

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-----o0o-----



ISO 9001:2015

ĐỒ ÁN TỐT NGHIỆP

NGÀNH CÔNG NGHỆ THÔNG TIN

HẢI PHÒNG 2019

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-----o0o-----

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ CHO VAY VỐN
CỦA QUỸ VAY VỐN CỦA LIÊN ĐOÀN LAO ĐỘNG
THÀNH PHỐ**

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

NGÀNH: CÔNG NGHỆ THÔNG TIN

HẢI PHÒNG - 2019

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-----o0o-----

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ CHO VAY VỐN
CỦA QUỸ VAY VỐN
CỦA LIÊN ĐOÀN LAO ĐỘNG THÀNH PHỐ**

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

NGÀNH: CÔNG NGHỆ THÔNG TIN

Sinh viên thực hiện: Nguyễn Văn Hiếu

Giáo viên hướng dẫn: TS. Đỗ Văn Chiểu

Mã số sinh viên: 1512111004

Hải Phòng - 2019

NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP

Sinh viên: Nguyễn Văn Hiếu

Mã số: 1512111004

Lớp: CT1901C

Ngành: Công nghệ Thông tin

Tên đề tài: Xây dựng ứng dụng quản lý cho vay vốn của quỹ vay vốn của liên đoàn lao động thành phố.

CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Người hướng dẫn thứ nhất:

Họ và tên: Đỗ Văn Chiêu

Học hàm, học vị: Tiến sĩ

Cơ quan công tác: Trường đại học Dân Lập Hải Phòng

Nội dung hướng dẫn:

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

Người hướng dẫn thứ hai:

Họ và tên:

Học hàm, học vị.....

Cơ quan công tác:

Nội dung hướng dẫn:

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

Đề tài tốt nghiệp được giao ngày 18 tháng 3 năm 2019

Yêu cầu phải hoàn thành trước ngày 07 tháng 6 năm 2019

Đã nhận nhiệm vụ: Đ.T.T.N

Sinh viên

Đã nhận nhiệm vụ: Đ.T.T.N

Cán bộ hướng dẫn Đ.T.T.N

Hải Phòng, ngày....tháng....năm 2019

HIỆU TRƯỞNG

GS.TS.NGŨT Trần Hữu Nghị

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN TỐT NGHIỆP

Họ và tên giảng viên:

Đơn vị công tác:

Họ và tên sinh viên: Ngành:

Nội dung hướng dẫn:

.....

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

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

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

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

3. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Đạt Không đạt Điểm:.....

Hải Phòng, ngày tháng 06 năm 2019

Giảng viên hướng dẫn

(Ký và ghi rõ họ tên)

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN CHẤM PHẢN BIỆN

Họ và tên giảng viên:

Đơn vị công tác:

Họ và tên sinh viên: Ngành:

Đề tài tốt nghiệp:

.....

1. Phần nhận xét của giảng viên chấm phản biện

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

2. Những mặt còn hạn chế

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

3. Ý kiến của giảng viên chấm phản biện

Được bảo vệ Không được bảo vệ Điểm:.....
vệ

Hải Phòng, ngày tháng 06 năm 2019

Giảng viên chấm phản biện

(Ký và ghi rõ họ tên)

LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành nhất đến quý thầy cô Trường Đại Học Dân Lập Hải Phòng, những người đã dìu dắt em tận tình, đã truyền đạt cho em những kiến thức và bài học quý báu trong suốt thời gian em theo học tại trường.

Em xin trân trọng gửi lời cảm ơn đến tất cả các thầy cô trong khoa Công Nghệ Thông Tin, đặc biệt là thầy giáo TS. Đỗ Văn Chiêu, thầy đã tận tình hướng dẫn và giúp đỡ em trong suốt quá trình làm tốt nghiệp. Với sự chỉ bảo của thầy, em đã có những định hướng tốt trong việc triển khai và thực hiện các yêu cầu trong quá trình làm đồ án tốt nghiệp.

Em xin cảm ơn những người thân và gia đình đã quan tâm, động viên và luôn tạo cho em những điều kiện tốt nhất trong suốt quá trình học tập và làm tốt nghiệp.

Ngoài ra, em cũng xin gửi lời cảm ơn tới tất cả bạn bè, đặc biệt là các bạn trong lớp CT1901 đã luôn gắn bó, cùng học tập và giúp đỡ em trong những năm qua và trong suốt quá trình thực hiện đồ án này.

Em xin chân thành cảm ơn!

Hải Phòng, ngày tháng 6 năm 2019

Sinh viên

Nguyễn Văn Hiếu

MỤC LỤC

MỞ ĐẦU	8
CHƯƠNG 1 TỔNG QUAN VỀ PHP	9
1.1 Lịch sử phát triển.....	9
1.1.1 PHP	9
1.1.2 PHP3	9
1.1.3 PHP4	9
1.1.4 PHP5	10
1.1.5 PHP6	10
1.2 Cấu trúc cơ bản của PHP	11
1.2.1 Các cấu trúc cơ bản	11
1.2.2 Xuất giá trị ra trình duyệt	12
1.2.3 Biến, hằng, chuỗi và các kiểu dữ liệu	12
1.2.4 Các phương thức được sử dụng trong lập trình PHP	14
1.2.5 Cookie và Session trong PHP	15
1.2.6 Cookie và Session trong PHP	17
1.3 MySQL	18
1.3.1 Giới thiệu cơ sở dữ liệu:	18
1.3.2 Mục đích sử dụng cơ sở dữ liệu:	19
1.3.3 Các kiểu dữ liệu trong cơ sở dữ liệu MySQL.....	20
1.3.4 Các thao tác cập nhật dữ liệu	22
1.3.5 Các hàm thông dụng trong MySQL	23
1.4 Phân tích thiết kế hệ thống hướng cấu trúc	24
1.4.1 Quá trình phát triển của một hệ thống thông tin.....	24
1.4.2 Tiếp cận định hướng cấu trúc	28
1.5 Thiết kế cơ sở dữ liệu quan hệ	30
1.5.1 Mô hình liên kết thực thể ER.....	30
CHƯƠNG 2 MÔ TẢ BÀI TOÁN	36
2.1 Phát biểu bài toán quản lý cho vay dành cho công nhân viên chức nghèo của liên đoàn lao động thành phố Hải Phòng.....	36
2.2 Sơ đồ tiến trình nghiệp vụ	37

2.2.1 Sơ đồ tiến trình nghiệp vụ thu vốn gốc, lãi	37
2.2.2 Sơ đồ tiến trình nghiệp vụ thanh lý hợp đồng và trả TKBB	39
2.2.3 Sơ đồ tiến trình nghiệp vụ lập báo cáo tổng hợp	41
CHƯƠNG 3 PHÂN TÍCH THIẾT KẾ HỆ THỐNG	43
3.1 Biểu đồ nghiệp vụ	43
3.1.1 Biểu đồ ngữ cảnh.....	43
3.1.2 Sơ đồ phân rã chức năng	45
3.1.3 Danh sách các hồ sơ cần sử dụng	46
3.1.4 Ma trận thực thể chức năng	46
3.2 Sơ đồ luồng dữ liệu	47
3.2.1 Sơ đồ luồng dữ liệu mức 0.....	47
3.2.2 Sơ đồ luồng dữ liệu mức 1	48
3.3 Thiết kế các bảng cơ sở dữ liệu	50
3.3.1 Mô hình liên kết thực thể (ER)	50
3.3.2 Mô hình quan hệ.....	52
3.3.3 Các bảng dữ liệu vật lý	54
CHƯƠNG 4 CÀI ĐẶT CHƯƠNG TRÌNH.....	57
4.1 Giới thiệu về hệ thống chương trình	57
4.1.1 Môi trường cài đặt.....	57
4.1.2 Các hệ thống con.....	57
4.1.3 Các chức năng chính của mỗi hệ thống.....	58
4.2 Giao diện website	58
4.2.1 Giao diện đăng nhập hệ thống	58
4.2.2 Các bảng tổng hợp.....	59
KẾT LUẬN.....	62
TÀI LIỆU THAM KHẢO	63

MỤC LỤC HÌNH ẢNH

Hình 1: xuất giá trị	12
Hình 2: Biến trong PHP	13
Hình 3: Hằng trong PHP	13
Hình 4: Chuỗi trong PHP	14
Hình 5: Kiểu dữ liệu trong PHP	14
Hình 6: Phương thức POST trong PHP	15
Hình 7: Cú pháp của session	17
Hình 8: Cú pháp hủy bỏ giá trị session	17
Hình 9: Cú pháp hàm tự định nghĩa	17
Hình 10: Cú pháp hàm tự định nghĩa	18
Hình 11: Hàm tự định nghĩa với giá trị trả về	18
Hình 12: Câu lệnh SELECT	22
Hình 13: Câu lệnh INSERT	22
Hình 14: Câu lệnh UPDATE	23
Hình 15: Câu lệnh DELETE	23
Hình 16: Cấu trúc hệ thống định hướng cấu trúc	29
Hình 17: Các ký hiệu trong lược đồ E-R	33
Hình 18: Ví dụ lược đồ E-R	34
Hình 19: Các kiểu liên kết trong lược đồ E-R	35
Hình 20: Ví dụ liên kết 1 nhiều	35
Hình 21: Sơ đồ tiến trình nghiệp vụ thu vốn gốc, lãi	37
Hình 22: Sơ đồ tiến trình nghiệp vụ thanh lý hợp đồng và trả TKBB	39

Hình 23: Sơ đồ tiến trình nghiệp vụ lập báo cáo tổng hợp	41
Hình 24: Biểu đồ ngữ cảnh.....	43
Hình 25: Sơ đồ phân rã chức năng.....	45
Hình 26: Ma trận thực thể chức năng	46
Hình 27: Sơ đồ luồng dữ liệu mức 0.....	47
Hình 28: Sơ đồ luồng dữ liệu mức 1 tiến trình “Thu hàng tháng”.....	48
Hình 29: Sơ đồ luồng dữ liệu mức 1 tiến trình “Báo cáo”.....	49
Hình 30: Liên kết NHÂN VIÊN thu vốn gốc KHÁCH HÀNG.....	50
Hình 31: Liên kết NHÂN VIÊN thanh lý KHÁCH HÀNG	51
Hình 32: Mô hình thực thể (ER).....	52
Hình 33: Mô hình quan hệ.....	53
Hình 34: Giao diện đăng nhập hệ thống	58
Hình 35: Giao diện trang quản lý	59
Hình 36: Bảng theo dõi nợ tiền của người vay.....	59
Hình 37: Phiếu thu	59
Hình 38: Bảng theo dõi thu vốn, lãi và tiết kiệm bắt buộc của người vay vốn ..	60
Hình 39: Bảng tính thu vốn, lãi vay và tiết kiệm bắt buộc	60
Hình 40: Bảng theo dõi thu vốn và lãi của đơn vị vay vốn.	61
Hình 41: Bảng thống kê thu chi - tổng hợp.....	61

MỞ ĐẦU

Hiện nay Công nghệ thông tin vô cùng phát triển. Internet dần trở thành thứ không thể thiếu trong cuộc sống. Việc quản lý thông tin trên mạng dần phổ biến trong những năm gần đây. Đây là một lợi thế cho các doanh nghiệp hay các cơ quan có thể tiếp cận khách hàng và quản lý dữ liệu một cách nhanh chóng bằng một cái website để giám đốc hoặc người quản lý có thể biết thông tin hay quản lý thông tin. Và nhu cầu thiết yếu là khi giám đốc đang đi công tác muốn xem hay quản lý thì làm sao họ có thể biết được thông tin? Từ thực tế đó chúng ta có thể nhìn thấy việc xây dựng hệ thống quản lý thông tin trên nền tảng website là rất quan trọng và cần thiết. Vì thế nên em chọn đề tài: “Xây dựng quản lý cho vay vốn của quỹ vay vốn của liên đoàn lao động thành phố” với mục đích là nghiên cứu tìm hiểu và xây dựng quản lý cho vay vốn của quỹ vay vốn của liên đoàn lao động thành phố trên website. Giúp các cơ quan, đoàn thể có thể dễ dàng quản lý. Tối ưu chi phí, công sức và quản lý dễ dàng tiện lợi.

Trong đề tài này em được giao nhiệm vụ quản lý cho vay vốn của quỹ vay vốn của liên đoàn lao động thành phố Hải Phòng. Đề tài sẽ tập trung trình bày về việc quản lý vay vốn, thanh lý, tính lãi hàng tháng, hàng năm cho các khách hàng. Ngoài ra sẽ tạo các báo cáo tổng hợp theo tháng, quý, năm cho lãnh đạo.

Đồ án gồm 4 chương:

Chương 1 Tổng quan về PHP

Chương 2 Mô tả bài toán

Chương 3 Phân tích thiết kế hệ thống

Chương 4 Cài đặt chương trình

CHƯƠNG 1 TỔNG QUAN VỀ PHP

Chương này nói về sự phát triển của PHP qua từng giai đoạn, các phiên bản nâng cấp của PHP, các cấu trúc cơ bản của PHP và cơ sở lý thuyết về phân tích thiết kế hệ thống.

1.1 Lịch sử phát triển

PHP (viết tắt hồi quy "PHP: Hypertext Preprocessor") là một ngôn ngữ lập trình kịch bản hay một loại mã lệnh chủ yếu được dùng để phát triển các ứng dụng viết cho máy chủ, mã nguồn mở, dùng cho mục đích tổng quát. Nó rất thích hợp với web và có thể dễ dàng nhúng vào trang HTML. Do được tối ưu hóa cho các ứng dụng web, tốc độ nhanh, nhỏ gọn, cú pháp giống C và Java, dễ học và thời gian xây dựng sản phẩm tương đối ngắn hơn so với các ngôn ngữ khác nên PHP đã nhanh chóng trở thành một ngôn ngữ lập trình web phổ biến nhất thế giới. Sau đây là các phiên bản của PHP.

1.1.1 PHP

Được phát triển từ một sản phẩm có tên là PHP/FI. PHP/FI do Rasmus Lerdorf tạo ra năm 1995, ban đầu được xem như là một tập con đơn giản của các mã kịch bản Perl để theo dõi tình hình truy cập đến bản sơ yếu lý lịch của ông trên mạng. Ông đã đặt tên cho bộ mã kịch bản này là 'Personal Home Page Tools'...

1.1.2 PHP3

PHP 3.0 là phiên bản đầu tiên cho chúng ta thấy một hình ảnh gần gũi với các phiên bản PHP mà chúng ta được biết ngày nay. Nó đã được Andi Gutmans và Zeev Suraski tạo ra năm 1997 sau khi viết lại hoàn toàn bộ mã nguồn trước đó. PHP 3.0 đã chính thức được công bố vào tháng 6 năm 1998, sau thời gian 9 tháng được cộng đồng kiểm nghiệm.

1.1.3 PHP4

Vào mùa đông năm 1998, ngay sau khi PHP 3.0 chính thức được công bố,

Andi Gutmans và Zeev Suraski đã bắt đầu bắt tay vào việc viết lại phần lõi của PHP. Một động cơ mới, có tên 'Zend Engine' (ghép từ các chữ đầu trong tên của Zeev và Andi), đã đáp ứng được các nhu cầu thiết kế này một cách thành công, và lần đầu tiên được giới thiệu vào giữa năm 1999. PHP 4.0, dựa trên động cơ này, và đi kèm với hàng loạt các tính năng mới bổ sung, đã chính thức được công bố vào tháng 5 năm 2000, gần 2 năm sau khi bản PHP 3.0 ra đời.

1.1.4 PHP5

Sự thành công hết sức to lớn của PHP 4.0 đã không làm cho nhóm phát triển PHP tự mãn. Cộng đồng PHP đã nhanh chóng giúp họ nhận ra những yếu kém của PHP 4 đặc biệt với khả năng hỗ trợ lập trình hướng đối tượng (OOP), xử lý XML, không hỗ trợ giao thức máy khách mới của MySQL 4.1 và 5.0, hỗ trợ dịch vụ web yếu. Những điểm này chính là mục đích để Zeev và Andi viết Zend Engine 2.0, lõi của PHP 5.0. Ngày 29 tháng 6 năm 2003, PHP 5 Beta 1 đã chính thức được công bố để cộng đồng kiểm nghiệm. Đó cũng là phiên bản đầu tiên của Zend Engine 2.0. Phiên bản Beta 2 sau đó đã ra mắt vào tháng 10 năm 2003 với sự xuất hiện của hai tính năng rất được chờ đợi: Iterators, Reflection nhưng namespaces một tính năng gây tranh cãi khác đã bị loại khỏi mã nguồn. Ngày 21 tháng 12 năm 2003: PHP 5 Beta 3 đã được công bố để kiểm tra với việc phân phối kèm với Tidy, bỏ hỗ trợ Windows 95, khả năng gọi các hàm PHP bên trong XSLT, sửa chữa nhiều lỗi và thêm khá nhiều hàm mới. PHP 5 bản chính thức đã ra mắt ngày 13 tháng 7 năm 2004 sau một chuỗi khá dài các bản kiểm tra thử bao gồm Beta 4, RC 1, RC2, RC3. Mặc dù coi đây là phiên bản sản xuất đầu tiên nhưng PHP 5.0 vẫn còn một số lỗi trong đó đáng kể là lỗi xác thực HTTP.

1.1.5 PHP6

Hiện nay phiên bản tiếp theo của PHP đang được phát triển, PHP 6 bản sử dụng thử đã có thể được download tại địa chỉ <http://snaps.php.net>. Phiên bản PHP 6 được kỳ vọng sẽ lấp đầy những khiếm khuyết của PHP ở phiên bản hiện

tại, ví dụ: hỗ trợ namespace (hiện tại các nhà phát triển vẫn chưa công bố rõ ràng về vấn đề này); hỗ trợ Unicode; sử dụng PDO làm API chuẩn cho việc truy cập cơ sở dữ liệu, các API cũ sẽ bị đưa ra thành thư viện PECL...

1.2 Cấu trúc cơ bản của PHP

Phần này chủ yếu nói về các cấu trúc và cách sử dụng PHP.

1.2.1 Các cấu trúc cơ bản

PHP cũng có thể bắt đầu và kết thúc giống với ngôn ngữ HTML. Chỉ khác, đối với PHP chúng ta có nhiều cách để thể hiện.

Cách 1: Cú pháp chính:

```
<?php Mã lệnh PHP ?>
```

Cách 2: Cú pháp ngắn gọn

```
<? Mã lệnh PHP ?>
```

Cách 3: Cú pháp giống với ASP.

```
<% Mã lệnh PHP %>
```

Cách 4: Cú pháp bắt đầu bằng script

```
<script language=php>
```

.....

```
</script>
```

Mặc dù có 4 cách thể hiện. Nhưng đối với 1 lập trình viên có kinh nghiệm thì việc sử dụng cách 1 vẫn là lựa chọn tối ưu.

Trong PHP để kết thúc 1 dòng lệnh chúng ta sử dụng dấu ";"

Để chú thích 1 đoạn dữ liệu nào đó trong PHP ta sử dụng dấu "//" cho từng dòng.

Hoặc dùng cặp thẻ "/*.....*/" cho từng cụm mã lệnh.

Ví dụ: `<?php echo "Hello "; ?>`

1.2.2 Xuất giá trị ra trình duyệt

Để xuất dữ liệu ra trình duyệt chúng ta có những dòng cú pháp sau:

- `echo "Thông tin";`
- `print "Thông tin";`

Thông tin bao gồm: biến, chuỗi, hoặc lệnh HTML

```
1 <?php
2     echo "Hello";
3     print "Hello";
4 ?>
```

Hình 1: xuất giá trị

Nếu giữa hai chuỗi muốn liên kết với nhau ta sử dụng dấu "."

1.2.3 Biến, hằng, chuỗi và các kiểu dữ liệu

1.2.3.1 Biến

Biến được xem là vùng nhớ dữ liệu tạm thời. Và giá trị có thể thay đổi được. Biến được bắt đầu bằng ký hiệu "\$". Và theo sau chúng là 1 từ, 1 cụm từ nhưng phải viết liền hoặc có gạch dưới. Một biến được xem là hợp lệ khi nó thỏa các yếu tố:

- Tên của biến phải bắt đầu bằng dấu gạch dưới và theo sau là các ký tự, số hay dấu gạch dưới.
- Tên của biến không được phép trùng với các từ khóa của PHP.

Trong PHP để sử dụng 1 biến chúng ta thường phải khai báo trước, tuy nhiên đối với các lập trình viên khi sử dụng họ thường xử lý cùng một lúc các công việc, nghĩa là vừa khai báo vừa gán dữ liệu cho biến. Bản thân biến cũng có thể gán cho các kiểu dữ liệu khác. Và tùy theo ý định của người lập trình mong muốn trên chúng.

```

<?php
    $bien_double_1 = 2.2888800;
    $bien_double_2 = 2.2111200;
    $ket_qua = $bien_double_1 + $bien_double_2;

    print("$bien_double_1 + $bien_double_2 = $ket_qua <br>");
?>

```

Hình 2: Biến trong PHP

1.2.3.2 Hằng

Nếu biến là cái có thể thay đổi được thì ngược lại hằng là cái chúng ta không thể thay đổi được. Hằng trong PHP được định nghĩa bởi hàm define theo cú pháp:

```
define (string tên_hằng, giá_trị_hằng ).
```

Cũng giống với biến hằng được xem là hợp lệ thì chúng phải đáp ứng 1 số yếu tố:

- Hằng không có dấu "\$" ở trước tên.
- Hằng có thể truy cập bất cứ vị trí nào trong mã lệnh
- Hằng chỉ được phép gán giá trị duy nhất 1 lần.
- Hằng thường viết bằng chữ in để phân biệt với biến

```

// Ví dụ các tên hằng hợp lệ
define("ONE",      "Ví dụ tên hằng hợp lệ 1");
define("TWO2",     "Ví dụ tên hằng hợp lệ 2");
define("THREE_3",  "Ví dụ tên hằng hợp lệ 3")

// Ví dụ các tên hằng không hợp lệ
define("2TWO",     "Ví dụ tên hằng không hợp lệ 1");
define("__THREE__", "Ví dụ tên hằng không hợp lệ 2");

```

Hình 3: Hằng trong PHP

1.2.3.3 Chuỗi

Chuỗi là một nhóm các ký tự, số, khoảng trắng, dấu ngắt được đặt trong

các dấu nháy.

Ví dụ: “Hello”

Để tạo 1 biến chuỗi, chúng ta phải gán giá trị chuỗi cho 1 biến hợp lệ.

Ví dụ:

```
$first_name= “Nguyen”;
```

```
$last_name= “Van A”;
```

Để liên kết 1 chuỗi và 1 biến chúng ta thường sử dụng dấu "."

```
$string_1 = "Vi du mot chuoai trong PHP";  
$string_2 = "Hoc PHP co ban va nang cao tai VietJack";  
$string_0 = ""; // ví dụ một chuỗi không có ký tự nào
```

Hình 4: Chuỗi trong PHP

1.2.3.4 Kiểu dữ liệu

Các kiểu dữ liệu khác nhau chiếm các lượng bộ nhớ khác nhau và có thể được xử lý theo cách khác nhau khi chúng được theo tác trong 1 script.

Trong PHP chúng ta có 6 kiểu dữ liệu chính như sau:

Kiểu Dữ Liệu	Ví dụ	Mô Tả
Integer	10	Một số nguyên
Double	5.208	Kiểu số thực
String	“How are you ?”	Một tập hợp các ký tự
Boolean	True or False	Giá trị true hoặc false
Object	Hướng đối tượng trong PHP	
Array	Mảng trong PHP, chứa các phần tử.	

Hình 5: Kiểu dữ liệu trong PHP

1.2.4 Các phương thức được sử dụng trong lập trình PHP

1.2.4.1 Phương thức GET

Phương thức này được dùng để lấy dữ liệu từ form nhập liệu. Tuy nhiên nhiệm vụ chính của nó vẫn là lấy nội dung từ trang dữ liệu từ web server.

Ví dụ:

Với url sau: `shownews.php?id=50`

Vậy với trang shownews ta dùng hàm `$_GET["id"]` sẽ được giá trị là 50.

1.2.4.2 Phương thức POST

Phương thức này được sử dụng để lấy dữ liệu từ form nhập liệu. Và chuyển chúng lên trình chủ webserver.

```
<form method="post">  
</form>
```

Hình 6: Phương thức POST trong PHP

1.2.5 Cookie và Session trong PHP

Cookie và Session là hai phương pháp sử dụng để quản lý các phiên làm việc giữa người sử dụng và hệ thống.

1.2.5.1 Cookie

Cookie là 1 đoạn dữ liệu được ghi vào đĩa cứng hoặc bộ nhớ của máy người sử dụng. Nó được trình duyệt gửi ngược lên lại server mỗi khi browser tải 1 trang web từ server.

Những thông tin được lưu trữ trong cookie hoàn toàn phụ thuộc vào Website trên server. Mỗi Website có thể lưu trữ những thông tin khác nhau trong cookie, ví dụ thời điểm lần cuối ta ghé thăm Website, đánh dấu ta đã login hay chưa, v.v...

Cookie được tạo ra bởi Website và gửi tới browser, do vậy 2 Website khác nhau cho dù cùng host trên 1 server sẽ có 2 cookie khác nhau gửi tới browser. Ngoài ra, mỗi browser quản lý và lưu trữ cookie theo cách riêng của mình, cho nên 2 browser cùng truy cập vào 1 Website sẽ nhận được 2 cookie khác nhau

- Để thiết lập cookie ta sử dụng cú pháp:

`setcookie("tên cookie","giá trị", thời gian sống)`

Tên cookie là tên mà chúng ta đặt cho phiên làm việc.

Giá trị là thông số của tên cookie.

Ví dụ: `setcookie("name","admin",time()+3600);`

- Để sử dụng lại cookie vừa thiết lập, chúng ta sử dụng cú pháp:

Cú pháp: `$_COOKIE["tên cookies"]`

Tên cookie là tên mà chúng ta thiết lập phía trên.

- Để hủy 1 cookie đã được tạo ta có thể dùng 1 trong 2 cách sau:

Cú pháp: `setcookie("Tên cookie")`

Gọi hàm `setcookie` với chỉ duy nhất tên cookie mà thôi

Dùng thời gian hết hạn cookie là thời điểm trong quá khứ.

Ví dụ: `setcookie("name","admin",time()-3600);`

1.2.5.2 Session

Một cách khác quản lý người sử dụng là Session. Session được hiểu là khoảng thời gian người sử dụng giao tiếp với 1 ứng dụng. Một session được bắt đầu khi người sử dụng truy cập vào ứng dụng lần đầu tiên, và kết thúc khi người sử dụng thoát khỏi ứng dụng. Mỗi session sẽ có được cấp một định danh (ID) khác nhau.

Để thiết lập 1 session ta sử dụng cú pháp: `session_start()` Đoạn code này phải được nằm trên các kịch bản HTML. Hoặc những lệnh `echo`, `printf`. Để thiết lập 1 giá trị session, ngoài việc cho phép bắt đầu thực thi session. Chúng ta còn phải đăng ký 1 giá trị session. Để tiện cho việc gán giá trị cho session đó.

Ta có cú pháp sau: `session_register("name")`

Giống với cookie. Để sử dụng giá trị của session ta sử dụng mã lệnh sau:

```
1 $_SESSION["name"]
```

Hình 7: Cú pháp của session

Chú thích: “name” là tên mà chúng ta sử dụng hàm `session_register("name")` để khai báo.

Để hủy bỏ giá trị của session ta có những cách sau:

```
1 session_destroy() // Cho phép hủy bỏ toàn bộ giá trị của session
2 session_unset() // Cho phép hủy bỏ session
```

Hình 8: Cú pháp hủy bỏ giá trị session

1.2.6 Cookie và Session trong PHP

Để giảm thời gian lặp lại 1 thao tác code nhiều lần, PHP hỗ trợ người lập trình việc tự định nghĩa cho mình những hàm có khả năng lặp lại nhiều lần trong Website. Việc này cũng giúp cho người lập trình kiểm soát mã nguồn một cách mạch lạc. Đồng thời có thể tùy biến ở mọi trang. Mà không cần phải khởi tạo hay viết lại mã lệnh như HTML thuần.

1.2.6.1 Hàm tự định nghĩa

Cú pháp:

```
1 function function_name()
2 {
3 //Lệnh thực thi
4 }
```

Hình 9: Cú pháp hàm tự định nghĩa

Tên hàm có thể là một tổ hợp bất kỳ những chữ cái, con số và dấu gạch dưới, nhưng phải bắt đầu từ chữ cái và dấu gạch dưới.

1.2.6.2 Hàm tự định nghĩa với các tham số

Cú pháp:

```
1 function function_name($gt1,$gt2)
2 {
3 // lệnh thực thi
4 }
```

Hình 10: Cú pháp hàm tự định nghĩa

Chú thích:

\$gt1 là giá trị thứ nhất

\$gt2 là giá trị thứ hai

Ngoài ra hàm có thể có thêm nhiều giá trị.

1.2.6.3 Hàm tự định nghĩa với giá trị trả về

Cú pháp:

```
1 function function_name()
2 {
3 // Lệnh thực thi
4 return kq;
5 }
```

Hình 11: Hàm tự định nghĩa với giá trị trả về

Chú thích:

Kq là kết quả trả về của hàm

Hàm có thể có hoặc không có đối số

1.2.6.4 Gọi lại hàm

PHP cung cấp nhiều hàm cho phép triệu gọi lại file. Như hàm `include("URL đến file")`, `require("URL Đến file")`. Ngoài hai cú pháp trên còn có `include_once()`, `require_once()`. Hai hàm này cũng có trách nhiệm gọi lại hàm. Những chúng sẽ chỉ gọi lại duy nhất 1 lần mà thôi.

1.3 MySQL

1.3.1 Giới thiệu cơ sở dữ liệu:

MySQL là ứng dụng cơ sở dữ liệu mã nguồn mở phổ biến nhất hiện nay (theo www.mysql.com) và được sử dụng phối hợp với PHP. Trước khi làm việc với MySQL cần xác định các nhu cầu cho ứng dụng.

MySQL là cơ sở dữ có trình giao diện trên Windows hay Linux, cho phép

người sử dụng có thể thao tác các hành động liên quan đến nó. Việc tìm hiểu từng công nghệ trước khi bắt tay vào việc viết mã kịch bản PHP, việc tích hợp hai công nghệ PHP và MySQL là một công việc cần thiết và rất quan trọng.

1.3.2 Mục đích sử dụng cơ sở dữ liệu:

Mục đích sử dụng cơ sở dữ liệu bao gồm các chức năng như: lưu trữ (storage), truy cập (accessibility), tổ chức (organization) và xử lý (manipulation).

- Lưu trữ: Lưu trữ trên đĩa và có thể chuyển đổi dữ liệu từ cơ sở dữ liệu này sang cơ sở dữ liệu khác, nếu sử dụng cho quy mô nhỏ, có thể chọn cơ sở dữ liệu nhỏ như:

Microsoft Exel, Microsoft Access, MySQL, Microsoft Visual FoxPro,...
Nếu ứng dụng có quy mô lớn, có thể chọn cơ sở dữ liệu có quy mô lớn như: Oracle, SQL Server,...

- Truy cập: Truy cập dữ liệu phụ thuộc vào mục đích và yêu cầu của người sử dụng, ở mức độ mang tính cục bộ, truy cập cơ sở dữ liệu ngay trong cơ sở dữ liệu với nhau, nhằm trao đổi hay xử lý dữ liệu ngay bên trong chính nó, nhưng do mục đích và yêu cầu người dùng vượt ra ngoài cơ sở dữ liệu, nên cần có các phương thức truy cập dữ liệu giữa các cơ sở dữ liệu với nhau như: Microsoft Access với SQL Server, hay SQL Server và cơ sở dữ liệu Oracle....
- Tổ chức: Tổ chức cơ sở dữ liệu phụ thuộc vào mô hình cơ sở dữ liệu, phân tích và thiết kế cơ sở dữ liệu tức là tổ chức cơ sở dữ liệu phụ thuộc vào đặc điểm riêng của từng ứng dụng. Tuy nhiên khi tổ chức cơ sở dữ liệu cần phải tuân theo một số tiêu chuẩn của hệ thống cơ sở dữ liệu nhằm tăng tính tối ưu khi truy cập và xử lý.
- Xử lý: Tùy vào nhu cầu tính toán và truy vấn cơ sở dữ liệu với các mục đích khác nhau, cần phải sử dụng các phát biểu truy vấn cùng các phép toán, phát biểu của cơ sở dữ liệu để xuất ra kết quả như yêu cầu. Để thao

tác hay xử lý dữ liệu bên trong chính cơ sở dữ liệu ta sử dụng các ngôn ngữ lập trình như: PHP, C++, Java, Visual Basic,...

1.3.3 Các kiểu dữ liệu trong cơ sở dữ liệu MySQL

Kiểu dữ liệu numeric: bao gồm số nguyên và kiểu số chấm động.

Loại	Phạm vi	Bytes
TINYINT	-127 => 128	1
SMALLINT	-32768 => 32767	2
MEDIUMINT	-8388608 => 8388607	3
INT	-2147483648 => 2147483647	4
BIGINT	-9223372036854775808 => 9223372036854775807	8

Kiểu chấm động

Loại	Phạm vi	Bytes
FLOAT	-3.402823466E+38=>-1.175494351E-38	4
DOUBLE	-1.7976931348623157E+ 308 => -2.2250738585072014E- 308	8

Kiểu dữ Date and Time cho phép nhập dữ liệu dưới dạng chuỗi ngày tháng hay dạng số

Loại	Định dạng hiển thị	Phạm vi
DATETIME	YYYY-MM-DD HH:MM:SS	'1000-01-01 00:00:00' to '9999-12-31 23:59:59'

DATE	YYYY-MM-DD	'1000-01-01' to '9999-12-31'
TIMESTAMP	YYYY-MM-DD HH:MM:SS	'1970-01-01 00:00:01' UTC to '2038-01-19 03:14:07' UTC

Kiểu dữ liệu String: chia làm 3 loại: char (chiều dài cố định) và varchar (chiều dài biến thiên); Text (cho phép lưu chuỗi lớn) và Blob (cho phép lưu đối tượng nhị phân); Enum và Set.

Loại	Giới hạn	Mô tả
CHAR	Kích thước tối đa 255 ký tự.	Có chiều dài cố định, có tối đa đoạn string là 255 ký tự, nếu vượt quá giới hạn cho phép, MySQL sẽ tự động cắt chuỗi.
VARCHAR	Kích thước tối đa 255 ký tự.	Có chiều dài cố định, có tối đa đoạn string là 255 ký tự, nếu vượt quá giới hạn cho phép, MySQL sẽ tự động cắt chuỗi.
TINYTEXT	Kích thước tối đa 255 ký tự.	Có chiều dài cố định, có tối đa đoạn string là 255 ký tự, nếu vượt quá giới hạn cho phép, MySQL sẽ tự động cắt chuỗi.

TEXT	Kích thước tối đa 65,535 ký tự.	Có tối đa đoạn string là 65,535 ký tự.
MEDIUMTEXT	Kích thước tối đa 16,777,215 ký tự.	Where <i>size</i> is the number of characters to store.
LONGTEXT	Kích thước tối đa 4GB or 4,294,967,295 ký tự.	Kích thước tối đa là 4GB or 4,294,967,295 ký tự.
BINARY	Kích thước tối đa 255 ký tự.	Có tối đa đoạn string là 255 ký tự.
VARBINARY	Kích thước tối đa 255 ký tự..	

1.3.4 Các thao tác cập nhật dữ liệu

SELECT (truy vấn mẫu tin): SELECT dùng để truy vấn từ một hay nhiều bảng khác nhau, kết quả trả về là một tập mẫu tin thỏa mãn các điều kiện cho trước nếu có, cú pháp phát biểu SQL dạng SELECT như sau:

```

1  SELECT
2  FROM < danh sách các bảng >
3  WHERE < các điều kiện ràng buộc >
4  GROUP BY < tên cột / biểu thức >

```

Hình 12: Câu lệnh SELECT

INSERT (thêm mẫu tin):

Cú pháp:

```

1  INSERT INTO Tên_bảng VALUE ( bộ giá trị )
2
3

```

Hình 13: Câu lệnh INSERT

Update (cập nhật dữ liệu):

Cú pháp:

```
1 UPDATE TABLE tên_bảng
2 SET tên_cột = Biểu_thức
3 WHERE <điều_kiện>
4
```

Hình 14: Câu lệnh UPDATE

Delete (xóa mẫu tin):

Cú pháp:

```
1 DELETE FROM tên_bảng
2 WHERE <điều_kiện>
3
```

Hình 15: Câu lệnh DELETE

1.3.5 Các hàm thông dụng trong MySQL

- Hàm AVG: hàm trả về giá trị bình quân của cột hay trường trong câu truy vấn.
- Hàm MIN: hàm trả về giá trị nhỏ nhất của cột hay trường trong câu truy vấn.
- Hàm MAX: hàm trả về giá trị lớn nhất của cột hay trường trong câu truy vấn.
- Hàm COUNT: hàm trả về số lượng mẫu tin trong câu truy vấn.
- Hàm SUM: hàm trả về tổng các giá trị của cột, trường trong câu truy vấn.
- Các hàm xử lý chuỗi
- Hàm ASCII: hàm trả về giá trị mã ASCII của ký tự bên trái của chuỗi.
- Hàm CHAR: hàm chuyển đổi kiểu mã ASCII từ số nguyên sang dạng chuỗi.
- Hàm UPPER: hàm chuyển đổi chuỗi sang kiểu chữ hoa.
- Hàm LOWER: hàm chuyển đổi chuỗi sang kiểu chữ thường.

- Hàm LEN: hàm trả về chiều dài của chuỗi.
- Hàm LTRIM: hàm loại bỏ khoảng trắng bên trái chuỗi.
- Hàm RTRIM: hàm loại bỏ khoảng trắng bên phải chuỗi.
- Hàm LEFT(STR, N): hàm trả về chuỗi bên trái tính từ đầu cho đến vị trí n.
- Hàm RIGHT(STR, N): hàm trả về chuỗi bên phải tính từ đầu cho đến vị trí n.
- Hàm INSTR: hàm trả về chuỗi vị trí bắt đầu của chuỗi con trong chuỗi xét.
- Các hàm xử lý về thời gian
- Hàm CURDATE(): hàm trả về ngày, tháng, năm hiện hành của hệ thống.
- Hàm CURTIME(): hàm trả về giờ, phút, giây hiện hành của hệ thống.
- Hàm Period_Diff: hàm trả về số ngày trong khoảng thời gian giữa 2 ngày.
- Hàm dayofmonth: hàm trả ngày thứ mấy trong tháng

1.4 Phân tích thiết kế hệ thống hướng cấu trúc

1.4.1 Quá trình phát triển của một hệ thống thông tin

Hệ thống thông tin (HTTT) được xây dựng là sản phẩm của một loạt các hoạt động được gọi là phát triển hệ thống. Quá trình phát triển một HTTT kể từ lúc nó sinh ra đến khi nó tàn lụi được gọi là vòng đời phát triển hệ thống. Vòng đời phát triển hệ thống là một phương pháp luận cho việc phát triển các HTTT. Nó được đặc trưng bằng một số pha chủ yếu phân biệt nhau của quá trình đó là phân tích, thiết kế và triển khai HTTT.

Quá trình phát triển một hệ thống trải qua các pha: Khảo sát dự án, phân tích hệ thống, thiết kế, thực hiện, kiểm thử, vận hành và bảo trì.

Giai đoạn 1: Khảo sát dự án

Khảo sát hiện trạng là giai đoạn đầu tiên trong quá trình phát triển một hệ thống thông tin. Nhiệm vụ chính trong giai đoạn này là tìm hiểu, thu thập thông tin cần thiết để chuẩn bị cho việc giải quyết các yêu cầu được đặt ra của dự án.

Giai đoạn khảo sát được chia làm hai bước:

Bước 1:

- Khảo sát sơ bộ: tìm hiểu các yếu tố cơ bản (tổ chức, văn hóa, đặc trưng, con người,...) tạo tiền đề để phát triển HTTT phù hợp với dự án và doanh nghiệp.
- Khảo sát chi tiết: thu thập thông tin chi tiết của hệ thống (chức năng xử lý, thông tin được phép nhập và xuất khỏi hệ thống, ràng buộc, giao diện cơ bản, nghiệp vụ) phục vụ cho việc phân tích và thiết kế.

Bước 2: Đặt ra các vấn đề trọng tâm cần phải giải quyết, như:

- Thông tin đưa vào hệ thống phải như thế nào?
- Dữ liệu hiển thị và xuất ra khác nhau ở những điểm nào?
- Ràng buộc giữa các đối tượng trong hệ thống cần xây dựng ra sao?
- Chức năng và quy trình xử lý của hệ thống phải đảm bảo những yêu cầu nào?
- Cần sử dụng những giải pháp nào? Tính khả thi của từng giải pháp ra sao?

Từ những thông tin thu thập được và vấn đề đã đặt ra trong giai đoạn khảo sát, nhà quản trị và các chuyên gia sẽ chọn lọc những yếu tố cần thiết để cấu thành hệ thống thông tin riêng cho doanh nghiệp.

Giai đoạn 2: Phân tích hệ thống

Mục tiêu của giai đoạn là xác định các thông tin và chức năng xử lý của hệ thống, cụ thể như sau:

- Xác định *yêu cầu của HTTT* gồm: các chức năng chính - phụ; nghiệp vụ cần phải xử lý đảm bảo tính chính xác, tuân thủ đúng các văn bản luật và quy định hiện hành; đảm bảo tốc độ xử lý và khả năng nâng cấp trong tương lai.

- Phân tích và đặc tả *mô hình phân cấp chức năng tổng thể* thông qua sơ đồ BFD (Business Flow Diagram), từ mô hình BFD sẽ tiếp tục được xây dựng thành mô hình luồng dữ liệu DFD (Data Flow Diagram) thông qua quá trình phân rã chức năng theo các mức 0, 1, 2 ở từng ô xử lý.
- Phân tích *bảng dữ liệu*. Cần đưa vào hệ thống những bảng dữ liệu (data table) gồm các trường dữ liệu (data field) nào? Xác định khóa chính (primary key), khóa ngoại (foreign key) cũng như mối quan hệ giữa các bảng dữ liệu (relationship) và ràng buộc (constraint) dữ liệu cần thiết.
- Ở giai đoạn này, các chuyên gia sẽ đặc tả sơ bộ các bảng dữ liệu trên giấy để có cái nhìn khách quan. Qua đó, xác định các giải pháp tốt nhất cho hệ thống đảm bảo đúng các yêu cầu đã khảo sát trước khi thực hiện trên các phần mềm chuyên dụng.

Giai đoạn 3: Thiết kế

Thông qua thông tin được thu thập từ quá trình khảo sát và phân tích, các chuyên gia sẽ chuyển hóa vào phần mềm, công cụ chuyên dụng để đặc tả thiết kế hệ thống chi tiết. Giai đoạn này được chia làm hai bước sau:

Bước 1: Thiết kế tổng thể

Trên cơ sở các bảng dữ liệu đã phân tích và đặc tả trên giấy sẽ được thiết kế dưới dạng mô hình mức ý niệm bằng phần mềm chuyên dụng như Sybase PowerDesigner, CA ERwin Data Modeler. Bảng mô hình mức ý niệm sẽ cho các chuyên gia có cái nhìn tổng quát nhất về mối quan hệ giữa các đối tượng trước khi chuyển đổi thành mô hình mức vật lý.

Bước 2: Thiết kế chi tiết

- Thiết kế cơ sở dữ liệu (Database): Với mô hình mức vật lý hoàn chỉnh ở giai đoạn thiết kế đại thể sẽ được kết sinh mã thành file sql.
- Thiết kế truy vấn, thủ tục, hàm: thu thập, xử lý thông tin nhập và đưa ra thông tin chuẩn xác theo đúng nghiệp vụ.

- Thiết kế giao diện chương trình đảm bảo phù hợp với môi trường, văn hóa và yêu cầu của doanh nghiệp thực hiện dự án.
- Thiết kế chức năng chương trình đảm bảo tính logic trong quá trình nhập liệu và xử lý cho người dùng.
- Thiết kế báo cáo. Dựa trên các yêu cầu của mỗi doanh nghiệp và quy định hiện hành sẽ thiết kế các mẫu báo cáo phù hợp hoặc cho phép doanh nghiệp tự tạo mẫu báo cáo ngay trên hệ thống.
- Thiết kế các kiểm soát bằng hình thức đưa ra các thông báo, cảnh báo hoặc lỗi cụ thể tạo tiện lợi và kiểm soát chặt chẽ quá trình nhập liệu với mục tiêu tăng độ chính xác cho dữ liệu.

Tóm lại, *thiết kế là việc áp dụng các công cụ, phương pháp, thủ tục để tạo ra mô hình hệ thống cần sử dụng*. Sản phẩm cuối cùng của giai đoạn thiết kế là đặc tả hệ thống ở dạng nó tồn tại thực tế, sao cho nhà lập trình và kỹ sư phần cứng có thể dễ dàng chuyển thành chương trình và cấu trúc hệ thống.

Giai đoạn 4: Thực hiện

Đây là giai đoạn nhằm xây dựng hệ thống theo các thiết kế đã xác định. Giai đoạn này bao gồm các công việc sau:

- Lựa chọn hệ quản trị cơ sở dữ liệu (SQL Server, Oracle, MySQL, ...) và cài đặt cơ sở dữ liệu cho hệ thống.
- Lựa chọn công cụ lập trình để xây dựng các modules chương trình của hệ thống (Microsoft Visual Studio, PHP Designer,...).
- Lựa chọn công cụ để xây dựng giao diện hệ thống (DevExpress, Dot Net Bar,...). Viết tài liệu hướng dẫn sử dụng, tài liệu kỹ thuật hoặc clip hướng dẫn.

Giai đoạn 5: Kiểm thử

- Trước hết phải lựa chọn công cụ kiểm thử.

- Kiểm chứng các modules chức năng của hệ thống thông tin, chuyển các thiết kế thành các chương trình (phần mềm).
- Thử nghiệm hệ thống thông tin.
- Cuối cùng là khắc phục các lỗi (nếu có).
- Viết test case theo yêu cầu.

Kết quả cuối cùng là một hệ thống thông tin đạt yêu cầu đặt ra.

Giai đoạn 6: Triển khai và bảo trì

- Lắp đặt phần cứng để làm cơ sở cho hệ thống.
- Cài đặt phần mềm.
- Chuyển đổi hoạt động của hệ thống cũ sang hệ thống mới, gồm có: chuyển đổi dữ liệu; bố trí, sắp xếp người làm việc trong hệ thống; tổ chức hệ thống quản lý và bảo trì.
- Phát hiện các sai sót, khuyết điểm của hệ thống thông tin.
- Đào tạo và hướng dẫn sử dụng.
- Cải tiến và chỉnh sửa hệ thống thông tin.
- Bảo hành.
- Nâng cấp chương trình khi có phiên bản mới.

1.4.2 Tiếp cận định hướng cấu trúc

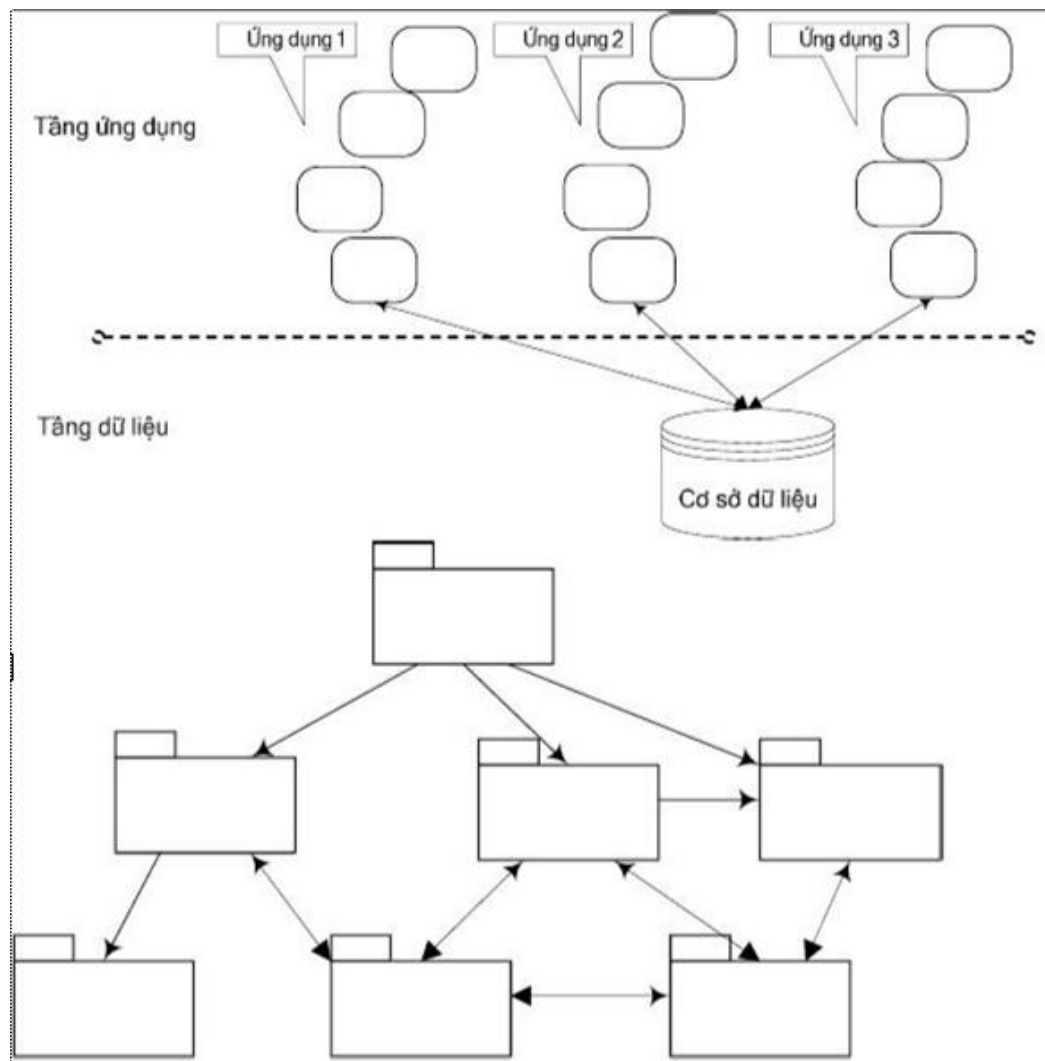
Tiếp cận định hướng cấu trúc như một bước phát triển tiếp tục của định hướng dữ liệu. Nhiều tài liệu thường gộp hai cách tiếp cận này làm một và gọi là tiếp cận hướng dữ liệu chức năng. Tiếp cận hướng cấu trúc hướng vào việc cải tiến cấu trúc các chương trình dựa trên cơ sở modun hóa để dễ theo dõi, quản lý, bảo trì.

Phát triển hướng cấu trúc đề cập đến quá trình sử dụng một cách có hệ thống và tích hợp các công cụ và kỹ thuật để trợ giúp thiết kế phân tích HTTT theo hướng modun hóa.

Các phương pháp luận hướng cấu trúc sử dụng một hay một số công cụ để xác định luồng thông tin và các quá trình xử lý.

Việc xác định và chi tiết hóa dần các luồng dữ liệu và các tiến trình là ý tưởng cơ bản của phương pháp luận từ trên xuống (Top-down).

Từ mức 0: mức chung nhất, quá trình tiếp tục làm mịn cho đến mức thấp nhất: mức cơ sở. Ở đó từ các sơ đồ nhận được ta có thể bắt đầu tạo lập các chương trình với các modul thấp nhất (modun lá)



Hình 16: Cấu trúc hệ thống định hướng cấu trúc

Phát triển có cấu trúc đã cung cấp một tập hợp đầy đủ các đặc tả hệ thống không dư thừa được phát triển theo quá trình logic và lặp lại.

Nó cho ta nhiều lợi ích so với cách tiếp cận trước đó:

- Làm giảm sự phức tạp (nhờ chia nhỏ, modun hóa)
- Tập trung vào ý tưởng (vào logic, kiến trúc trước khi thiết kế)
- Chuẩn mực hóa (theo các phương pháp, công cụ đã cho)
- Hướng về tương lai (kiến trúc tốt, modun hóa dễ bảo trì)
- Giảm bớt tính nghệ thuật trong thiết kế

1.5 Thiết kế cơ sở dữ liệu quan hệ

1.5.1 Mô hình liên kết thực thể ER

1.5.1.1 Định nghĩa

E-R là mô hình trung gian để chuyển những yêu cầu quản lý dữ liệu trong thế giới thực thành mô hình cơ sở dữ liệu quan hệ

1.5.1.2 Các thành phần cơ bản của mô hình E-R

➤ Thực thể và tập thực thể

Thực thể là một đối tượng trong thế giới thực.

Một nhóm bao gồm các thực thể tương tự nhau tạo thành một tập thực thể

Việc lựa chọn các tập thực thể là một bước vô cùng quan trọng trong việc xây dựng sơ đồ về mối quan hệ thực thể

Ví dụ: “Quản lý các dự án của công ty”

Một nhân viên là một thực thể

Tập hợp các nhân viên là tập thực thể

Một dự án là một thực thể

Tập hợp các dự án là tập thực thể

Một phòng ban là một thực thể

Tập hợp các phòng ban là tập thực thể

➤ Thuộc tính:

Mỗi tập thực thể có một tập các tính chất đặc trưng, mỗi tính chất đặc trưng này gọi là thuộc tính của tập thực thể. Ứng với mỗi thuộc tính có một tập các giá trị cho thuộc tính đó gọi là miền giá trị.

Miền giá trị của thuộc tính gồm các loại giá trị như sau:

Kiểu chuỗi (string)

Kiểu số nguyên (integer)

Kiểu số thực (real)

...

Ví dụ tập thực thể NHANVIEN có các thuộc tính:

Họ tên (hoten: string[20])

Ngày sinh (ns: date)

...

Thuộc tính bao gồm các loại như sau:

Thuộc tính đơn – không thể tách nhỏ ra được

Thuộc tính phức hợp – có thể tách ra thành các thành phần nhỏ hơn

Các loại giá trị của thuộc tính:

Đơn trị: các thuộc tính có giá trị duy nhất cho một thực thể (VD: số CMND, ...)

Đa trị: các thuộc tính có một tập giá trị cho cùng một thực thể

(VD: bằng cấp, ...)

Suy diễn được (năm sinh <----> tuổi)

Mỗi thực thể đều được phân biệt bởi thuộc tính khóa

Ví dụ 1: tập thực thể NHANVIEN có các thuộc tính

Mã NV (MaNV: integer)

Họ tên (Hoten: string[50])

Ngày sinh (ns:date)

Địa chỉ (diachi:string[100])

Quê quán (quequan:string[30])

Hệ số lương (hsluong:float)

Hệ số phụ cấp (hsphucap:float)

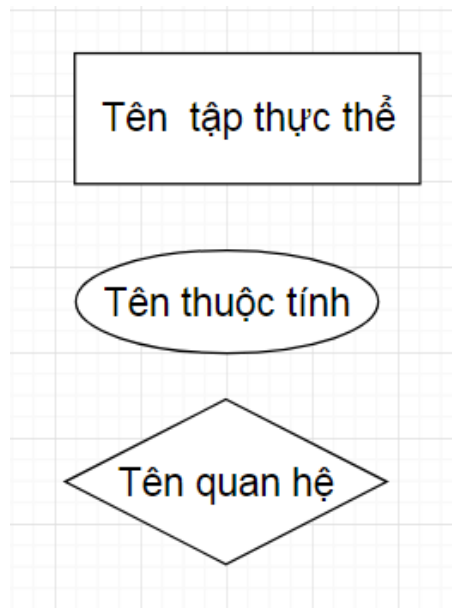
Tổng lương (tongluong:float)

➤ Lược đồ E-R:

Là đồ thị biểu diễn các tập thực thể, thuộc tính và mối quan hệ

- Các ký hiệu trong lược đồ E-R

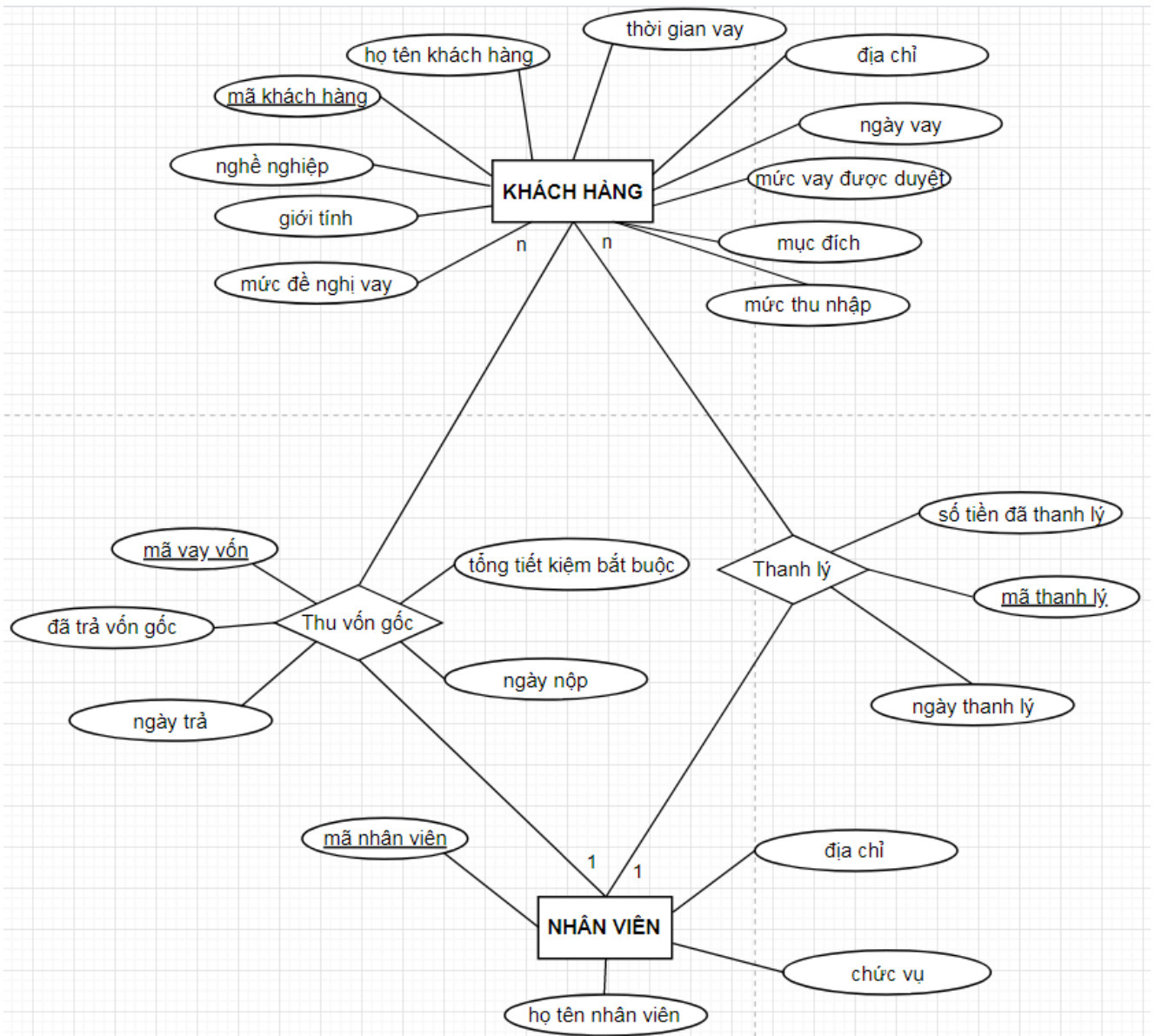
Đỉnh:



Hình 17: Các ký hiệu trong lược đồ E-R

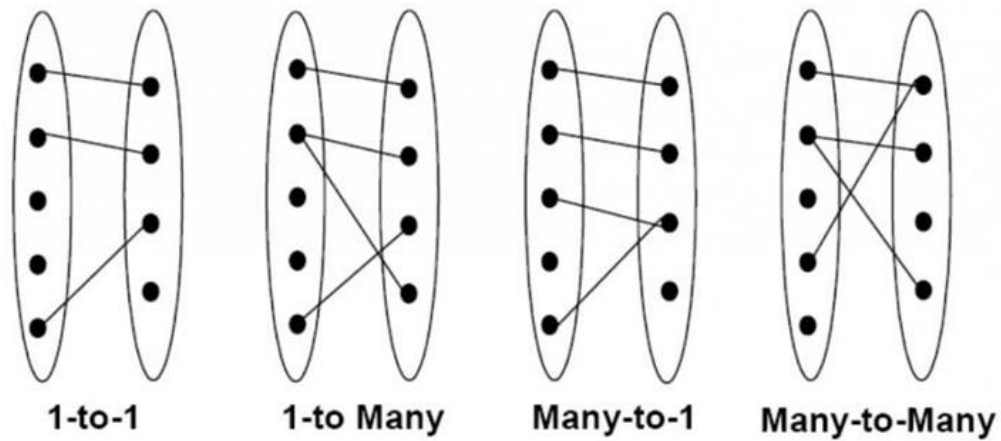
Cung: là đường nối giữa tập thực thể và thuộc tính, mối quan hệ và tập thực thể

Ví dụ lược đồ E-R:



Hình 18: Ví dụ lược đồ E-R

- Các kiểu liên kết trong lược đồ E-R:



Hình 19: Các kiểu liên kết trong lược đồ E-R

Ví dụ:

Một nhân viên thu vốn của nhiều khách hàng



Hình 20: Ví dụ liên kết 1 nhiều

- Thuộc tính khóa:

Mỗi tập thực thể phải có 1 khóa

Một khóa có thể có 1 hay nhiều thuộc tính

Có thể có nhiều khóa trong 1 tập thực thể, ta sẽ chọn ra 1 khóa làm khóa chính cho tập thực thể đó.

CHƯƠNG 2 MÔ TẢ BÀI TOÁN

Chương này giúp ta hiểu được bài toán, cách phát biểu bài toán và cách phân tích bài toán một cách cụ thể. Ngoài ra bài toán sẽ được phân tách ra thành các tiến trình và nghiệp vụ cụ thể.

Trong đề án này em sẽ trình bày về ứng dụng quản lý cho vay vốn của quỹ vay vốn của liên đoàn lao động thành phố Hải Phòng. Ứng dụng sẽ gồm các bảng biểu, danh sách số liệu của gốc vay, vốn đã trả, còn phải trả và thu tiết kiệm bắt buộc, các bảng báo cáo tổng hợp lãi vay, thu hàng tháng của từng khách hàng.

2.1 Phát biểu bài toán quản lý cho vay dành cho công nhân viên chức nghèo của liên đoàn lao động thành phố Hải Phòng

Khách hàng được phê duyệt vay sẽ ký kết hợp đồng vay vốn và hoàn tất thủ tục vay. Khi được duyệt, trong thời gian vay hàng tháng khách hàng phải trả 3 khoản bắt buộc (gốc, lãi được tính 0.6% số tiền còn lại, tiết kiệm bắt buộc được tính 1% tổng tiền vay). Khi kết thúc hợp đồng thì số tiền tiết kiệm bắt buộc trong thời gian vay được trả lại cho người vay.

Hàng tháng CĐCT xuất ra phiếu thu đến từng CĐCS để báo thu. Người vay có thể nộp trực tiếp lên CĐCT hoặc nộp cho 1 người đại diện CĐCS lên nộp. Hàng tháng sẽ xuất báo cáo thống kê số người vay của từng CĐCS, thống kê số người thanh lý, thống kê theo dõi thu của từng CĐCT, thống kê tổng hợp hàng tháng gửi lên ban lãnh đạo.

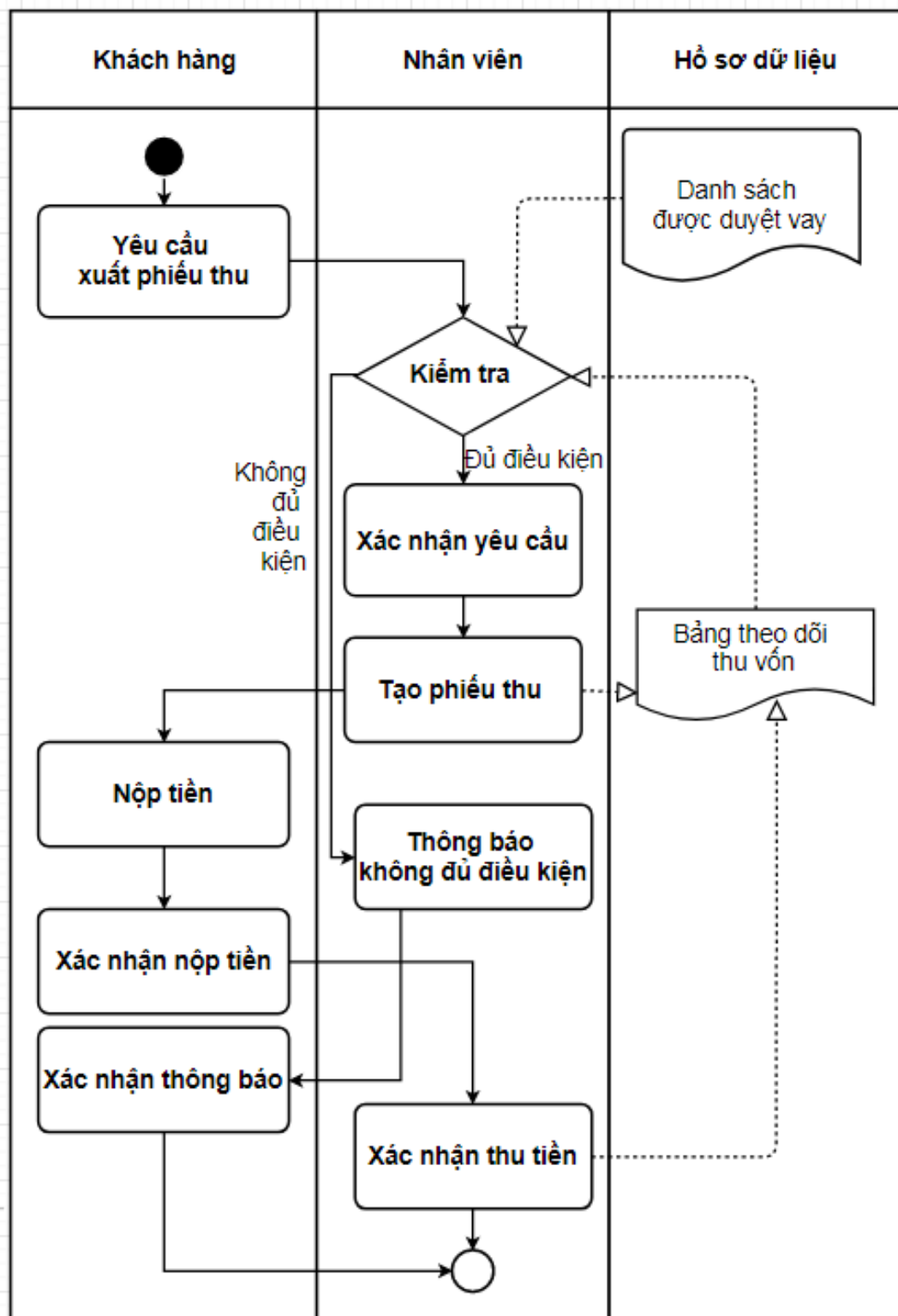
Chú thích:

CĐCS là công đoàn cơ sở

CĐCT là công đoàn cấp trên

2.2 Sơ đồ tiến trình nghiệp vụ

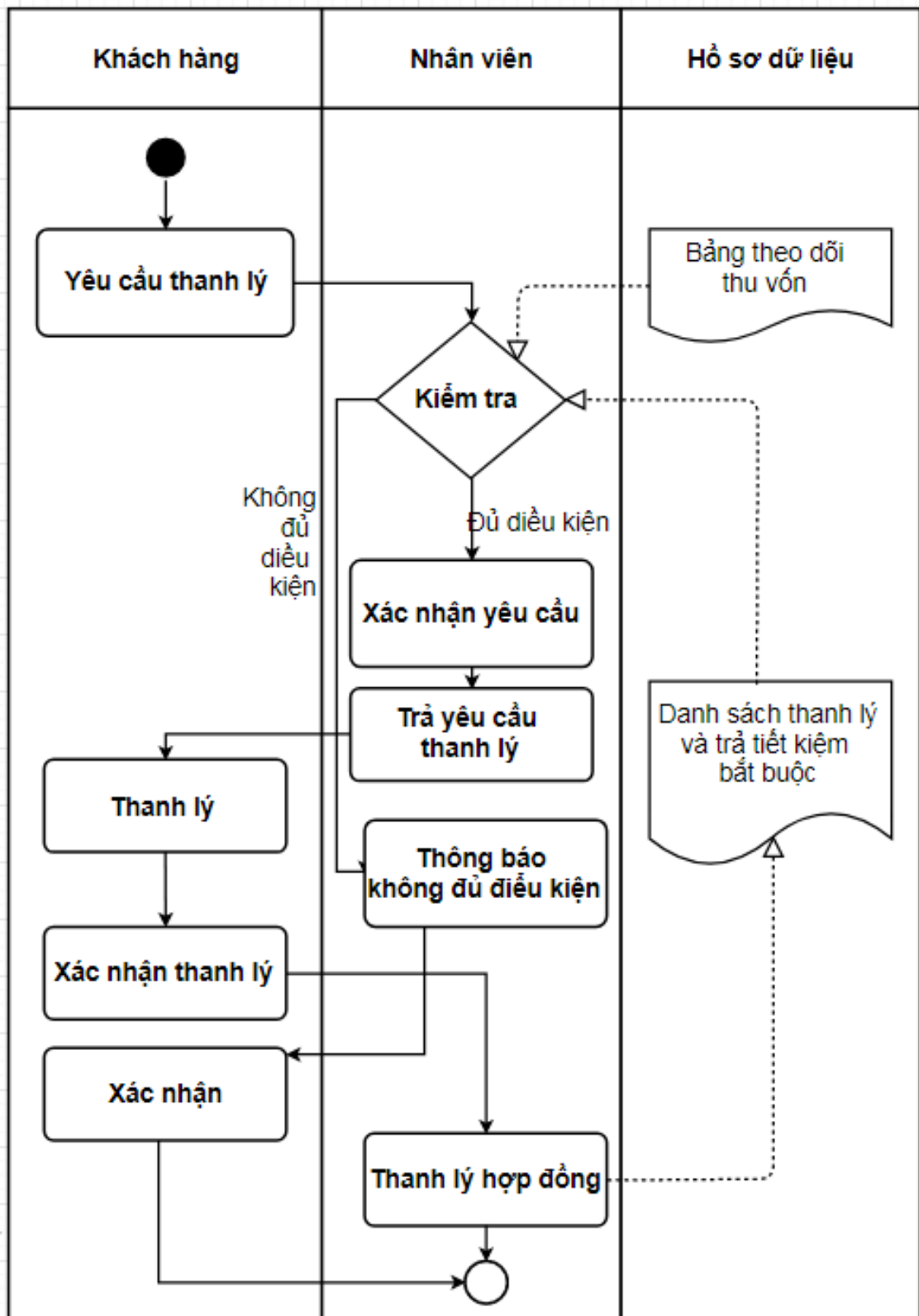
2.2.1 Sơ đồ tiến trình nghiệp vụ thu vốn gốc, lãi



Hình 21: Sơ đồ tiến trình nghiệp vụ thu vốn gốc, lãi

Mô tả: Khách hàng yêu cầu xuất phiếu thu để nộp tiền thì nhân viên sẽ kiểm tra từ bảng theo dõi thu vốn và danh sách được duyệt vay, nếu đủ điều kiện thì nhân viên sẽ xác nhận yêu cầu. Sau đó tạo phiếu thu cho khách hàng. Khách hàng sẽ nộp tiền và xác nhận nộp tiền sau đó nhân viên sẽ xác nhận thu tiền và cập nhật bảng theo dõi thu vốn và kết thúc tiến trình. Ngược lại thì nhân viên sẽ thông báo không đủ điều kiện với khách hàng. Khách hàng sẽ xác nhận và kết thúc tiến trình.

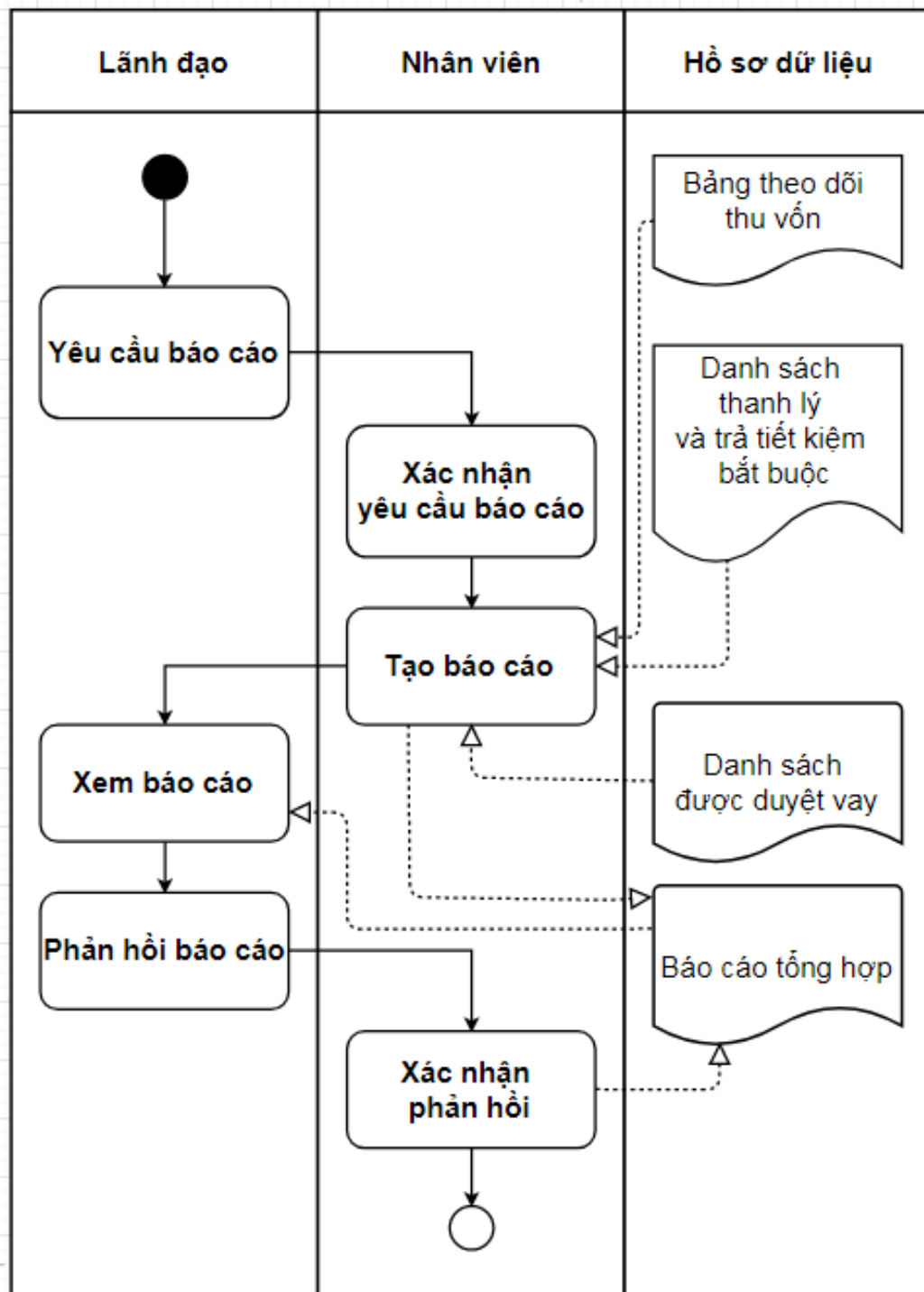
2.2.2 Sơ đồ tiến trình nghiệp vụ thanh lý hợp đồng và trả TKBB



Hình 22: Sơ đồ tiến trình nghiệp vụ thanh lý hợp đồng và trả TKBB

Mô tả: Khách hàng yêu cầu thanh lý, khi đó nhân viên sẽ kiểm tra từ bảng theo dõi thu vốn và danh sách thanh lý và trả tiết kiệm bắt buộc. Nếu đủ điều kiện thì nhân viên sẽ xác nhận yêu cầu. Sau đó trả lời yêu cầu thanh lý của khách hàng. Khách hàng sẽ làm thủ tục thanh lý và xác nhận thanh lý sau đó nhân viên sẽ thanh lý hợp đồng của khách hàng và cập nhật lại danh sách thanh lý và trả tiết kiệm bắt buộc và kết thúc tiến trình. Ngược lại thì nhân viên sẽ thông báo không đủ điều kiện cho khách hàng, khách hàng sẽ xác nhận và kết thúc tiến trình.

2.2.3 Sơ đồ tiến trình nghiệp vụ lập báo cáo tổng hợp



Hình 23: Sơ đồ tiến trình nghiệp vụ lập báo cáo tổng hợp

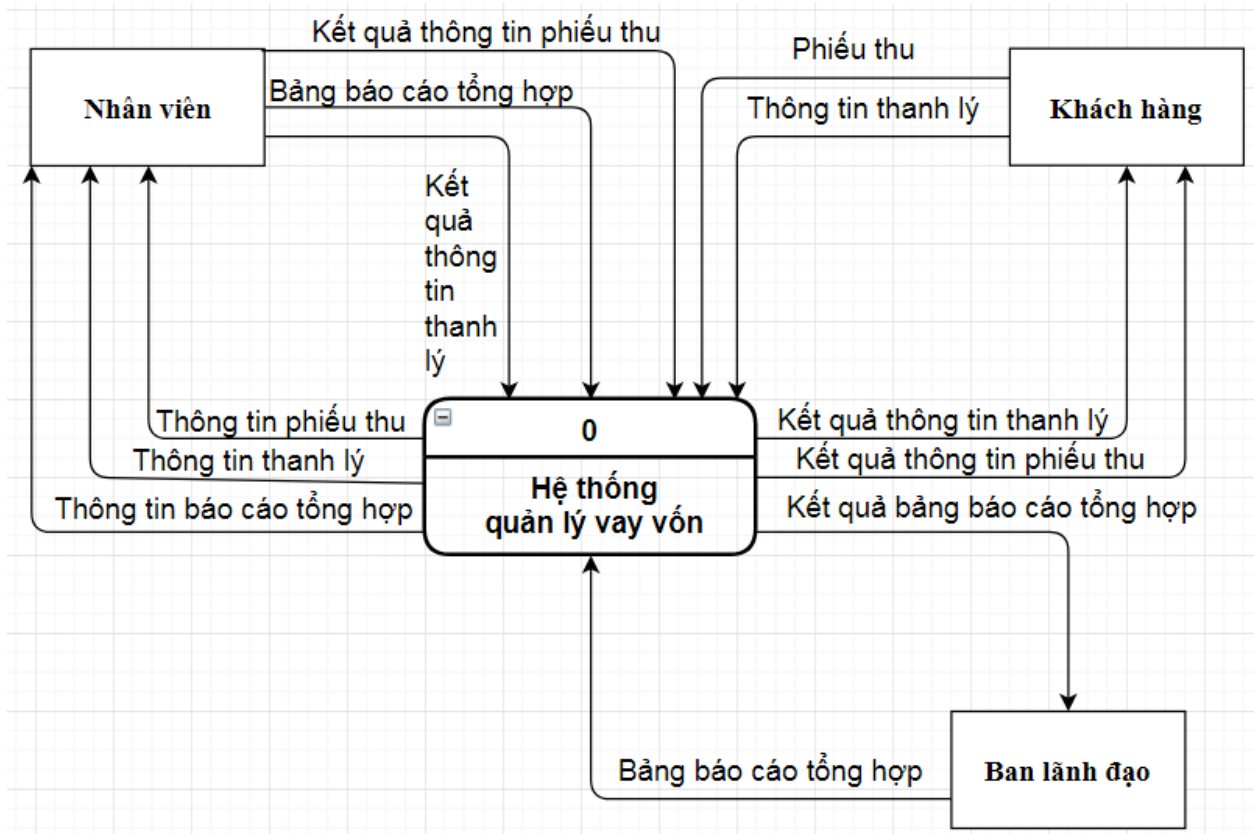
Mô tả: Lãnh đạo yêu cầu báo cáo thì nhân viên sẽ xác nhận yêu cầu của lãnh đạo và tạo báo cáo từ bảng theo dõi thu vốn gốc và danh sách thanh lý và trả tiết kiệm bắt buộc, danh sách được duyệt vay rồi nhập vào bảng báo cáo tổng hợp. Sau đó sẽ trả thông tin báo cáo cho lãnh đạo, lãnh đạo xem và phản hồi báo cáo cho nhân viên. Nhân viên xác nhận phản hồi và cập nhật lại bảng báo cáo tổng hợp cuối cùng kết thúc tiến trình.

CHƯƠNG 3 PHÂN TÍCH THIẾT KẾ HỆ THỐNG

3.1 Biểu đồ nghiệp vụ

3.1.1 Biểu đồ ngữ cảnh

a. Sơ đồ



Hình 24: Biểu đồ ngữ cảnh

b. Mô tả hoạt động

- KHÁCH HÀNG

- Khách hàng sẽ yêu cầu xuất phiếu thu
- Hệ thống sẽ xuất thông tin phiếu thu cho khách hàng
- Khách hàng sẽ yêu cầu thông tin thanh lý
- Hệ thống sẽ trả thông tin thanh lý cho khách hàng

- NHÂN VIÊN

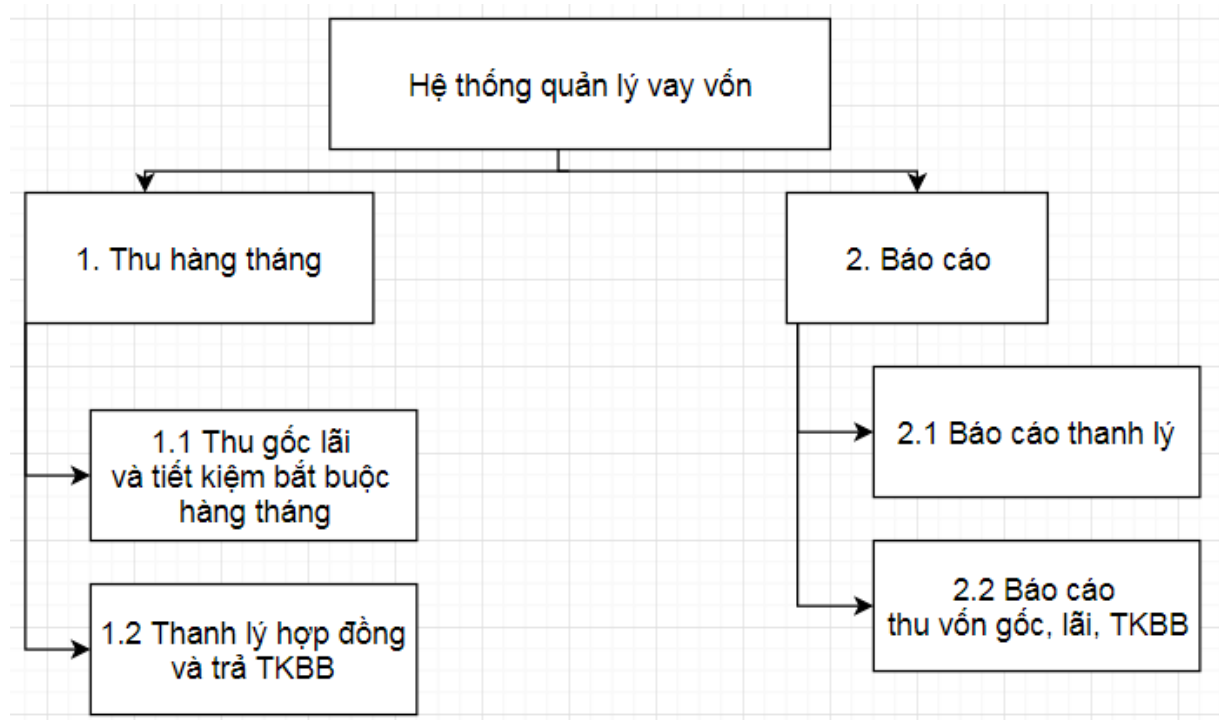
- Hệ thống thông báo xuất thông tin phiếu thu
- Nhân viên gửi kết quả thông tin phiếu thu đến hệ thống
- Hệ thống thông báo xuất thông tin thanh lý
- Nhân viên gửi kết quả thông tin thanh lý đến hệ thống
- Hệ thống thông báo xuất thông tin báo cáo tổng hợp
- Nhân viên gửi bảng báo cáo tổng hợp đến hệ thống

- LÃNH ĐẠO

- Lãnh đạo yêu cầu bảng báo cáo tổng hợp
- Hệ thống gửi kết quả bảng báo cáo tổng hợp cho lãnh đạo

3.1.2 Sơ đồ phân rã chức năng

a. Sơ đồ



Hình 25: Sơ đồ phân rã chức năng

Chú thích:

TKBB là tiết kiệm bắt buộc

b. Mô tả chi tiết

➤ Thu hàng tháng:

Chức năng này cho phép nhân viên thu vốn gốc, lãi hàng tháng và tiền tiết kiệm bắt buộc của khách hàng. Sau đó thanh lý hợp đồng và trả tiết kiệm bắt buộc đối với khách hàng đã trả hết vốn gốc và lãi.

➤ Báo cáo:

Báo cáo chi tiết tình hình vay của khách hàng đến ban lãnh đạo, báo cáo thu lãi, gốc, tiết kiệm bắt buộc của từng tháng, năm đến ban lãnh đạo.

3.1.3 Danh sách các hồ sơ cần sử dụng

- D1. Danh sách được duyệt vay
- D2. Bảng theo dõi thu vốn
- D3. Danh sách thanh lý và trả TKBB
- D4. Bảng báo cáo tổng hợp

Chú thích:

TKBB là tiết kiệm bắt buộc

3.1.4 Ma trận thực thể chức năng

Các thực thể				
D1. Danh sách được duyệt vay				
D2. Bảng theo dõi thu vốn				
D3. Danh sách thanh lý và trả TKBB				
D4. Bảng báo cáo tổng hợp				
Các chức năng	D1	D2	D3	D4
1. Thu hàng tháng	R	U	U	
2. Báo cáo	R	R	R	U

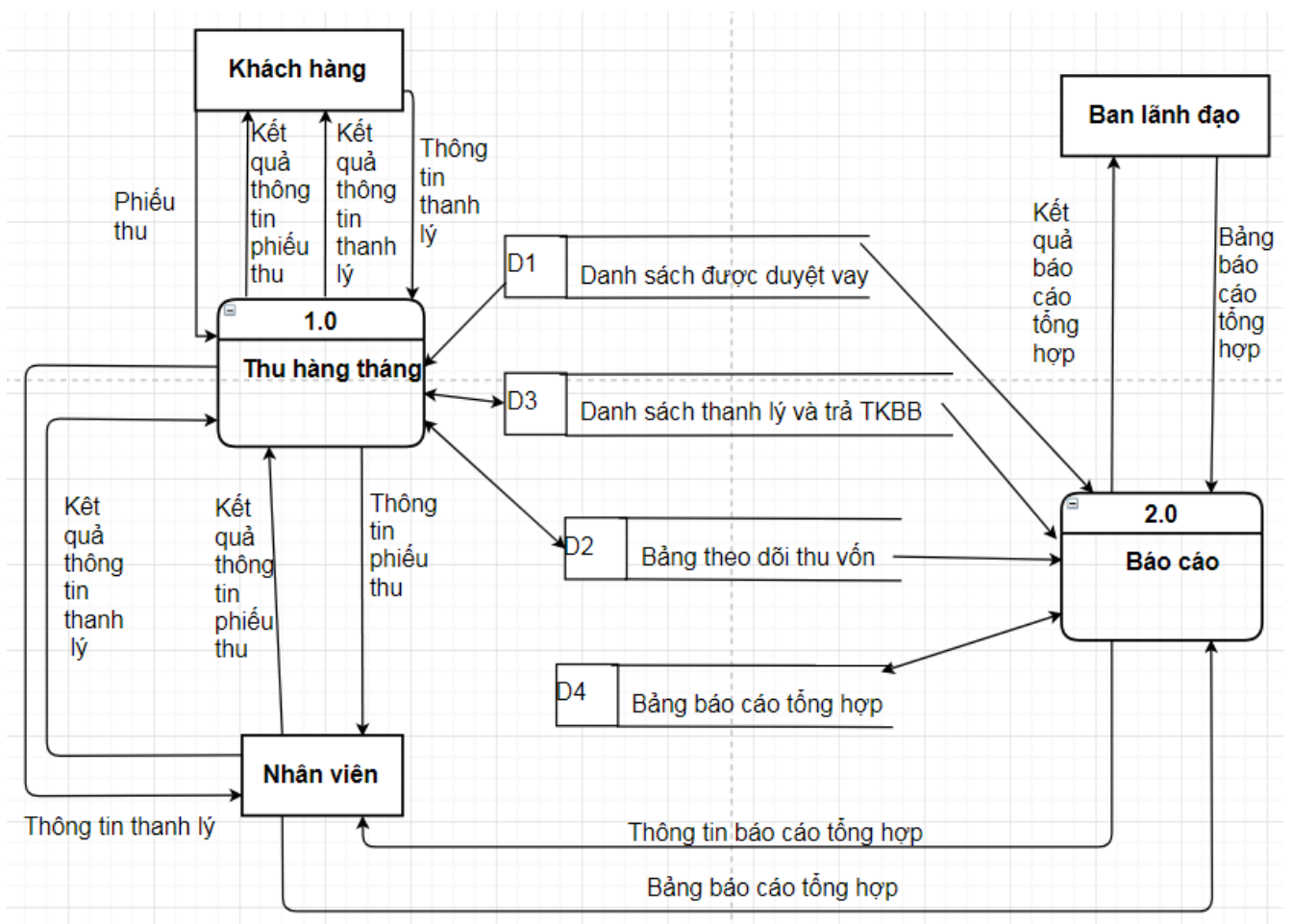
Hình 26: Ma trận thực thể chức năng

Chú thích:

TKBB là tiết kiệm bắt buộc

3.2 Sơ đồ luồng dữ liệu

3.2.1 Sơ đồ luồng dữ liệu mức 0



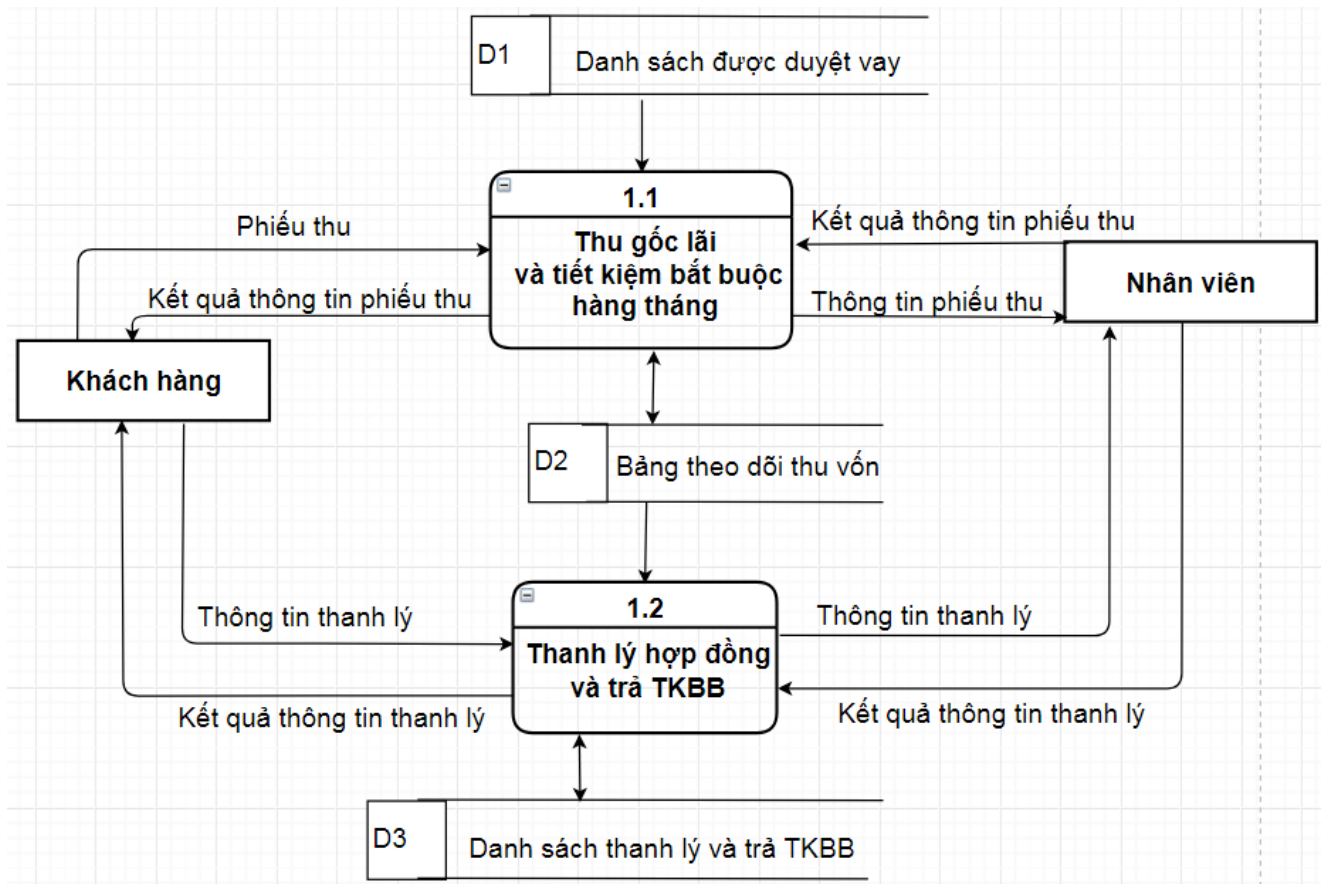
Hình 27: Sơ đồ luồng dữ liệu mức 0

Chú thích:

TKBB là tiết kiệm bắt buộc

3.2.2 Sơ đồ luồng dữ liệu mức 1

3.2.2.1 Sơ đồ luồng dữ liệu mức 1 tiến trình “Thu hàng tháng”

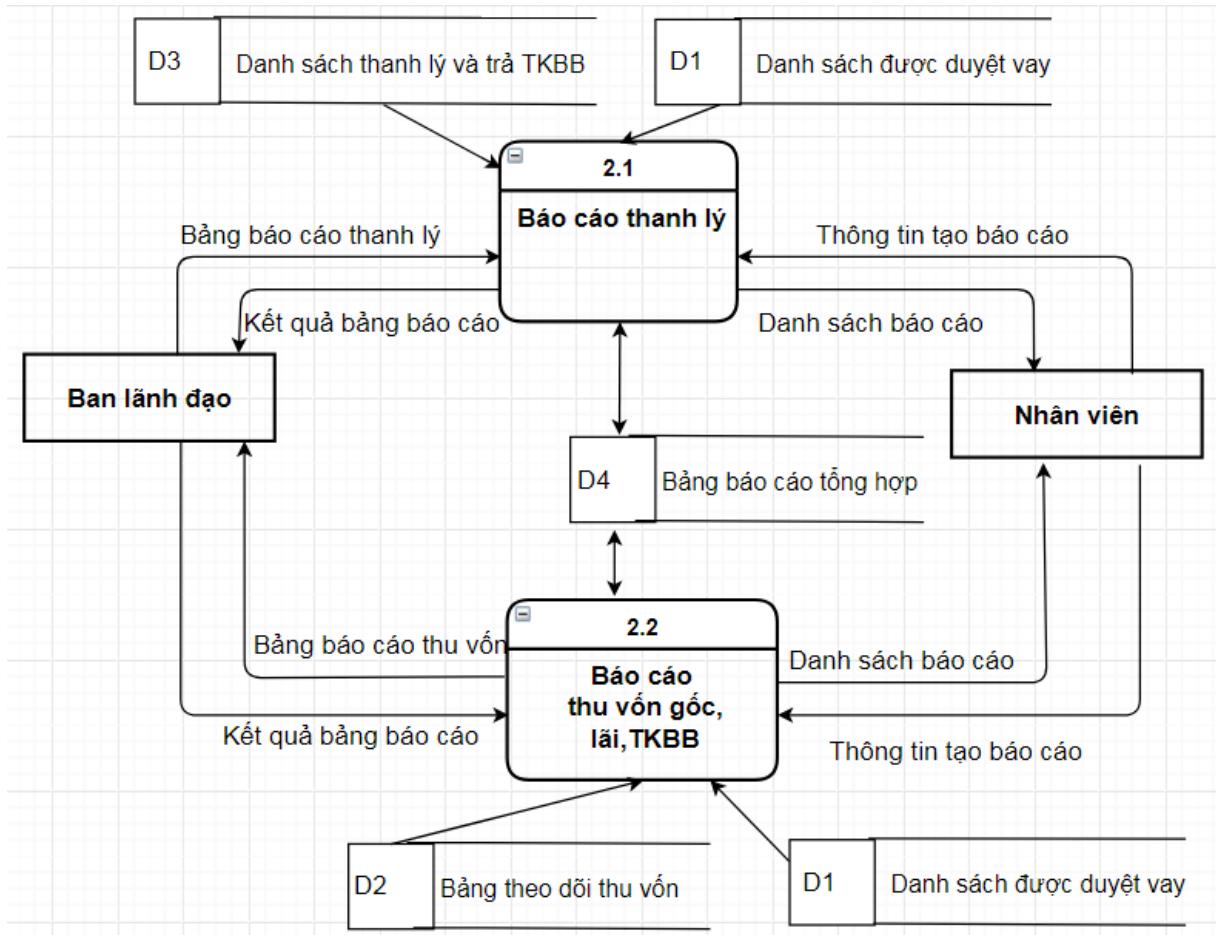


Hình 28: Sơ đồ luồng dữ liệu mức 1 tiến trình “Thu hàng tháng”

Chú thích:

TKBB là tiết kiệm bắt buộc

3.2.2.2 Sơ đồ luồng dữ liệu mức 1 tiến trình “Báo cáo”



Hình 29: Sơ đồ luồng dữ liệu mức 1 tiến trình “Báo cáo”

Chú thích:

TKBB là tiết kiệm bắt buộc

3.3 Thiết kế các bảng cơ sở dữ liệu

3.3.1 Mô hình liên kết thực thể (ER)

3.3.1.1 Các kiểu thực thể

KHÁCH HÀNG có các thuộc tính: mã khách hàng, họ tên khách hàng, nghề nghiệp, giới tính, mức đề nghị vay, mức thu nhập, mục đích, mức vay được duyệt, ngày vay, địa chỉ, thời gian vay

NHÂN VIÊN có các thuộc tính: mã nhân viên, họ tên nhân viên, địa chỉ, chức vụ

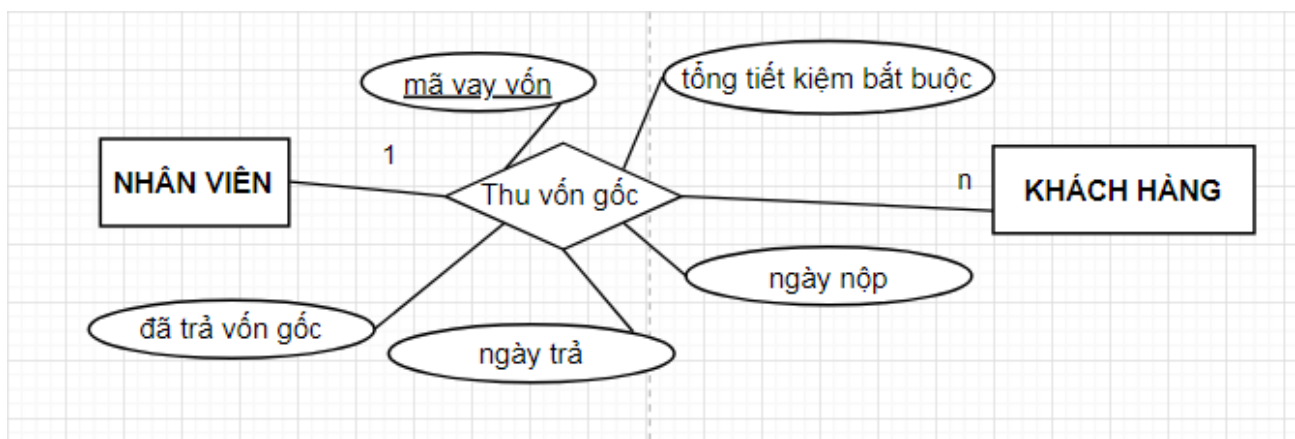
3.3.1.2 Các kiểu liên kết

NHÂN VIÊN “Thu vốn gốc” KHÁCH HÀNG

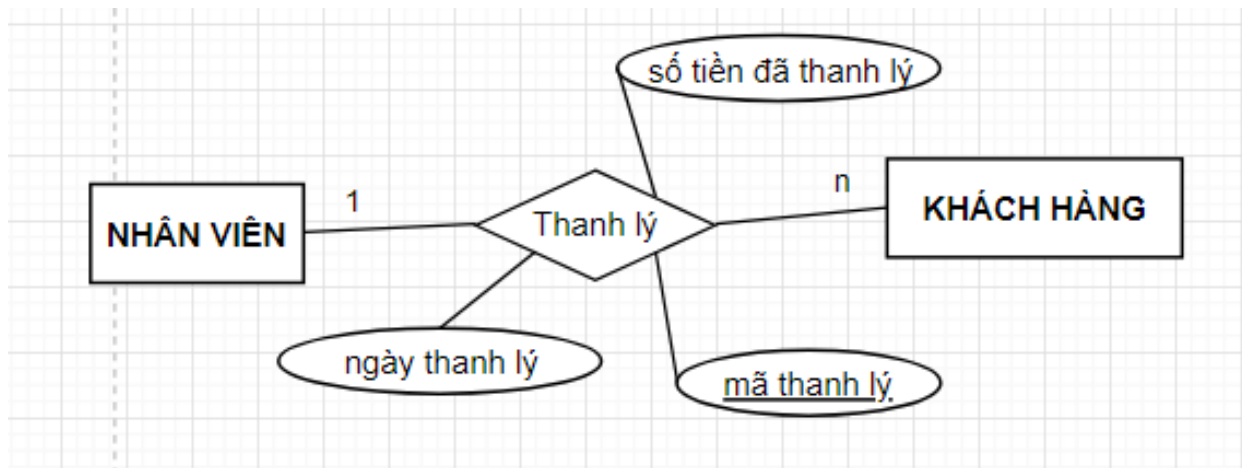
Thông tin riêng của Thu vốn gốc gồm: mã vay vốn, đã trả vốn gốc, ngày trả, ngày nộp, tổng tiết kiệm bắt buộc.

NHÂN VIÊN “ Thanh lý” KHÁCH HÀNG

Thông tin riêng của Thanh lý gồm: mã thanh lý, ngày thanh lý, số tiền đã thanh lý.

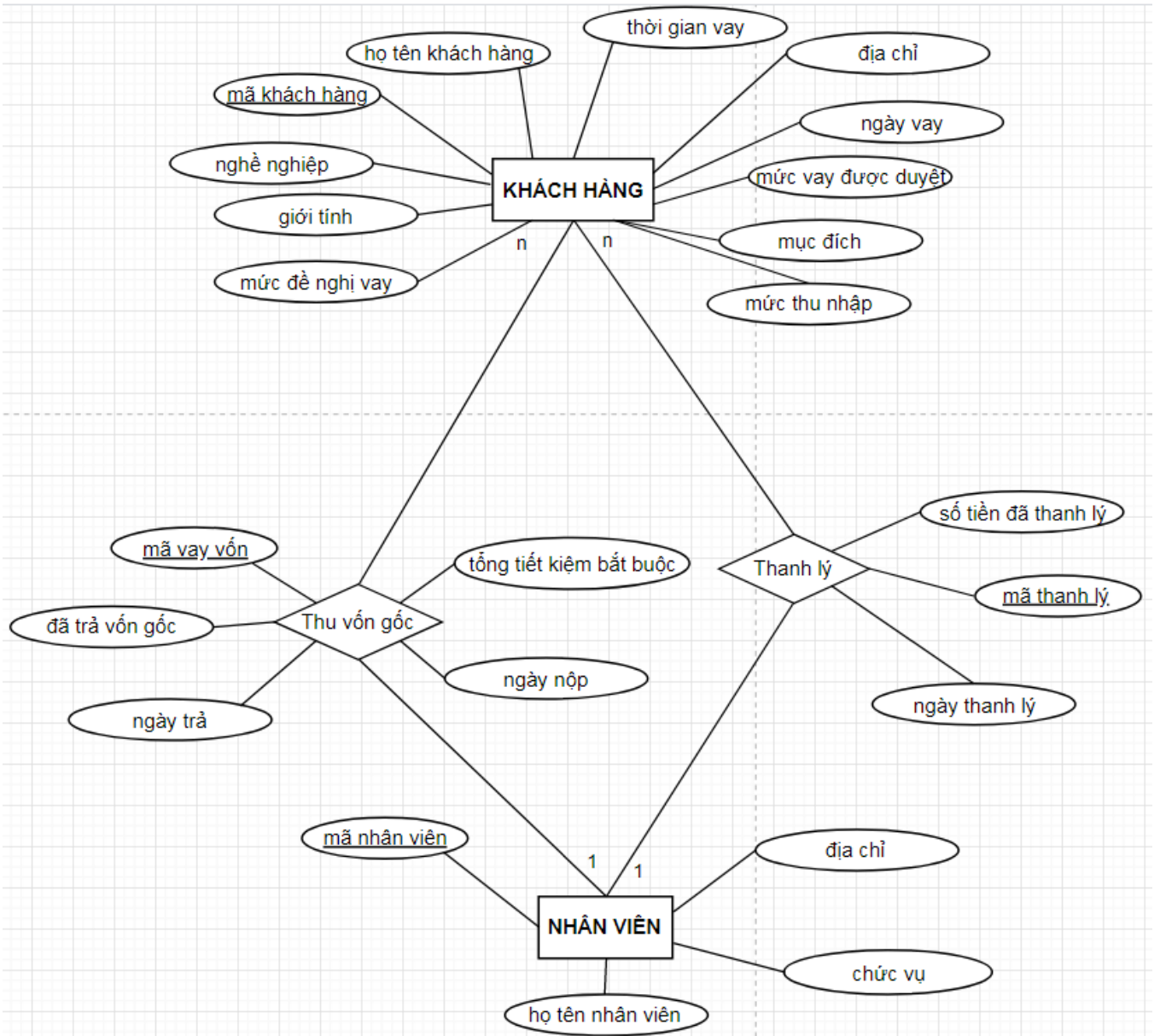


Hình 30: Liên kết **NHÂN VIÊN** thu vốn gốc **KHÁCH HÀNG**



Hình 31: Liên kết **NHÂN VIÊN** thanh lý **KHÁCH HÀNG**

3.3.1.3 Mô hình ER



Hình 32: Mô hình thực thể (ER)

3.3.2 Mô hình quan hệ

3.3.2.1 Áp dụng thuật toán chuyển mô hình quan hệ E-R thành các quan hệ sau

1. Biểu diễn các thực thể

- **KHÁCH HÀNG:** (mã khách hàng, họ tên khách hàng, địa chỉ, thời gian vay, ngày vay, nghề nghiệp, giới tính, mức đề nghị vay, mức thu nhập, mục đích, mức vay được duyệt)

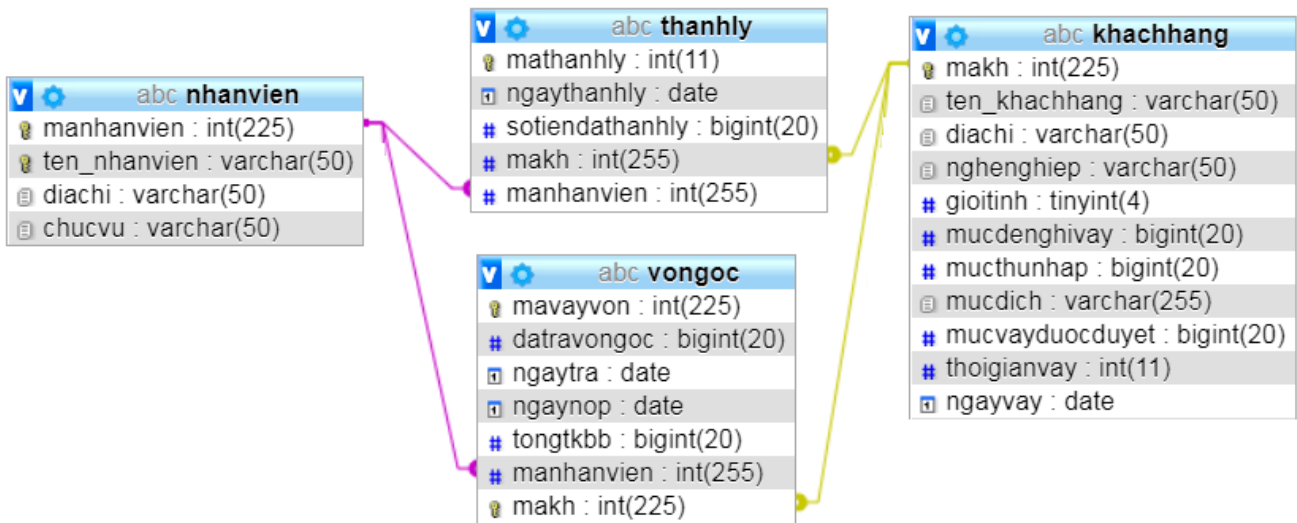
➤ **NHÂN VIÊN** : (mã nhân viên, họ tên nhân viên, địa chỉ, chức vụ)

2. Biểu diễn các mối quan hệ

➤ **VỐN GỐC**: (mã nhân viên, mã khách hàng, mã vay vốn, đã trả vốn gốc, ngày trả, ngày nộp, tổng tiết kiệm bắt buộc)

➤ **THANH LÝ**: (mã nhân viên, mã khách hàng, mã thanh lý, số tiền đã thanh lý, ngày thanh lý)

3.3.2.2 Mô hình quan hệ



Hình 33: Mô hình quan hệ

3.3.3 Các bảng dữ liệu vay lý

3.3.3.1 Bảng KHACHHANG dùng để lưu trữ thông tin nhân viên, có cấu trúc như sau:

STT	Tên trường	Kiểu dữ liệu	Kích cỡ	Ghi chú
1	makh	INT	255	Mã khách hàng
2	ten_khachhang	VARCHAR	50	Họ tên khách hàng
3	diachi	VARCHAR	50	Địa chỉ
4	ngheNghiep	VARCHAR	50	Nghề nghiệp
5	gioitinh	TINYINT	4	Giới tính
6	mucdenghivay	BIGINT	20	Mức đề nghị vay
7	mucthunhap	BIGINT	20	Mức thu nhập
8	mucdich	VARCHAR	255	Mục đích
9	mucvayduocduyet	BIGINT	20	Mức vay được duyệt
10	thoigianvay	INT	11	Thời gian vat
11	ngayvay	DATE		Ngày vay

3.3.3.2 Bảng VONGOC dùng để lưu trữ thông tin vốn gốc đã trả và tiết kiệm bắt buộc hàng tháng

STT	Tên trường	Kiểu dữ liệu	Kích cỡ	Ghi chú
1	mavayvon	INT	255	Mã vay vốn
2	datravongoc	BIGINT	20	Đã trả vốn gốc
3	ngaytra	DATE		Ngày trả
4	ngaynop	DATE		Ngày nộp
5	tongtkbb	BIGINT	20	Tổng tiết kiệm bắt buộc
6	makh	INT	255	Mã khách hàng
7	manhanvien	INT	255	Mã nhân viên

3.3.3.3 Bảng THANHLY dùng để lưu trữ thông tin thanh lý tiền của khách hàng

STT	Tên trường	Kiểu dữ liệu	Kích cỡ	Ghi chú
1	mathanhly	INT	11	Mã thanh lý
2	sotientathanhly	BIGINT	20	Số tiền đã thanh lý
3	ngaythanhly	DATE		Ngày thanh lý
4	makh	INT	255	Mã khách hàng
5	manhanvien	INT	255	Mã nhân viên

3.3.3.4 Bảng NHANVIEN dùng để lưu trữ thông tin nhân viên

STT	Tên trường	Kiểu dữ liệu	Kích cỡ	Ghi chú
1	manhanvien	INT	255	Mã thanh lý
2	ten_nhanvien	VARCHAR	50	Họ tên nhân viên
3	diachi	VARCHAR	50	Địa chỉ
4	chucvu	VARCHAR	50	Chức vụ

CHƯƠNG 4 CÀI ĐẶT CHƯƠNG TRÌNH

Chương này trình bày về hệ thống chương trình, giao diện và các bảng liên quan. Phần của em sẽ tập trung cài đặt về phần thu vốn gốc, lãi vay, tiết kiệm bắt buộc và các báo cáo tổng hợp.

Chương trình hiện nay đã được cài đặt và đang hoạt động tại website: vayvon.congngheso360.net của liên đoàn lao động thành phố Hải Phòng. Vì lý do bảo mật và đang có các dữ liệu, hồ sơ lớn và liên quan đến bản quyền hệ thống nên em không có quyền truy cập vào. Sau đây sẽ chỉ là demo từ chương trình thật.

4.1 Giới thiệu về hệ thống chương trình

Chương trình quản lý vay vốn cho Liên Đoàn Lao Động Thành Phố Hải Phòng với mục tiêu dễ sử dụng, thuận tiện cho việc xử lý. Chương trình viết trên nền tảng web để đáp ứng nhu cầu về thời gian và địa lý. Sau đây sẽ là các phần chi tiết:

4.1.1 Môi trường cài đặt

- Hệ điều hành Windows 7, 8, 10
- Hệ quản trị cơ sở dữ liệu MySQL
- Ngôn ngữ lập trình: PHP, HTML, CSS, Javascript

4.1.2 Các hệ thống con

- Bảng theo dõi nộp tiền của người vay
- Bảng theo dõi thu vốn, lãi và tiết kiệm bắt buộc của người vay vốn.
- Bảng tính thu vốn, lãi vay và tiết kiệm bắt buộc.
- Bảng theo dõi thu vốn và lãi của đơn vị vay vốn.
- Bảng thống kê thu chi - tổng hợp.

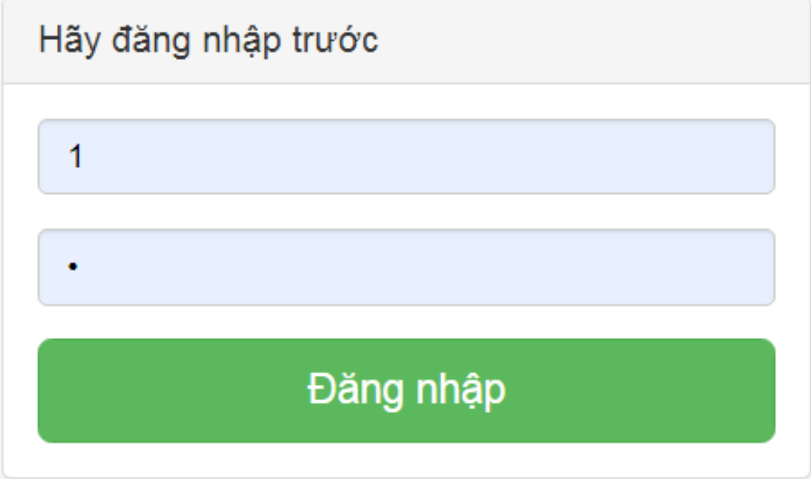
4.1.3 Các chức năng chính của mỗi hệ thống

- Thu hàng tháng
Xem chi tiết thông tin vay vốn của khách hàng.
- Báo cáo
Xuất file excel của từng bảng tổng hợp.

4.2 Giao diện website

4.2.1 Giao diện đăng nhập hệ thống

Để đảm bảo việc an toàn và bảo mật của hệ thống, cần phải có tài khoản thì mới đăng nhập vào và sử dụng hệ thống, mỗi tài khoản có các quyền khác nhau. Tại đây người dùng điền đầy đủ thông tin tài khoản và mật khẩu cũng như chọn vai trò của mình.



Hãy đăng nhập trước

1

.

Đăng nhập

Hình 34: Giao diện đăng nhập hệ thống

Người quản lý chương trình đăng nhập vào phần mềm bằng tài khoản riêng sẽ có quyền xem, sửa, xóa hoặc thêm tất cả các dữ liệu trong trang quản lý.

Sau đây là giao diện trang quản lý (Hệ thống đang chạy của liên đoàn lao động thành phố Hải Phòng):

Hình 35: Giao diện trang quản lý

4.2.2 Các bảng tổng hợp

- Bảng theo dõi nợ tiền của người vay

Excel

Show 10 entries Search:

STT	Ngày vay	Tên khách hàng	Số tiền vay	Đã trả vốn gốc	Hạn nợ tiếp theo	Phải thu vốn gốc	trả vốn gốc hàng tháng	Lãi vay theo kỳ/tháng/năm	Tiết kiệm bắt buộc	Tổng số tiền phải nộp
1	2019-06-03	Đoàn Văn Thọ	20000000	20000000	0	0	0	0	0	0
2	2019-06-03	Nguyễn Văn Vinh	30000000	3000000	2019-08-03	27000000	1000000	162000	300000	1462000
3	2019-06-03	Nguyễn Văn Huy	20000000	0	2019-08-03	20000000	1000000	120000	200000	1320000
4	2019-06-03	Nguyễn Văn Hanh	30000000	0	2019-08-03	30000000	1000000	180000	300000	1480000
5	2019-06-03	Nguyễn Văn Hiếu	30000000	0	2019-08-03	30000000	1000000	180000	300000	1480000

Showing 1 to 5 of 5 entries Previous 1 Next

Hình 36: Bảng theo dõi nợ tiền của người vay

Đơn vị: LIÊN ĐOÀN LAO ĐỘNG THÀNH PHỐ HẢI PHÒNG			Mẫu số: C30-BB	
Bộ phận:			Ban hành theo quy định số 19/2006/QĐ-BTC	
Mã đơn vị SDNS: 1004221			Ngày 30/03/2006 của bộ trưởng BTC	
Người nộp tiền: Nguyễn Văn Hiếu			Số phiếu:	
Khách hàng: Nguyễn Văn Hiếu			Ngày 10 tháng 6 năm 2019	
Lý do thu:				
Số tiền: 1474000				
Giám đốc (Kí và ghi rõ họ tên)			Kế toán (Kí và ghi rõ họ tên)	Thủ quỹ (Kí và ghi rõ họ tên)
			Người nộp phiếu (Kí và ghi rõ họ tên)	Người nộp tiền (Kí và ghi rõ họ tên)

Hình 37: Phiếu thu

- Bảng theo dõi thu vốn, lãi và tiết kiệm bắt buộc của người vay vốn.

Excel

Show 10 entries Search:

STT	Tên khách hàng	Hạn nộp tiếp theo	Số tiền vay	Thời gian vay	Đã trả vốn gốc	trả vốn gốc hàng tháng	Lãi vay theo tháng	Tiết kiệm bắt buộc	Tổng số tiền phải nộp
1	Đoàn Văn Thọ	2019-07-03	20000000	20	20000000	1000000	0	200000	1200000
2	Nguyễn Văn Vinh	2019-07-03	30000000	30	30000000	1000000	162000	300000	1462000
3	Nguyễn Văn Huy	2019-07-03	20000000	20	0	1000000	120000	200000	1320000
4	Nguyễn Văn Hanh	2019-07-03	30000000	30	0	1000000	180000	300000	1480000
5	Nguyễn Văn Hiếu	2019-07-03	30000000	30	0	1000000	180000	300000	1480000

Showing 1 to 5 of 5 entries Previous 1 Next

Hình 38: Bảng theo dõi thu vốn, lãi và tiết kiệm bắt buộc của người vay vốn

- Bảng tính thu vốn, lãi vay và tiết kiệm bắt buộc.

Excel

Show 10 entries Search:

STT	Ngày vay	Tên khách hàng	Số tiền vay	Đã trả vốn gốc	Số dư nợ	Lãi tháng
1	2019-06-03	Đoàn Văn Thọ	20000000	20000000	0	0
2	2019-06-03	Nguyễn Văn Vinh	30000000	30000000	27000000	162000
3	2019-06-03	Nguyễn Văn Huy	20000000	0	20000000	120000
4	2019-06-03	Nguyễn Văn Hanh	30000000	0	30000000	180000
5	2019-06-03	Nguyễn Văn Hiếu	30000000	0	30000000	180000

Showing 1 to 5 of 5 entries Previous 1 Next

Hình 39: Bảng tính thu vốn, lãi vay và tiết kiệm bắt buộc

- Bảng theo dõi thu vốn và lãi của đơn vị vay vốn.

Show 10 entries Search:

STT	Ngày vay	Tên khách hàng	Số tiền vay	Đã trả vốn gốc	Số dư nợ	Vốn gốc phải trả	Lãi tháng	Tiết kiệm bắt buộc	Thu tiền
1	2019-06-03	Đoàn Văn Thọ	20000000	20000000	0	0	0	0	Thu
2	2019-06-03	Nguyễn Văn Vinh	30000000	3000000	27000000	1000000	162000	300000	Thu
3	2019-06-03	Nguyễn Văn Huy	20000000	0	20000000	1000000	120000	200000	Thu
4	2019-06-03	Nguyễn Văn Hanh	30000000	0	30000000	1000000	180000	300000	Thu
5	2019-06-03	Nguyễn Văn Hiếu	30000000	0	30000000	1000000	180000	300000	Thu

Showing 1 to 5 of 5 entries Previous 1 Next

Hình 40: Bảng theo dõi thu vốn và lãi của đơn vị vay vốn.

- Bảng thống kê thu chi - tổng hợp.

[Excel](#) Show 10 entries Search:

STT	Người làm phiếu	Số phiếu	Ngày thu	Thu	Người nộp/nhận tiền	Nhân viên	Khách hàng
1	admin	20	0000-00-00	1200000	Đoàn Văn Thọ	lãnh đạo	Đoàn Văn Thọ
2	admin	19	0000-00-00	1462000	Nguyễn Văn Vinh	lãnh đạo	Nguyễn Văn Vinh
3	admin	18	2019-07-03	1320000	Nguyễn Văn Huy	lãnh đạo	Nguyễn Văn Huy
4	admin	17	2019-07-03	1480000	Nguyễn Văn Hanh	lãnh đạo	Nguyễn Văn Hanh
5	admin	16	2019-07-03	1480000	Nguyễn Văn Hiếu	lãnh đạo	Nguyễn Văn Hiếu

Showing 1 to 5 of 5 entries Previous 1 Next

Hình 41: Bảng thống kê thu chi - tổng hợp.

KẾT LUẬN

Trong đồ án này em đã vận dụng được những kiến thức đã học ở trường và tìm hiểu thêm kiến thức bên ngoài. Kết quả đạt được bao gồm:

- Lý thuyết:
 - Phát biểu và mô tả được nghiệp vụ bài toán.
 - Phân tích cơ sở dữ liệu theo sơ đồ tiến trình nghiệp vụ.
 - Thiết kế giao diện.
- Chương trình:
 - Sử dụng hệ quản trị cơ sở dữ liệu MySQL.
 - Sử dụng ngôn ngữ PHP, HTML, CSS, Javascript để xây dựng chương trình.
 - Hệ thống được cài đặt thử nghiệm với một số dữ liệu chạy thông suốt và cho ra kết quả, đáp ứng được yêu cầu bài toán.

Trong thời gian nghiên cứu, xây dựng chương trình, em đã hết sức cố gắng làm với sự giúp đỡ tận tình của thầy giáo hướng dẫn. Chương trình đã đạt được kết quả nhất định. Tuy nhiên do thời gian ngắn, trình độ và kinh nghiệm còn hạn chế nên chương trình còn nhiều thiếu sót. Rất mong các thầy cô giáo và các bạn giúp đỡ để chương trình ngày càng hoàn thiện.

TÀI LIỆU THAM KHẢO

- [1]. <https://kungfuphp.com>
- [2]. <http://congdoanhaiphong.vn/>
- [3]. <http://vayvon.congnggheso360.net/>
- [4]. <http://youtube.com>
- [5]. Nguyễn Văn Vy, *Giáo trình phân tích thiết kế các hệ thống thông tin*, NXB Giáo dục Việt Nam, 2010.
- [6]. Liên hệ với giáo viên, email: chieudv@hpu.edu.vn.
- [7]. Sách học lập trình php: PHP & MySQL Novice to Ninja, by Tom Butler & Kevin Yank
- [8]. https://vietjack.com/php/tai_lieu_php_tham_khao.jsp
- [9]. <https://vi.wikipedia.org/wiki/PHP>
- [10]. Phạm Hữu Khang – Lập trình web bằng php 5.3 và MySQL 5.1 – NXB Phương Đông.
- [11]. Lê Văn Phùng (2010), *CSDL quan hệ và công nghệ phân tích – thiết kế*, NXB Thông tin và Truyền thông.