

บทที่ 1

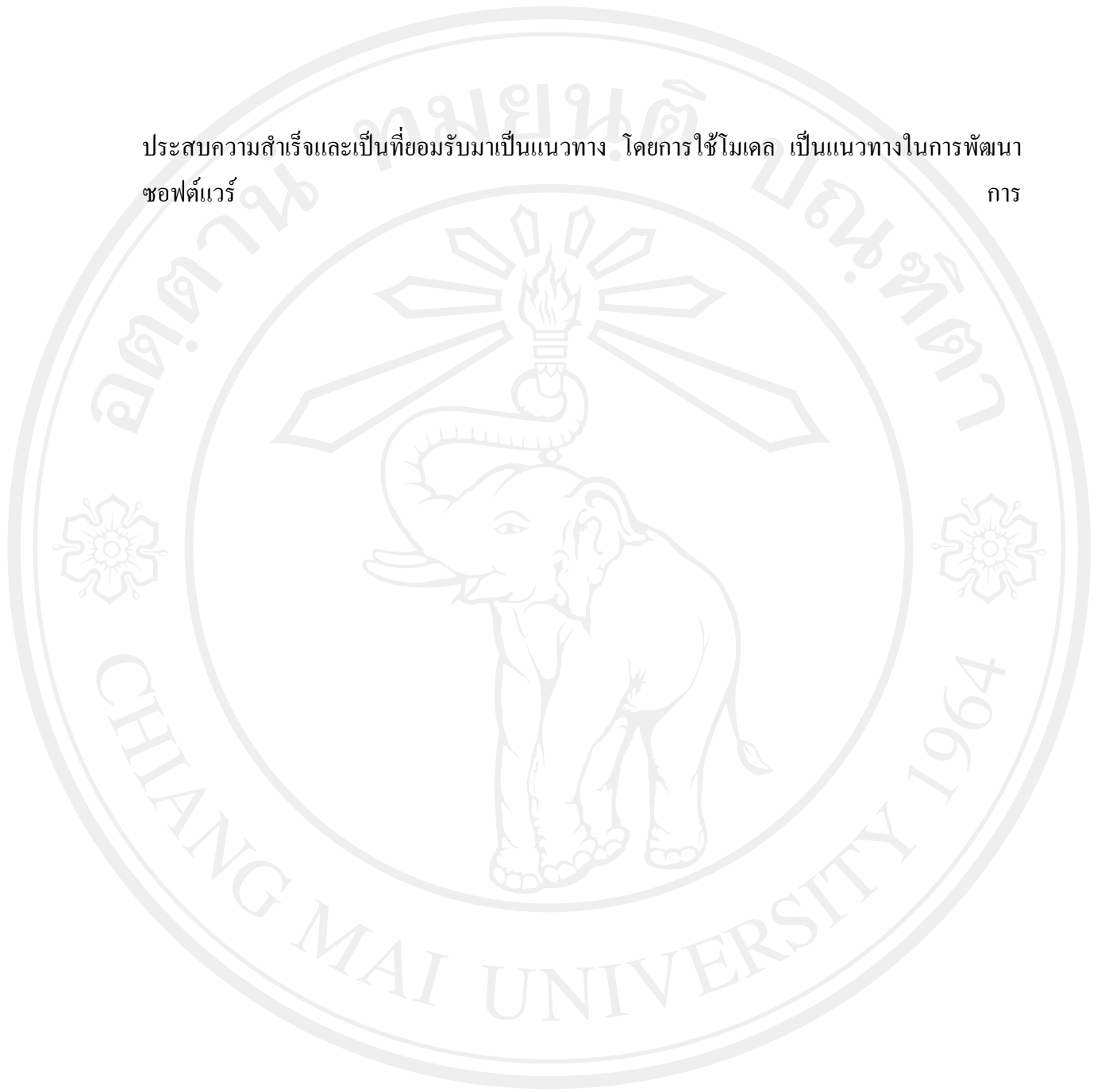
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ปัจจุบันเทคโนโลยีคอมพิวเตอร์มีบทบาทสำคัญกับธุรกิจต่างๆ เป็นอย่างมาก เทคโนโลยีคอมพิวเตอร์ทำให้กระบวนการทำงานรวดเร็วถูกต้องและมีประสิทธิภาพมากขึ้น โดยมีซอฟต์แวร์เป็นส่วนสำคัญสำหรับช่วยบริหารจัดการงานต่างๆ ภายในธุรกิจ เช่นการเก็บข้อมูล การสืบค้น การติดตาม รวมไปถึงการประมวลผลและรายงานต่างๆ แต่การนำเอาซอฟต์แวร์เข้าไปใช้นั้นไม่ใช่เรื่องง่าย ปกติแล้วกระบวนการทางธุรกิจของแต่ละธุรกิจที่มีลักษณะใกล้เคียงกัน แต่จะแตกต่างกันไม่มากนัก ส่วนใหญ่ขั้นตอนการบริหารจัดการต่างๆ จะเป็นลักษณะแบบเดียวกัน แต่ในรายละเอียดของแต่ละธุรกิจยังคงแตกต่างกันจะไม่เหมือนกันทั้งหมด เช่นการติดตามงานวิจัยของหน่วยงานย่อยๆ ภายในสถาบันวิจัยและพัฒนาพื้นที่สูง จะมีการทำงานที่เหมือนกัน แต่อาจจะมีบางสิ่งที่มีความแตกต่างกันไป ซึ่งรายละเอียดที่แตกต่างกันนั้น จำเป็นต้องได้รับการปรับปรุงแก้ไข เพื่อให้สามารถทำงานกับแต่ละหน่วยงาน การนำเอาซอฟต์แวร์มาใช้ จึงจำเป็นต้องปรับแต่งซอฟต์แวร์ให้ตรงตามความต้องการและกระบวนการทำงานของแต่ละธุรกิจให้ได้มากที่สุด กระบวนการทางวิศวกรรมซอฟต์แวร์ จึงเป็นสิ่งสำคัญอย่างมากเพื่อนำมาแก้ปัญหาเหล่านี้ และช่วยทำให้เกิดกระบวนการพัฒนาอย่างมีแบบแผน ซอฟต์แวร์ที่ได้มีประสิทธิภาพ หากแต่ในปัจจุบันนี้ หลายธุรกิจที่มีขนาดเล็กไม่สามารถพัฒนาซอฟต์แวร์ภายในธุรกิจของตัวเองได้ เพราะต้องใช้เวลาและงบประมาณจำนวนมาก ซอฟต์แวร์สำเร็จ ที่หาได้ง่ายและราคาถูก จึงเป็นอีกทางเลือกหนึ่งที่องค์กรขนาดเล็กเหล่านี้ นำมาใช้กับระบบงานของตัวเอง แต่ซอฟต์แวร์เหล่านี้ก็ยังคงไม่สามารถตอบสนองความต้องการภายในธุรกิจได้ทั้งหมด

ปัญหาเรื่องของซอฟต์แวร์ ที่ไม่สามารถตอบสนองความต้องการภายในธุรกิจได้ทั้งหมดนี้ สามารถแก้ไขได้ โดยการพัฒนาซอฟต์แวร์ให้เหมาะสมกับแต่ละหน่วยงาน ซึ่งต้องนำเอากระบวนการพัฒนาที่สามารถพัฒนาได้รวดเร็วและประหยัดงบประมาณไปใช้ในการพัฒนา และอยู่ภายใต้งบประมาณที่สามารถรับได้ ซึ่งในปัจจุบันการพัฒนาซอฟต์แวร์โดยใช้สถาปัตยกรรมการขับเคลื่อนด้วยแบบจำลอง หรือเอ็มดีเอ (MDA: Model-Driven Architecture) เป็นกระบวนการพัฒนาซอฟต์แวร์ที่เน้นความเร็ว โดยการนำเอาแบบโครงสร้างการพัฒนาซอฟต์แวร์ที่

ประสบความสำเร็จและเป็นที่ยอมรับมาเป็นแนวทาง โดยการใช้โมเดล เป็นแนวทางในการพัฒนา
ซอฟต์แวร์ การ



ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright © by Chiang Mai University
All rights reserved

พัฒนาแบบเอ็มดีเอ จะสร้างโมเดลขึ้นมาก่อน ซึ่งโมเดลที่สร้างขึ้นนี้ มาจากพื้นฐานโครงสร้างของแต่ละธุรกิจ เมื่อพัฒนาซอฟต์แวร์โดยใช้เอ็มดีเอให้กับธุรกิจหนึ่งไปแล้ว ก็สามารถเอาโมเดลเหล่านั้นมาประยุกต์ใช้ได้อีก เพียงแต่จะต้องปรับปรุงรายละเอียดเพิ่มเติม เพื่อให้สามารถนำไปใช้กับแต่ละธุรกิจได้จริง

1.2 วัตถุประสงค์ของการศึกษา

- 1) เพื่อศึกษาการพัฒนาซอฟต์แวร์โดยใช้สถาปัตยกรรมการขับเคลื่อนด้วยโมเดลหรือเอ็มดีเอ
- 2) เพื่อศึกษาการใช้เครื่องมือสำหรับการพัฒนาซอฟต์แวร์โดยใช้เอ็มดีเอบนพื้นฐานของเอสพีคอทเน็ต เวอร์ชัน 4.0 (ASP .Net 4.0)
- 3) เพื่อให้ได้ต้นแบบซอฟต์แวร์ระบบติดตามงานวิจัย

1.3 ประโยชน์ที่จะได้รับจากการศึกษาเชิงทฤษฎีและ / หรือเชิงประยุกต์

- 1) เป็นต้นแบบให้นักพัฒนาสามารถนำข้อมูลไปประกอบการตัดสินใจสำหรับการพัฒนาซอฟต์แวร์ด้วยเอ็มดีเอ
- 2) เป็นแนวทางหนึ่งของการบวนการพัฒนาซอฟต์แวร์ที่แสดงให้เห็นถึงกระบวนการพัฒนาที่รวดเร็วโดยใช้เครื่องมือเข้ามาช่วย ซึ่งขึ้นอยู่กับลักษณะของแต่ละธุรกิจ
- 3) เพื่อให้ได้ต้นแบบระบบติดตามงานวิจัย
- 4) เพื่อให้ได้ต้นแบบโมเดลที่สามารถนำไปใช้กับระบบติดตามงานวิจัยของหน่วยงานอื่นๆได้ด้วย

1.4 ขอบเขตของการค้นคว้าแบบอิสระ

1.4.1 ขอบเขตของการศึกษาและพัฒนาระบบงาน

ศึกษาปัญหาความต้องการของระบบและพัฒนาโปรแกรมต้นแบบระบบติดตามงานวิจัย โดยใช้การวิเคราะห์และออกแบบระบบด้วยกระบวนการของเอ็มดีเอ มีขอบเขตการศึกษาดังนี้

- 1) ศึกษาแนวทางการพัฒนาซอฟต์แวร์ด้วยเอ็มดีเอ
- 2) ศึกษาการออกแบบและพัฒนาโดยใช้เครื่องมือของเอ็มดีเอ
- 3) ใช้กรณีศึกษาระบบติดตามงานวิจัย โดยพัฒนาบางส่วนของระบบเท่านั้น
- 4) ศึกษาวิเคราะห์ความต้องการของต้นแบบระบบติดตามงานวิจัย และนำเสนอออกมาในรูปแบบของยูสเคส (Use Case)

- 5) วิเคราะห์และออกแบบโมเดลของต้นแบบระบบติดตามงานวิจัย โดยได้โมเดลต้นแบบที่เป็นกลางสามารถนำไปใช้กับระบบอื่นได้
- 6) แต่ละส่วนของระบบสามารถทำงานเบื้องต้นได้ตามยูสเคสที่ได้ออกแบบไว้
- 7) โปรแกรมสามารถทำงานภายใต้เทคโนโลยีเอเอสพีคอตเน็ตเวอร์ชัน 4.0 (ASP .Net 4.0)

1.4.2 ขอบเขตของการพัฒนาตามมาตรฐานไอเอสโอ 29110 (ISO 29110)

- 1) กระบวนการด้านการบริหารโครงการ
 - การวางแผนโครงการ (Project Planning)
 - การดำเนินโครงการตามแผน (Project Plan Execution)
 - การประเมินและควบคุมโครงการ (Project Assessment and Control)
 - การปิดโครงการ (Project Closure)
- 2) กระบวนการด้านการสร้างซอฟต์แวร์
 - การเริ่มสร้างซอฟต์แวร์ (Software Implementation Initiation)
 - การวิเคราะห์ความต้องการของซอฟต์แวร์ (Software Requirement Analysis)
 - การออกแบบสถาปัตยกรรมและรายละเอียดของซอฟต์แวร์ (Software Architecture and Detailed Design)
 - การสร้างซอฟต์แวร์ (Software Construction)
 - การบูรณาการและการทดสอบซอฟต์แวร์ (Software Integration and Tests)
 - การส่งมอบซอฟต์แวร์ (Product Delivery)