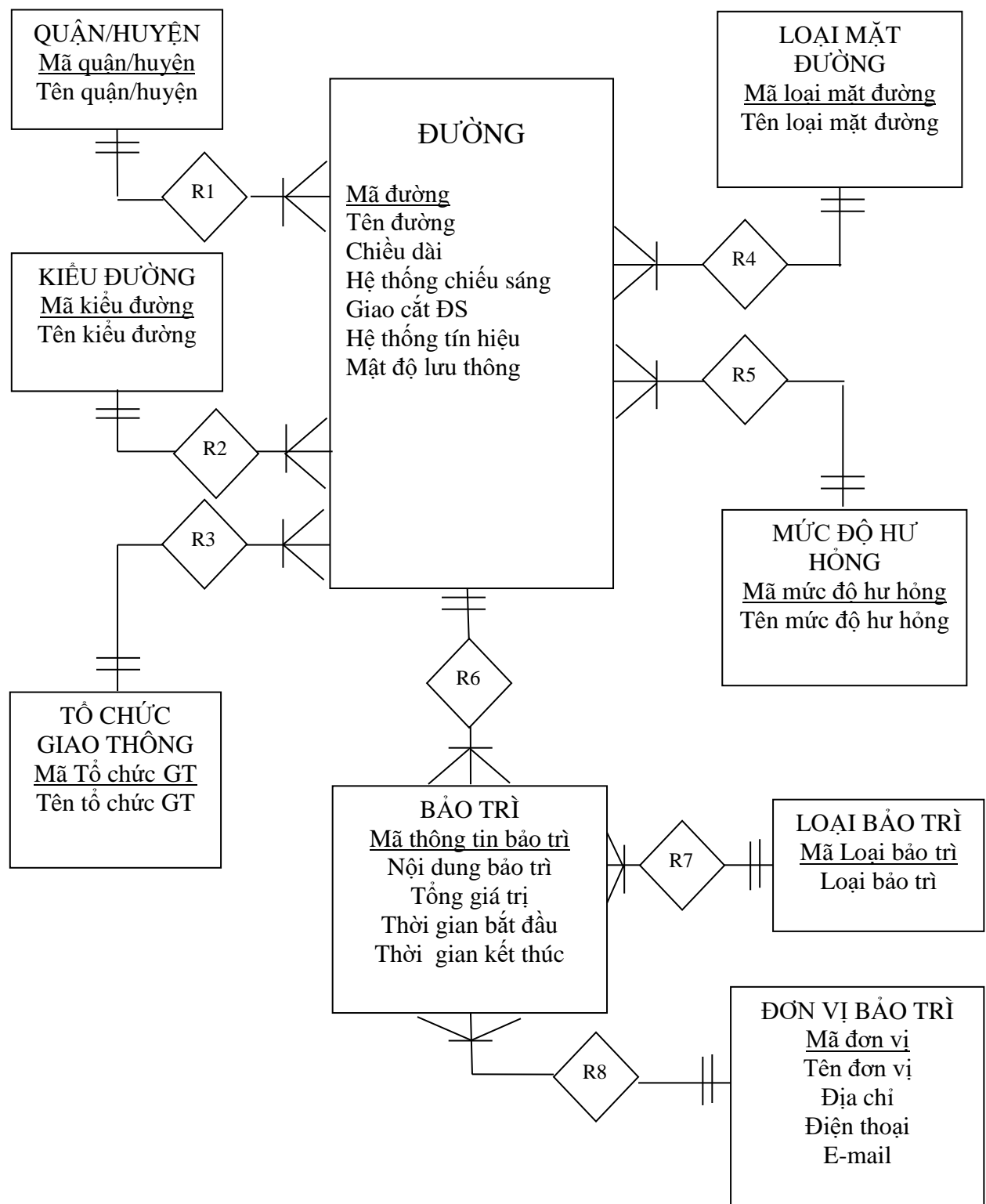
R7-ĐƠN VỊ BẢO TRÌ (Mã đơn vị bảo trì, Tên đơn vị bảo trì, Địa chỉ, Điện thoại, Email)

R8-LOẠI BẢO TRÌ (Mã Loại bảo trì, Loại bảo trì)

R9-BẢO TRÌ (Mã thông tin bảo trì, Nội dung bảo trì, Tổng giá trị, Thời gian bắt đầu, Thời gian kết thúc)

5. Mô hình khái niệm dữ liệu:

Từ các thực thể và quan hệ đã nhận diện, ta có thể vẽ lên một mô hình khái niệm dữ liệu như sau:



Hình 3.1. Mô hình khái niệm dữ liệu bài toán quản lý cung đường giao thông

Chuyển mô hình khái niệm dữ liệu về mô hình quan hệ

Dựa vào 3 quy tắc chuyển :

Quy tắc 1 (quy tắc biến đổi cơ bản):

Mỗi thực thể được chuyển thành một quan hệ trong đó các thuộc tính của thực thể được chuyển thành thuộc tính của quan hệ, định danh của thực thể trở thành khóa của quan hệ.

Quy tắc 2:

Mỗi mối quan hệ 1-N mà không có thuộc tính riêng sẽ không được chuyển thành một quan hệ. Nhưng thực thể tham gia vào mối quan hệ về phía N (phía 1:1 trong mô hình) sẽ đổi mới bằng cách sau khi dùng phép biến đổi cơ bản sẽ nhận thêm khóa của thực thể tham gia vào mối quan hệ ở phía 1 làm khóa liên kết. Còn thực thể tham gia vào mối quan hệ ở phía 1 sẽ biến đổi theo quy tắc 1.

Quy tắc 3:

Mỗi mối quan hệ N-N hoặc mỗi quan hệ có thuộc tính riêng sẽ được chuyển thành một quan hệ mới. Quan hệ mới này có thuộc tính gồm định danh của tất cả các thực thể trong mối quan hệ và các thuộc tính riêng của nó. Khóa của quan hệ được xác định lại sau đó. Các thực thể tham gia vào mối quan hệ đều biến đổi theo quy tắc 1.

Kết quả có hệ lược đồ sau đạt 3NF:

R1-ĐƯỜNG (Mã đường, Tên đường, Mã quận/huyện, Mã Kiểu đường, Mã Tổ chức GT, Mã Loại mặt đường, Mã Mức độ hư hỏng, Chiều dài, Hệ thống chiếu sáng, Giao cắt ĐS, Hệ thống tín hiệu, Mật độ lưu thông)

R2-QUẬN/HUYỆN (Mã quận/huyện, Tên quận/huyện)

R3-KIỂUĐƯỜNG (Mã Kiểu đường, Kiểu đường)

R4-TỔCHỨC GIAOTHÔNG (Mã Tổ chức GT, Tổ chức GT)

R5-LOẠI MẶT ĐƯỜNG (Mã Loại mặt đường, Loại mặt đường)

R6-MỨC ĐỘ HƯ HỎNG (Mã Mức độ hư hỏng, Mức độ hư hỏng)

R7-ĐƠN VỊ BẢO TRÌ (Mã đơn vị bảo trì, Tên đơn vị bảo trì, Địa chỉ, Điện thoại, Email)

R8-LOẠI BẢO TRÌ (Mã Loại bảo trì, Loại bảo trì)

R9-BẢOTRÌ (Mã thông tin bảo trì, Mã đơn vị bảo trì, mã đường, mã loại bảo trì, Nội dung bảo trì, Tổng giá trị, Thời gian bắt đầu, Thời gian kết thúc)

Xây dựng mô hình E_R

Mô hình E_R được biểu diễn dưới dạng một đồ thị, trong đó các nút là các thực thể, còn các cung là các mối quan hệ (các kiểu liên kết các thực thể).

Mô hình được lập như sau:

Mỗi thực thể được biểu diễn bằng một hình chữ nhật có 2 phần: phần trên là tên thực thể (viết in), phần dưới chứa danh sách các thuộc tính, trong đó thuộc tính khóa được đánh dấu (mỗi thực thể chỉ xác định một khóa tối thiểu). Tên thực thể thường là danh từ (chỉ đối tượng).

Một mối quan hệ được biểu diễn thường bằng hình thoi, được nối bằng nét liền tới các thực thể tham gia vào mối quan hệ đó. Trong hình thoi là tên của mối quan hệ cũng được viết in, danh sách các thuộc tính của nó thì được viết thường. Tên của mối quan hệ thường là động từ chủ động hay bị động. Mô hình E_R cuối cùng thường mối quan hệ không còn loại N – N. Trong mối quan hệ nhị nguyên thì ở hai đầu nút các đường nối, sát với thực thể, người ta vẽ đường ba chẽ (còn gọi là đường chân gà) về phía có khóa ngoại (khóa liên kết) thể hiện nhiều, còn phía kia thể hiện một. Bản số trong mỗi đặc tả mối quan hệ giữa 2 thực thể là cặp Max của hai bản số xác định trong đặc tả và được gọi là bản số trực tiếp.

Chú ý:

+ Mối quan hệ có thể không có thuộc tính. Khi có, ta thường gọi là thuộc tính riêng và cũng được viết trong hình thoi song nhớ rằng chỉ viết chữ thường (phân biệt tên của mối quan hệ viết bằng chữ in)

+ Giữa 2 thực thể có thể có nhiều mối quan hệ và chúng cần vẽ riêng rẽ, không chập vào nhau.

Việc chuyển từ hệ lược đồ quan hệ sang sơ đồ E_R được thực hiện theo các bước sau:

Bước 1:

Từ mỗi lược đồ quan hệ, vẽ một hình chữ nhật bao gồm tên lược đồ cùng với các thuộc tính của nó.

Bước 2: Xác định quan hệ (thể hiện bằng đường nối) giữa 2 thực thể (thể hiện bằng hình chữ nhật) bất kỳ.

+*Bước 2.1:* Xác định nhóm thuộc tính chung.

Nếu không có nhóm thuộc tính chung thì kết luận: không có quan hệ

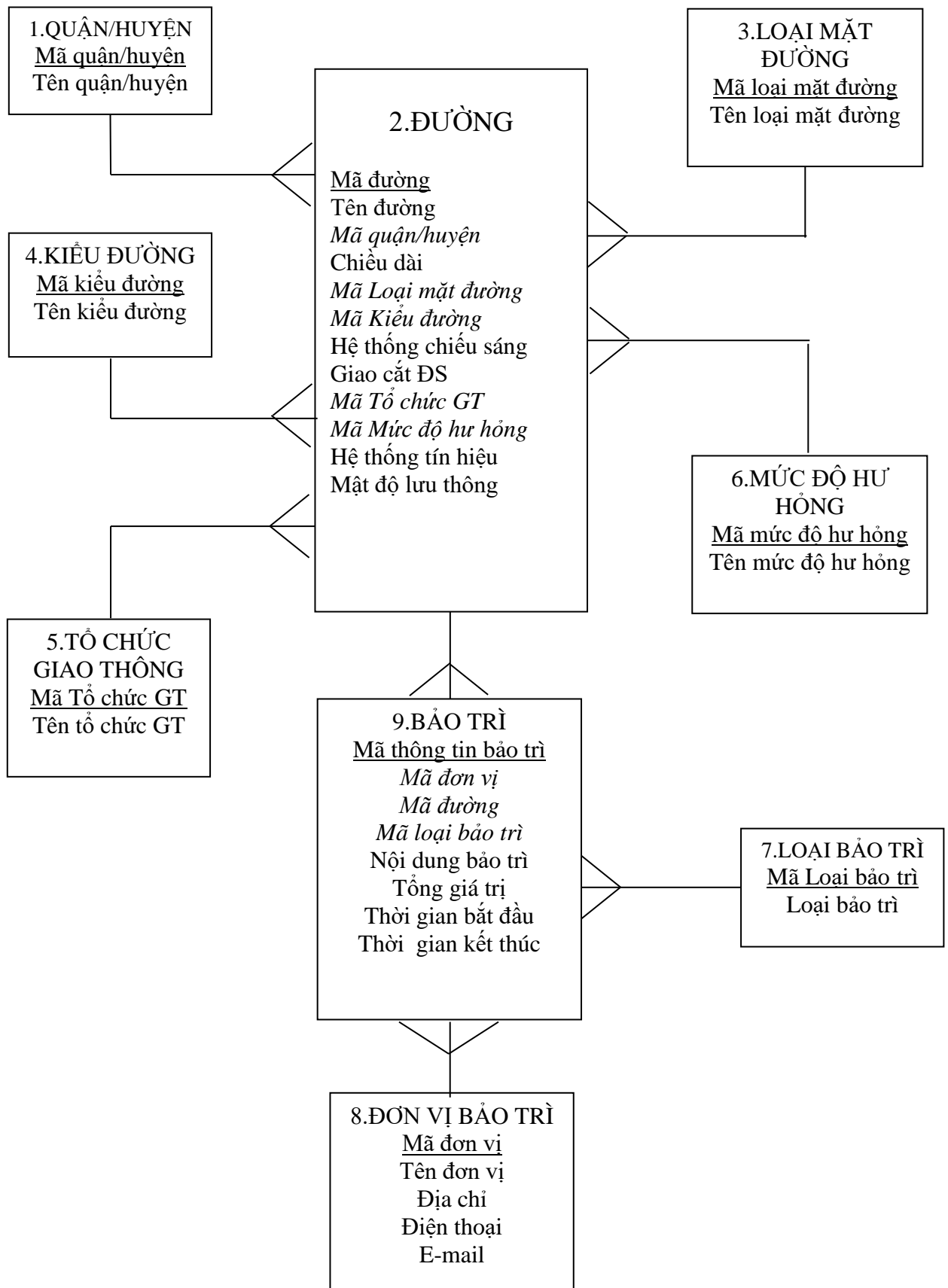
+*Bước 2.2:* Kiểm tra nhóm thuộc tính chung đó có phải là khóa chính của 1 trong 2 thực thể? Nếu không có thì kết luận không có quan hệ.

+*Bước 2.3:* Kết luận tồn tại khóa ngoại: Nhóm thuộc tính chung ở bên thực thể không phải là khóa chính mà sẽ là khóa ngoại.

Bước 3: Vẽ đường nối giữa 2 hình chữ nhật và đường 3 chẽ gắn với thực thể chứa khóa ngoại.

Kết quả chúng ta có *Mô hình Thực thể- Mối quan hệ (Mô hình E_R)*

Hình 3.2. Mô hình Thực thể- Mối quan hệ (Mô hình E_R) bài toán quản lý cung đường giao thông:



Thiết kế CSDL vật lý

Mỗi một hình chữ nhật trong mô hình E_R được lập tương ứng với một bảng từ điển dữ liệu (kiểu dữ liệu C#) để cài đặt trong máy.

1. Quan: Dùng để lưu trữ các thông tin về các quận/huyện trong thành phố HP:

Bảng QUẬN có cấu trúc như sau:

TT	Tên trường	Kiểu dữ liệu, độ rộng	NU LL	Ghi chú
1	MaQuan	Nchar(5)	N	Mã quận (PK)
2	TenQuan	Nvarchar2(50)	N	Tên quận

Bảng 3.1. Cấu trúc bảng QUAN

2. Duong: Mô tả chi tiết các thông tin cần thiết về các cung đường

Bảng DUONG có cấu trúc như sau:

TT	Tên trường	Kiểu dữ liệu, độ rộng	NULL	Ghi chú
1	MaDuong	Nchar(5)	N	Mã đường (PK)
2	TenDuong	Nvarchar(50)	N	Tên đường
3	MaQuan	Integer	N	Mã quận (FK)
4	ChieuDai	Integer	N	Chiều dài
5	MaLoaiMatDuong	Integer	N	Mã loại mặt đường(FK)
6	MaKieuDuong	Integer	N	Mã kiểu đường FK)
7	HeThongChieuSang	Bit	N	Hệ thống chiếu sáng
8	GiaoCatVoiDuongSat	Bit	N	Giao cắt với đường sắt
9	MaToChucGiaoThong	Integer	N	Mã tổ chức giao thông (FK)
10	MaMucDoHuHong	Integer	N	Mã mức độ hư hỏng (FK)
11	HeThongTinHieu	Bit	N	Hệ thống tín hiệu (xác định đường có hệ thống chiếu hay không)
12	MatDoLuuThong	Integer	N	Mật độ lưu thông (xác định mật độ xe lưu thông)

Bảng 3.2. Cấu trúc bảng DUONG

3. LoaiMatDuong: Phân loại mặt đường có các loại mặt đường sau: Bê tông nhựa, Bê tông xi măng(BTXM), Đá dăm trộn nhựa, Thảm nhập nhựa; láng nhựa 2, 3 lớp, Đá dăm tiêu chuẩn, cấp phối đá dăm, Cấp phối thiên nhiên)

Bảng LOAIMATDUONG có cấu trúc như sau:

TT	Tên trường	Kiểu dữ liệu, độ rộng	NULL	Ghi chú
1	MaLoaiMatDuong	Integer	N	Mã loại mặt đường (PK)
2	TenLoaiMatDuong	Nvarchar(50)	N	Tên loại mặt đường

Bảng 3.3. Cấu trúc bảng LOAIMATDUONG

4. KieuDuong: Phân loại kiểu đường, gồm: Đường liên Phường, Đường nội phường. Các thuộc tính:

Bảng KIEUDUONG có cấu trúc như sau:

Tt	Tên trường	Kiểu dữ liệu,độ rộng	NULL	Ghi chú
1	MaKieuDuong	Integer	N	Mã kiểu đường (PK)
2	TenKieuDuong	Nvarchar(50)	N	Tên kiểu đường

Bảng 3.4. Cấu trúc bảng KIEUDUONG

5. ToChucGiaoThong: Xác định cách tổ chức đường giao thông (đường 1 chiều, đường 2 chiều có giải phân cách cứng, đường 2 chiều không có giải phân cách). Các thuộc tính:

Bảng TOCHUCGIAOTHONG có cấu trúc như sau:

TT	Tên trường	Kiểu dữ liệu, Độ rộng	NULL	Ghi chú
1	MaToChucGiaoThong	Integer	N	Mã tổ chức giao thông (PK)
2	TenToChucGiaoThong	Nvarchar(50)	N	Tên tổ chức giao thông

Bảng 3.5. Cấu trúc bảng TOCHUCGIAOTHONG

6. MucDoHuHong: Xác định mức độ hư hỏng của đường:

Xếp loại 1 và 2: ít hoặc không có vết nứt hay biến dạng, tình trạng mặt đường tốt, không cần sửa chữa hoặc công tác sửa chữa có thể trì hoãn được.

Xếp loại 3 và 4: nứt nhưng không có hoặc ít biến dạng; tình trạng mặt đường không tốt cần tiến hành sửa chữa không cần nghiên cứu gì dung.

Xếp loại 5, 6, 7: nứt và biến dạng trên diện rộng; tình trạng lớp mặt đường rất xấu và tòi tệ cần sửa chữa lớn hoặc tiến hành trải thảm.

Bảng MUCDOHUHONG có cấu trúc như sau:

TT	Tên trường	Kiểu dữ liệu, độ rộng	NULL	Ghi chú
1	MaMucDoHuHong	Integer	N	Mã mức độ hư hỏng (PK)
2	TenMucDoHuHong	Nvarchar(50)	N	Tên mức độ hư hỏng

Bảng 3.6. Cấu trúc bảng MUCDOHUHONG

7. LoaiBaoTri: Xác định loại bảo trì (Sửa chữa thường xuyên, sửa chữa định kỳ, sửa chữa đột suất). Các thuộc tính:

Bảng LOAIBAOTRI có cấu trúc như sau:

TT	Tên trường	Kiểu dữ liệu, độ rộng	NULL	Ghi chú
1	MaLoaiBaoTri	Integer	N	Mã loại bảo trì (PK)
2	TenLoaiBaoTri	Nvarchar2(50)	N	Tên loại bảo trì

Bảng 3.7. Cấu trúc bảng LOAIBAOTRI

8. DonViThiCong: Xác định đơn vị thi công. Các thuộc tính:

Bảng DONVITHICONG sau:

TT	Tên trường	Kiểu dữ liệu, độ rộng	NULL	Ghi chú
1	MaDonViThiCong	Integer	N	Mã đơn vị thi công (PK)
2	TenViThiCong	Integer	N	Tên đơn vị thi công
3	DiaChi	Nvarchar(100)	N	Địa chỉ
4	DiệnThoai	Nvarchar(12)	Y	Điện thoại
5	Email	Nvarchar(100)	Y	Email

Bảng 3.8. Cấu trúc bảng DONVITHICONG

9. ThôngTinBảoTri: Xác định thông tin của việc bảo trì đường bộ (Mã cung đường tiến hành bảo trì, Nội dung các công việc trong 1 đợt bảo trì, Đơn vị thi công, Tổng Giá trị của đợt bảo trì, Thời gian bắt đầu bảo trì, Thời gian kết thúc bảo trì, Mã loại bảo trì). Các thuộc tính:

Bảng THONGTINBAOTRI có cấu trúc như sau:

TT	Tên trường	Kiểu dữ liệu, độ rộng	NULL	Ghi chú
1	MaThongTinBaoTri	Integer	N	Mã thông tin bảo trì (PK)
2	MaDuong	Integer	N	Mã đường (FK)
3	MaLoaiBaoTri	Intege	N	Mã loại bảo trì (FK)
4	NoiDungBaoTri	Nvarchar(500)	N	Nội dung bảo trì
5	MaDonViThiCong	Intege	N	Mã đơn vị thi công (FK)
6	TongGiaTri	Float	N	Tổng giá trị
7	NgayBatDau	Date	N	Ngày bắt đầu
8	NgayKetThuc	Date	N	Ngày kết thúc

Bảng 3.9. Cấu trúc bảng THONGTINBAOTRI

3.4.2. Giới thiệu chương trình

3.4.2.1. Mục tiêu hệ thống

Hệ thống chương trình sau khi xây dựng phải đạt được các yêu cầu sau:

- Đảm bảo thông tin nhanh chóng, chính xác, kịp thời.
- Thao tác đơn giản với người sử dụng.
- CSDL tại các máy chủ phải đảm bảo an toàn, bảo mật với hệ thống sao lưu tốt, tính sẵn dụng cao.

3.4.2.2. Cấu trúc chương trình

Chương trình được viết bằng ngôn ngữ lập trình C#.

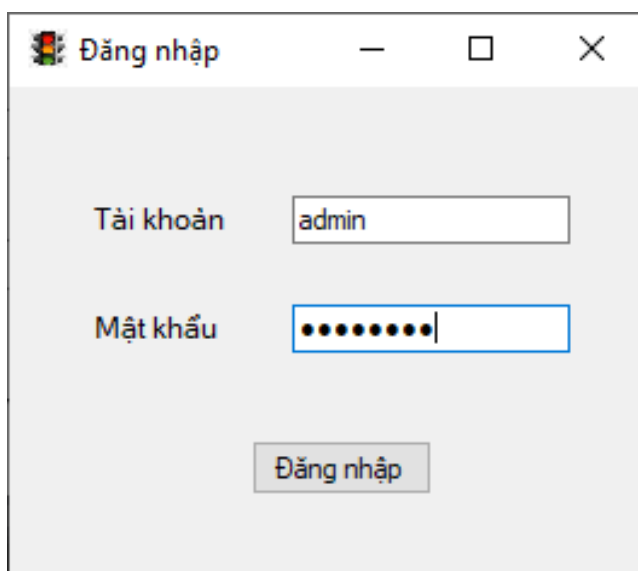
Hệ quản trị cơ sở dữ liệu được sử dụng: SQL SERVER 2019

Chương trình được xây dựng gồm các mô-đun chính sau đây.

- Mô-đun quản trị và phân quyền người dùng: Mô-đun này cho phép quản lý, cập nhật thông tin người dùng: dung mới, loại bỏ và thay đổi mật khẩu người dùng.
- Mô-đun quản lý các bảng danh mục: Mô-đun này cho phép quản lý, cập nhật các danh mục được sử dụng trong chương trình: dung mới, loại bỏ và thay đổi các danh mục.
- Mô-đun quản lý cung đường: cho phép quản lý các thông tin liên quan đến các cung đường.
- Mô-đun quản lý thông tin bảo trì: cho phép quản lý các thông tin liên quan đến thông tin về bảo trì các cung đường.
- Mô-đun tìm kiếm: cho phép tìm kiếm thông tin các cung đường, thông tin bảo trì của các cung đường.

3.4.2.3. Một số giao diện chính

Giao diện form đăng nhập: Sau khi khởi động phần mềm, người dùng sẽ phải thực hiện bước đăng nhập bằng cách nhập tài khoản và mật khẩu, sau đó nhấn vào nút “Đăng nhập”.

A screenshot of a Windows-style login window titled "Đăng nhập". The window has a standard title bar with minimize, maximize, and close buttons. Inside the window, there are two input fields. The first is labeled "Tài khoản" and contains the text "admin". The second is labeled "Mật khẩu" and contains ten dots, indicating a masked password. Below these fields is a button labeled "Đăng nhập".

Hình 3.3. Giao diện form đăng nhập

Giao diện form quản lý thông tin cung đường: Ở giao diện này, toàn bộ thông tin về các cung đường sẽ được hiện ra, người dùng muốn xem chi tiết ở đường nào sẽ click chuột vào dòng có tên đường đó, mọi thông tin về cung đường người dùng chọn sẽ hiển thị ở phía trên. Nếu muốn sửa thông tin nào, người dùng chỉ cần thay đổi thông tin đã hiển thị ở phía trên rồi bấm “Sửa” thì phần mềm sẽ lưu lại những thông tin đã được thay đổi bởi người dùng. Nếu muốn thêm cung đường mới, người dùng bấm “Thêm”, phần mềm sẽ tự động xóa những ô text (Mã đường, Tên đường, Chiều dài, Mật độ lưu thông), những ô còn lại được giữ nguyên, người dùng sẽ tự nhập những thông tin trong dòng text và chọn những thông tin phù hợp rồi bấm “Lưu”, sau khi bấm “Lưu”, phần mềm sẽ tự động lưu vào trong cơ sở dữ liệu. Nếu bấm “Xóa”, phần mềm sẽ xóa đi dòng mà người dùng đã chọn trước đó. Nếu bấm “Thoát”, phần giao diện sẽ tự động thoát.

Phần mềm quản lý cung đường thành phố Hải Phòng

THÔNG TIN CUNG ĐƯỜNG

Mã đường: Quận: Hệ thống chiếu sáng: Có Không

Tên đường: Tổ chức giao thông: Giao cắt với đường sắt: Có Không

Chiều dài: Loại mặt đường: Hệ thống tín hiệu: Có Không

Kiểu đường: Mức độ hư hỏng: Mật độ lưu thông:

	Mã đường	Tên đường	Mã quận	Chiều dài	Mã loại mặt đường	Mã kiểu đường	Hệ thống chiếu sáng	Giao cắt với đường sắt	Mã tổ chức giao thông	Mã mức độ hư hỏng	Hệ t tín h
▶	DD001	Hai Bà Trưng	1	12	4	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	2	1	
	DD002	Ngô Gia Tự	2	5	1	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	2	2	
	DD003	Nguyễn Bình Khiêm	3	8	5	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	2	1	
*							<input type="checkbox"/>	<input type="checkbox"/>			

Hình 3.4. Giao diện form quản lý thông tin cung đường

Giao diện form quản lý thông tin bảo trì: Ở giao diện này, toàn bộ thông tin về bảo trì đường sẽ được hiện ra, người dùng muốn xem chi tiết thông tin nào sẽ click chuột vào dòng thông tin đó, mọi thông tin về thông tin mà người dùng chọn sẽ hiển thị ở phía trên. Nếu muốn sửa thông tin nào, người dùng chỉ cần thay đổi thông tin đã hiển thị ở phía trên rồi bấm “Sửa” thì phần mềm sẽ lưu lại những thông tin đã được thay đổi bởi người dùng. Nếu muốn thêm thông tin bảo trì mới, người dùng bấm “Thêm”, phần mềm sẽ tự động xóa những ô text (Mã thông tin bảo trì, Nội dung bảo trì, Tổng giá trị), những ô còn lại được giữ nguyên, người dùng sẽ tự nhập những thông tin trong dòng text và chọn những thông tin phù hợp rồi bấm “Lưu”, sau khi bấm “Lưu”, phần mềm sẽ tự động lưu vào trong cơ sở dữ liệu. Nếu bấm “Xóa”, phần mềm sẽ xóa đi dòng mà người dùng đã chọn trước đó. Nếu bấm “Thoát”, phần giao diện sẽ tự động thoát.

Phân mềm quản lý cung đường thành phố Hải Phòng

THÔNG TIN BẢO TRÌ

Mã thông tin bảo trì: Đơn vị thi công: Ngày bắt đầu:

Tên đường: Nội dung bảo trì: Ngày kết thúc:

Loại bảo trì: Tổng giá trị:

	Mã thông tin bảo trì	Mã đường	Mã loại bảo trì	Nội dung bảo trì	Mã đơn vị thi công	Tổng giá trị	Ngày bắt đầu	Ngày kết thúc
▶	1	DD001	2	Vá mặt đường	5	10000000000	12-Oct-20	12-Oct-22
	2	DD002	3	Sửa lại mặt đường	4	20000000000	06-Dec-19	02-May-21
*								

Hình 3.5. Giao diện form quản lý thông tin bảo trì

Giao diện form quản lý thông tin người dùng: Ở giao diện này, toàn bộ thông tin người dùng sẽ được hiện ra, người dùng muốn xem chi tiết thông tin nào sẽ click chuột vào dòng thông tin đó, mọi thông tin về thông tin mà người dùng chọn sẽ hiển thị ở phía trên. Nếu muốn sửa thông tin nào, người dùng chỉ cần thay đổi thông tin đã hiển thị ở phía trên rồi bấm “Sửa” thì phần mềm sẽ lưu lại những thông tin đã được thay đổi bởi người dùng. Nếu muốn thêm thông tin người dùng mới, người dùng bấm “Thêm”, phần mềm sẽ tự động xóa những ô text (Họ tên, Số điện thoại, Địa chỉ, Địa chỉ email), những ô còn lại được giữ nguyên, người dùng sẽ tự nhập những thông tin trong dòng text và chọn những thông tin phù hợp rồi bấm “Lưu”, sau khi bấm “Lưu”, phần mềm sẽ tự động lưu vào trong cơ sở dữ liệu. Nếu bấm “Xóa”, phần mềm sẽ xóa đi dòng mà người dùng đã chọn trước đó. Nếu bấm “Thoát”, phần giao diện sẽ tự động thoát.

Phần mềm quản lý cung đường thành phố Hải Phòng

QUẢN LÝ THÔNG TIN NGƯỜI DÙNG

Họ tên Số điện thoại Giới tính Nam Nữ

Địa chỉ Địa chỉ email

	Tài khoản	Mật khẩu	Tên người dùng	Email	Số điện thoại	Giới tính	Địa chỉ
▶	admin	admin123	Nguyễn Văn A	anvhpu@gmail...	04123812923	<input checked="" type="checkbox"/>	Số 15 Đường D...
	khach	khach0123	Nguyễn Thị T	tttttt@gmail.com	09123039400	<input type="checkbox"/>	16B Ngô Gia Tự
	mod	mod2021	Trần Thị Z	tranz2021@gm...	07439324822	<input type="checkbox"/>	Số 10A Ngõ 20 ...
*						<input type="checkbox"/>	

Hình 3.6. Giao diện form quản lý thông tin người dùng

Giao diện tìm kiếm cung đường: Ở giao diện này, người dùng có thể tra cứu thông tin của các cung đường như tên đường, kiểu đường, tổ chức giao thông,...Người dùng tra cứu thông tin có thể bằng cách nhập tên con đường vào ô “Tên đường”, có thể chọn từ một trong các combo box (Tên quận, Kiểu đường, Mức độ hư hỏng và Tổ chức giao thông) phần mềm sẽ đưa ra những kết quả tương ứng

Phần mềm quản lý cung đường thành phố Hải Phòng

TÌM KIẾM CUNG ĐƯỜNG

Quận

Tên đường Mức độ hư hỏng

Kiểu đường Tổ chức giao thông

	Mã đường	Tên đường	Tên quận	Chiều dài	Loại mặt đường	Kiểu đường	Hệ thống chiếu sáng	Giao cắt với đường sắt	Tổ chức giao thông	Mức độ hư hỏng	Hệ thống tín hiệu	Mật độ lưu thông
▶	DD001	Hai Bà Trưng	Hồng Bàng	12	Láng nhựa 2...	Đường liên q...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Đường 2 chi...	Loại 1 và 2	<input checked="" type="checkbox"/>	124
	DD002	Ngô Gia Tự	Lê Chân	5	Bê tông nhựa	Đường nội q...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Đường 2 chi...	Loại 3 và 4	<input checked="" type="checkbox"/>	15
	DD003	Nguyễn Bình...	Ngô Quyền	8	Láng nhựa 3...	Đường liên q...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Đường 2 chi...	Loại 1 và 2	<input checked="" type="checkbox"/>	200
*							<input type="checkbox"/>	<input type="checkbox"/>			<input type="checkbox"/>	

Hình 3.7. Giao diện form tìm kiếm thông tin cung đường

KẾT LUẬN

1. Kết quả đạt được của đồ án

Trong thời gian thực hiện đồ án, em đã tìm hiểu về phương pháp Blanpre, vận dụng phương pháp đó để có thể thiết kế được cơ sở dữ liệu cho phần mềm quản lý các cung đường trong thành phố Hải Phòng.

Phần mềm đã thực hiện được các chức năng: tra cứu thông tin các cung đường trong địa bàn thành phố Hải Phòng, thông tin bảo trì cung đường và quản lý thông tin người dùng.

Thông qua phần mềm, người dùng sẽ dễ dàng hơn trong việc tra cứu và nắm bắt được thông tin các cung đường trong thành phố, dễ dàng nắm bắt thông tin để có thể làm báo cáo.

2. Những hạn chế

Phần mềm vẫn còn hạn chế về giao diện vì mới phát triển được ở mức đơn giản.

Trong khoảng thời gian ngắn để thực hiện đề tài, em mới chỉ xây dựng được phần mềm với các chức năng cơ bản để tra cứu thông tin cung đường.

Vẫn còn một số chức năng cần hoàn thiện.

3. Hướng phát triển

Chỉnh sửa các giao diện nhập liệu, xử lý, báo cáo cho đẹp hơn và phù hợp với yêu cầu quản lý.

Hoàn thiện những chức năng đã có.

Tạo menu có thêm chức năng mới: chức năng tìm kiếm thông tin của đơn vị bảo trì, tìm kiếm thông tin người dùng, cho người dùng mới đăng ký tài khoản và khai báo thông tin, phân quyền cho người dùng.

Thiết kế cơ sở dữ liệu ở mức chi tiết hơn đến mức phường/xã.

Phát triển những giao diện có chức năng báo cáo dưới dạng bảng hay đồ thị.

TÀI LIỆU THAM KHẢO

[1] Nguyễn Xuân Huy (2011), *Bài tập CSDL*, Tái bản lần thứ 5, Trường Đại Học Duy Tân, NXB Thông tin và Truyền thông.

[2] Vũ Xuân Oanh (2020), *Bài toán quản lý thông tin các cung đường bộ trên địa bàn TP. Lạng Sơn*, Đại học Thông tin và truyền thông Thái Nguyên.

[3] Lê Văn Phùng (2014), *Kỹ thuật phân tích và thiết kế hệ thống thông tin hướng cấu trúc*, Tái bản lần 3, NXB Thông tin và Truyền thông.

[4] Lê Văn Phùng (2018), *Cơ sở dữ liệu quan hệ và công nghệ phân tích thiết kế*, Tái bản lần 1, NXB Thông tin và Truyền thông.

[5] Vũ Đức Thi (1997), *Cơ sở dữ liệu - Kiến thức và thực hành*, NXB Thống kê.

[6] Nguyễn Đăng Ty, Đỗ Phúc (2003), *Giáo trình cơ sở dữ liệu*, Đại học quốc gia TP. Hồ Chí Minh.

[7] <https://www.w3schools.com/sql/default.asp>

[8] <https://haiphong.gov.vn/>

[9] <https://support.microsoft.com/vi-vn/office/kiến-thức-cơ-bản-về-thiết-kế-cơ-sở-dữ-liệu-eb2159cf-1e30-401a-8084-bd4f9c9ca1f5>