

**ĐẠI HỌC THÁI NGUYÊN**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**Lê Thị Bắc**

**PHÂN TÍCH THIẾT KẾ HƯỚNG MẪU  
VÀ ỨNG DỤNG VÀO BÀI TOÁN QUẢN LÝ  
ĐỀ TÀI, DỰ ÁN CỦA SỞ KHOA HỌC VÀ  
CÔNG NGHỆ THÁI NGUYÊN**

**CHUYÊN NGÀNH: KHOA HỌC MÁY TÍNH**  
**MÃ SỐ: 60. 48. 01**

**LUẬN VĂN THẠC SĨ KHOA HỌC MÁY TÍNH**

**NGƯỜI HƯỚNG DẪN KHOA HỌC**  
**PGS.TS. Nguyễn Văn Vy**

**Thái Nguyên – 2012**

## **LỜI CAM ĐOAN**

Tôi xin cam đoan về toàn bộ nội dung của luận văn, những điều được trình bày hoặc là của cá nhân hoặc là được tổng hợp từ nhiều nguồn tài liệu. Tất cả các tài liệu tham khảo đều có xuất xứ rõ ràng và được trích dẫn hợp pháp.

Tôi xin hoàn toàn chịu trách nhiệm và chịu mọi hình thức kỷ luật theo quy định cho lời cam đoan của mình.

Lê Thị Bắc

### BẢNG CÁC CHỮ VIẾT TẮT

Viết tắt	Tên đầy đủ
RUP	Rational Unified Process
OOPSLA	Object-Oriented Programming, Systems, Languages, and Applications conference
PloP	Pattern Languages of Programs
POSA	Pattern-Oriented Software Architecture
POAD	Pattern Oriented Analysis and Design
UML	Unified Modeling Language
GoF	Gang of Four
ĐTDA	Đề tài dự án
KHCN	Khoa học Công nghệ
CNTT	Công nghệ thông tin
UBND	Ủy ban Nhân dân
CSDL	Cơ sở dữ liệu
QLKH	Quản lý khoa học
NCKH	Nghiên cứu khoa học
DM	Danh mục
NSD	Người sử dụng
DL	Dữ liệu
PK	Primary Key
FK	Foreign Key

## DANH SÁCH CÁC BẢNG VÀ HÌNH VẼ

Số	Tên bảng và hình	Trang
Hình 3.1	Sơ đồ tiến trình hoạt động quản lý đề tài, dự án	34
Hình 3.2	Mô hình triển khai hệ thống	40
Hình 4.1	Mô hình ca sử dụng mức tổng thể của hệ thống quản lý đề tài dự án	44
Hình 4.2	Biểu đồ tuần tự hệ thống đăng nhập	56
Hình 4.3	Biểu đồ tuần tự hệ thống Quản trị người sử dụng	57
Hình 4.4	Biểu đồ tuần tự chức năng QL DTDA đang triển khai	59
Hình 4.5	Biểu đồ trình tự thống kê, báo cáo	61
Hình 4.6	Mô hình khái niệm phân tích lĩnh vực	63
Hình 4.7	Biểu đồ cộng tác quản trị người sử dụng	64
Hình 4.8	Biểu đồ cộng tác quản trị danh mục	65
Hình 4.9	Biểu đồ cộng tác quản lý đề tài, dự án đang triển khai	66
Hình 4.10	Biểu đồ cộng tác thống kê báo cáo.	67
Hình 4.11	Các lớp thiết kế cơ bản của hệ thống	69
Hình 4.12	Giao diện chương trình quản lý đề tài dự án	75
Hình 4.13	Danh sách đề tài, dự án	76
Hình 4.14	Danh mục lĩnh vực công nghệ	76
Hình 4.15	Bảng danh mục cán bộ tham gia đề tài	77
Hình 4.16	Chi tiết danh sách đề tài dự án đang triển khai	78

## MỤC LỤC

LỜI CAM ĐOAN .....	1
BẢNG CÁC CHỮ VIẾT TẮT.....	2
DANH SÁCH CÁC BẢNG VÀ HÌNH VẼ.....	3
MỤC LỤC.....	4
LỜI NÓI ĐẦU .....	6
CHƯƠNG 1: TỔNG QUAN VỀ MẪU THIẾT KẾ.....	8
1.1 Lịch sử phát triển mẫu thiết kế.....	8
1.2 Khái niệm về mẫu thiết kế (Design pattern) .....	10
1.3 Hệ thống các mẫu thiết kế và phân loại.....	11
1.4 Phân loại mẫu .....	15
1.5 Lợi ích của việc sử dụng các mẫu trong thiết kế .....	17
1.6 Áp dụng mẫu thiết kế trong phát triển phần mềm.....	21
CHƯƠNG 2 : QUY TRÌNH PHÂN TÍCH VÀ THIẾT KẾ HƯỚNG MẪU .	22
2.1 Các bước của tiến trình phân tích và thiết kế hướng mẫu .....	22
2.2 Phân tích và đặc tả yêu cầu hệ thống.....	26
2.3 Tiến trình sử dụng mẫu thiết kế.....	27
CHƯƠNG 3 : BÀI TOÁN NGHIỆP VỤ VÀ GIẢI PHÁP GIẢI QUYẾT	
VẤN ĐỀ.....	29
3.1 Khảo sát thu thập dữ liệu và mô tả bài toán .....	29
3.2 Mô tả hoạt động nghiệp vụ của hệ thống (mô hình nghiệp vụ).....	29
3.3 Những vấn đề và tồn tại trong hệ thống quản lý đề tài NCKH .....	38
3.4 Giải pháp tổng thể công nghệ thông tin cho bài toán đặt ra. ....	38
3.5 Mô hình triển khai .....	40
CHƯƠNG 4. PHÂN TÍCH VÀ THIẾT KẾ BÀI TOÁN ĐỊNH HƯỚNG	
MẪU.....	42
4.1 Phát triển mô hình nghiệp vụ.....	42

4.2 Mô hình ca sử dụng: .....	44
4.3 Phân tích hệ thống .....	56
4.4 Mô hình khái niệm phân tích lĩnh vực:.....	63
4.5 Thiết kế hệ thống : .....	64
4.6 Bảng dữ liệu: .....	70
4.7 Cài đặt và thử nghiệm một số modul .....	75

## LỜI NÓI ĐẦU

Phát triển phần mềm theo định hướng đối tượng ngày càng phát triển mạnh mẽ và đang chiếm ưu thế do những đặc trưng vượt trội của nó. Trong toàn bộ tiến trình phát triển phần mềm, phân tích thiết kế vẫn là một khâu khó khăn, phức tạp nhất và đòi hỏi người thực hiện có trình độ cao, có nhiều kinh nghiệm. Chất lượng của phần mềm đạt được phụ thuộc chủ yếu ở khâu này, tức là phụ thuộc vào chất lượng thiết kế. Tuân thủ theo quy trình RUP, sau một quá trình phát triển sẽ ta nhận được một thiết kế hướng đối tượng của hệ thống. Có một số tiêu chí về thiết kế tốt cho phép người ta xem xét nó và hoàn thiện. Nhưng một cách khác để hoàn thiện thiết kế thường được áp dụng, đó là xem xét thiết kế để cải tiến nó trên cơ sở các kiến thức về các mẫu thiết kế (design patterns). Các mẫu thiết kế là các giải pháp đã được các nhà thiết kế có kinh nghiệm nghiên cứu và hoàn thiện cho những vấn đề thường gặp trong thiết kế.

Một cách làm triệt để hơn để sử dụng lại các mẫu cho thiết kế là phân tích thiết kế định hướng mẫu. Đây là một trong ba hướng sử dụng lại của phát triển phần mềm hướng đối tượng – sử dụng lại các mẫu. Với mong muốn áp dụng các công nghệ mới cho phát triển phần mềm, tôi đã chọn đề tài “*Phân tích thiết kế hướng mẫu và ứng dụng cho bài toán quản lý đề tài, dự án của sở Khoa học và Công nghệ Thái Nguyên*” làm đề tài của luận văn.

Theo phương pháp phân tích thiết kế định hướng mẫu, người ta sử dụng các mẫu thiết kế ngay sau khi đặc tả yêu cầu. Như vậy, sau khi đặc tả yêu cầu của bài toán theo phương pháp hướng đối tượng, ta phải tìm kiếm các mẫu tương ứng cho mỗi đặc tả chi tiết. Khó khăn lớn ở đây là có rất nhiều mẫu khác nhau, làm sao chọn được một mẫu thích hợp. Hơn nữa, các đặc trưng mô tả mẫu là tương đối trừu tượng, có sự khác biệt đáng kể với các đặc trưng đặc tả yêu cầu. Vì thế đòi hỏi người phát triển hệ thống có hiểu biết sâu sắc về mỗi mẫu, nắm chắc được yêu cầu của vấn đề đặt ra, để từ đó chọn ra một mẫu giải quyết được yêu cầu của vấn đề. Mặt khác, cùng một yêu cầu, có thể có nhiều mẫu có khả năng đáp ứng được yêu cầu đó. Đây lại là một cách lựa

chọn đòi hỏi phải có kinh nghiệm từ thực tiễn triển khai ứng dụng. Theo phương pháp này, ta đã bỏ qua được các bước đi tuần tự, từ mức cao đến mức chi tiết của giai đoạn phân tích và thiết kế, thường tốn nhiều thời gian công sức với nhiều mô hình phương pháp khác nhau. Vì vậy đây là cách làm hiệu quả, vừa tiết kiệm thời gian, công sức và vẫn cho phép nhận được một thiết kế tốt. Mặc dù hướng này là rất khó khăn, với mong muốn thử nghiệm công nghệ và nâng cao kỹ năng phân tích thiết kế, tôi chọn nó để giải quyết bài toán đặt ra.



## CHƯƠNG 1: TỔNG QUAN VỀ MẪU THIẾT KẾ

### 1.1 Lịch sử phát triển mẫu thiết kế

Sự xuất hiện của mẫu xuất phát từ rất nhiều sáng kiến khác nhau. Kiến trúc sư Christopher Alexander, giáo sư kiến trúc tại trường đại học California ở Berkeley, đã phát triển nền tảng cho các Mẫu. Từ «Mẫu» (Pattern) có liên quan hầu hết tới toàn bộ công việc của ông. Ông và nhóm nghiên cứu của mình đã sử dụng trên 20 năm cho việc phát triển một cách tiếp cận đến kiến trúc lớn bằng cách dùng các Mẫu. Alexander đã mô tả trên 250 mẫu qua một hệ quan điểm trừu tượng rất rộng, từ các kiến trúc của thị trấn đến các thiết kế phòng. Ông đã tìm ra một khuôn mẫu mô tả các yếu tố cơ bản của Pattern, đó là *Giải pháp-Vấn đề-Ngữ cảnh*. Ông đã viết một cuốn sách về các Mẫu kiến trúc [4].

Kent Beck and Ward Cunningham đã rất nhiệt tình trong việc áp dụng các ý tưởng của Alexander vào việc phát triển phần mềm. Họ đã viết bộ Pattern đầu tiên về giao diện người dùng.

Ấn phẩm đầu tiên trình bày về việc dùng các Pattern trong phát triển phần mềm là luận án tiến sĩ năm 1991 của Erich Gamma, được viết ở Đức, lúc đó tác phẩm này chưa phổ biến. “*Gần một nửa các Mẫu Pattern được mô tả sau này [3] được chứng minh là trước đó có trong luận văn tiến sĩ của ông*” [5].

Bruce Anderson là một trong những người đi đầu trong nghiên cứu về Pattern. Ông có một cuộc hội thảo về chủ đề Pattern ở OOPSLA (Object-Oriented Programming, Systems, Languages, and Applications conference) vào khoảng những năm 1990; **Jim ?** đã mô tả các thành ngữ bằng ngôn ngữ C++ trong cuốn sách của ông có tựa đề là “*Advanced C++ Programming*

*Styles and Idioms*”. Các thành ngữ đó dù cách này hay cách khác đều liên quan đến các ý tưởng của các giải pháp cho các vấn đề thường xuyên xảy ra. Một nhóm được gọi là Hillside Group được thành lập để khám phá thêm về các ý tưởng này và đẩy mạnh việc dùng các Mẫu trong việc phát triển phần mềm. Họ đã làm việc để chỉ đạo và hỗ trợ cho các thành viên mới trong cộng đồng *Pattern*. Nhóm này đã thành lập nên Ngôn ngữ mẫu cho chương trình (Pattern Languages of Programs- PloP) lần đầu tiên vào năm 1994 tại một hội nghị. Tri thức chung về mẫu đã được thể hiện tốt trong một cuốn sách gồm 4 người cùng tham gia viết, *Design Patterns: Elements of Object-Oriented Software* [3], họ đã phân loại và mô tả rất rõ 23 mẫu thiết kế được dùng rất phổ biến trong lập trình hướng đối tượng.

Peter Coad đã làm việc với các Mẫu hướng đối tượng từ rất sớm [6]. Ông đã mô tả 7 mẫu đơn giản trong phân tích và thiết kế. Ông đã làm việc trên các Mẫu hỗ trợ phân tích miền ứng dụng và dùng công nghệ hướng đối tượng để xây dựng các ứng dụng [7]. Douglas Schmidt cũng là một trong những người đầu tiên nghiên cứu về mẫu; ông là tác giả của nhiều Mẫu trong các hệ thống giao tiếp và các ứng dụng phân tán [8]. Wolfgang Pree đã làm việc trên các mẫu dành cho việc phát triển khung làm việc (framework) [9]. Ông đã phân nhóm các nguyên lý mang tính cấu trúc thành các Siêu Mẫu (MetaPattern) được dùng để phát triển các khung làm việc. *Pattern-Oriented Software: A Pattern System*, cũng được coi là cuốn sách “Tóp 5” [5], chú trọng vào việc dùng các Mẫu ở mức kiến trúc của phát triển phần mềm. Các tác giả đã phân loại các Mẫu phần mềm như là các Mẫu kiến trúc, các mẫu thiết kế và các Thành ngữ. Phần lớn sự đóng góp của họ đều được hướng tới các Mẫu kiến trúc. Cuốn sách của họ cùng với cuốn sách của bốn tác giả *Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides* là những tài liệu rất tốt cho những người mới bắt đầu nghiên cứu mẫu. Quyển 2 của tập sách *Pattern-*