บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของงานวิจัย

ในกระบวนการพัฒนาซอฟต์แวร์ส่วนใหญ่ นักพัฒนาให้ความสำคัญในการพัฒนาแอพพลิเคชั่นในส่วนของความต้องการที่เป็นฟังก์ชันหลักของระบบ (Functional Requirement) เป็นลำดับแรก ส่วนความต้องการที่ไม่ใช่เชิงฟังก์ชันหลักของระบบ (Non-Functional Requirement) โดยเฉพาะอย่างยิ่งความปลอดภัย (Security) จะถูกนำไปพัฒนาเพิ่มเติมหลังจากที่มีการพัฒนาแอพพลิเคชั่นเสร็จเรียบร้อยแล้ว ซึ่งจากเหตุการณ์ดังกล่าวทำให้เกิดปัญหาในด้านค่าใช้จ่ายของการพัฒนาที่เพิ่มขึ้น ถ้าแอพพลิเคชั่นนั้นมีข้อบกพร่องเกิดขึ้นในระหว่างการพัฒนา (Devanbu & Stubbleine, 2000) จึงมีงานวิจัยที่เสนอแนวทางในการนำความต้องการที่ไม่ใช่เชิงฟังก์ชันหลักของระบบ (Cortellessa, Marco & Inverardi, 2007; Jurjens, 2002; Lodderstedt, Basin & Dorser, 2002; Nakamura, Tatsubori, Imamura & Ono, 2005) อย่างเช่นความน่าเชื่อถือ (Reliability), ประสิทธิภาพ (Performance) รวมถึงความปลอดภัย (Security) มารวมไว้ในขั้นตอนการวิเคราะห์และออกแบบระบบ เพื่อที่จะช่วยลดข้อผิดพลาดและค่าใช้จ่ายในการพัฒนาแอพพลิเคชั่น นอกจากนี้การเพิ่มขอบเขตของขั้นตอนการวิเคราะห์และออกแบบให้ครอบคลุมความต้องการที่ไม่ใช่เชิงฟังก์ชันหลักของระบบ จะเป็นการเพิ่มความสามารถในการผลิต (Productivity) และความสามารถในการดูแลรักษา (Maintainability) ด้วย

ในขณะที่แนวคิดหรือกระบวนการในการพัฒนาซอฟต์แวร์มีการพัฒนาไปอย่างต่อเนื่อง และมีความก้าวหน้าไปในระดับที่นักพัฒนาโปรแกรมสามารถสร้างแบบจำลองการออกแบบระบบ (Software Model) เป็นแผนภาพ (Diagram) และสามารถสร้างโค้ดโปรแกรมของซอฟต์แวร์ได้แบบอัตโนมัติ ทำให้ลดระยะเวลาในการพัฒนาโปรแกรมลงไปอย่างมาก กระบวนการพัฒนาซอฟต์แวร์แบบนี้เรียกว่า Model-Driven Architecture (MDA) (Miller & Mukerji, 2003) ซึ่งเป็นแนวคิดในการใช้แบบจำลองเป็นศูนย์กลางในการขับเคลื่อนกระบวนการพัฒนาซอฟต์แวร์ โดยมีการแบ่งแบบจำลองออกเป็นหลายระดับ แบบจำลองในแต่ละระดับจะมีการกำหนดรายละเอียดที่ใช้อธิบายถึงระบบที่ต้องการพัฒนาแตกต่างกัน ซึ่งในการช่วงวิเคราะห์และออกแบบระบบ นักวิเคราะห์และออกแบบระบบจะสร้างแบบจำลองที่อิสระจากแพลตฟอร์ม (Platform Indenpendent Model: PIM) สำหรับใช้อธิบายถึงการทำงานของระบบ

โดย PIM เป็นแบบจำลองที่ยังไม่มีการกำหนดแพลตฟอร์ม (Platform) ที่ต้องการเลือกใช้ และโดยส่วนใหญ่แบบจำลอง PIM จะถูกอธิบายโดยใช้ภาษา UML (Unified Modeling Language) ซึ่งในขั้นตอนนี้แบบจำลอง PIM จะใช้อธิบายในส่วนของฟังก์ชันหลักของระบบ ส่วนความต้องการที่ไม่ใช่เป็นฟังก์ชันหลัก โดยเฉพาะเรื่องความปลอดภัยยังไม่ได้ถูกเพิ่มเติมเข้ามา

จากปัญหาดังกล่าวงานวิจัยนี้จึงเสนอแนวทางในการเพิ่มคุณสมบัติความปลอดภัยในแบบจำลอง PIM ในระหว่างการวิเคราะห์และออกแบบระบบ โดยกลไกในการเพิ่มคุณสมบัติความปลอดภัยในแบบจำลองทำได้โดยการเพิ่มขยาย (Extension) แบบจำลองความปลอดภัยในเมตาโมเดล (Metamodel) ของ UML แล้วให้นักวิเคราะห์และออกแบบนำเมตาโมเดลดังกล่าวไปใช้ในการสร้างแบบจำลอง PIM ที่มีคุณสมบัติความปลอดภัย และเสนอแนวทางในการแปลงแบบจำลอง (Model Transformation) จากแบบจำลอง PIM ที่มีคุณสมบัติความปลอดภัย ให้เป็นแบบจำลองที่เฉพาะเจาะจงกับแพลตฟอร์ม (Platform Specific Model: PSM) ซึ่งเมื่อนำแบบจำลอง PSM นี้ พัฒนาเป็นซอร์สโค้ดก็จะมีคุณสมบัติความปลอดภัยครบตามความต้องการที่ระบุในแบบจำลอง PIM ซึ่งข้อดีอีกประการหนึ่งในการสร้างแบบจำลอง PIM ที่มีคุณสมบัติความปลอดภัยคือช่วยให้นักวิเคราะห์ที่ไม่ได้เป็นผู้เชี่ยวชาญด้านความปลอดภัย (Security Expert) สามารถนำความต้องการด้านความปลอดภัยมากำหนดในแบบจำลองได้ง่ายขึ้น (Rodriguez, Fernandez-Medina & Piattini, 2006)

1.2 วัตถุประสงค์ของงานวิจัย

ในงานวิจัยฉบับนี้ มีวัตถุประสงค์ในการค้นคว้าวิจัยดังนี้

1. เพื่อศึกษาและกำหนดวิธีการในการนำความต้องการของระบบที่เป็นความต้องการที่ไม่ใช่เชิงฟังก์ชันของระบบ (Non-Functional Requirement) ไปขยายในแบบจำลองการออกแบบที่อิสระจากแพลตฟอร์ม (Platform Independent Model: PIM) ให้รองรับความต้องการที่ไม่ใช่เชิงฟังก์ชันของระบบได้ โดยในงานวิจัยนี้ได้เน้นไปที่ความต้องการด้านความปลอดภัย

2. เพื่อวิจัยและพัฒนาวิธีการในการแปลงแบบจำลอง PIM ที่มีคุณสมบัติด้านความปลอดภัยสร้างเป็นแบบจำลองที่เฉพาะเจาะจงกับเพลตฟอร์ม (Platform Specific Model: PSM) ที่มีคุณสมบัติความปลอดภัย และพร้อมที่จะนำไปสร้างเป็นซอร์สโค้ดต่อไป

1.3 ขอบเขตของงานวิจัย

งานวิจัยนี้มุ่งเน้นในการศึกษาและกำหนดวิธีการในการเพิ่มแบบคุณสมบัติความปลอดภัยในแปลงแบบจำลองการออกแบบที่อิสระจากแพลตฟอร์มและวิธีการในการแปลงแบบจำลองการออกแบบดังกล่าวเป็นแบบจำลองที่เฉพาะเจาะจงกับแพลตฟอร์ม โดยมีขอบเขตการวิจัย ดังนี้

1. ศึกษาและสร้างแบบจำลองการออกแบบให้มีคุณสมบัติความปลอดภัย โดยครอบคลุมความปลอดภัยเรื่อง การพิสูจน์ตัวจริง (Authentication) การรักษาความลับ (Confidentiality) และการรักษาความสมบูรณ์ (Integrity)

2. กำหนดวิธีการแปลงแบบจำลองการออกแบบที่อิสระจากแพลตฟอร์มที่มีคุณสมบัติด้านความปลอดภัย และได้แบบจำลองที่เฉพาะเจาะจงกับแพลตฟอร์ม เพื่อที่จะนำแบบจำลองดังกล่าวสร้างโค้ดโปรแกรมที่เพิ่มคุณภาพด้านความปลอดภัยให้กับซอฟต์แวร์ได้

3. การตรวจสอบแนวทางที่นำเสนอในงานวิจัยนี้ ทำได้โดยการพัฒนาแอพพลิเคชั่นกรณีศึกษาเพื่อสร้างแอพพลิเคชั่นที่เฉพาะเจาะจงกับแพลตฟอร์ม 2 แพลตฟอร์ม คือแอพพลิเคชั่นเว็บเซอร์วิส (Web Service) และใช้การรักษาความปลอดภัยบนเว็บเซอร์วิส (Web Service Security: WS-Security) เพื่อแทนคุณสมบัติความปลอดภัย และอีกแพลตฟอร์มคือแอพพลิเคชั่นจาวาเซิร์ฟเล็ต (Java Servlet) ที่มีการรักษาความปลอดภัยตามข้อกำหนดของ J2EE