

บทที่ 7

บทสรุปผลการวิจัย

7.1. บทสรุปผลการวิจัย

วิทยานิพนธ์นี้ ได้นำเสนอวิธีการแปลงอ็อบเจกต์โมเดลที่สามารถปรับเปลี่ยนได้หรือเอไอเอ็ม ซึ่งประกอบด้วยดีไซน์แพทเทิร์นต่างๆ ได้แก่ ไทป์อ็อบเจกต์แพทเทิร์น พรอบเพอร์ตีแพทเทิร์น ไทป์สแควร์แพทเทิร์น และ แอ็คเคาน์ทะเลบิลิตีแพทเทิร์น ซึ่งยากต่อการทำความเข้าใจและบำรุงรักษาให้อยู่ในรูปของยูนิตโมเดล เพื่อช่วยให้การพัฒนาทำได้ง่ายขึ้นและเพิ่มความสามารถในด้านการบำรุงรักษาซอฟต์แวร์ โดยทดลองด้วยการพัฒนาแอปพลิเคชันขึ้นมาจำนวน 3 ระบบที่เป็นแบบเอไอเอ็มเดิมและแอปพลิเคชันอีก 3 ระบบที่ใช้วิธีการแปลงเป็นยูนิตโมเดลที่นำเสนอ แล้วนำมาเปรียบเทียบความสามารถในการทำความเข้าใจ ด้วยการวัดค่าความซับซ้อนของแผนภาพคลาส

ผลการทดลองพบว่า การแปลงอ็อบเจกต์โมเดลแบบเอไอเอ็มให้อยู่ในรูปของยูนิตโมเดลนั้น ความซับซ้อนของแผนภาพคลาสนั้นลดลงเล็กน้อยหรือในบางระบบก็อาจจะไม่ลดลงเลย ซึ่งอาจหมายความว่า การแก้ปัญหาของโมเดลที่เป็นแบบไดนามิก ไม่สามารถแก้ปัญหได้ด้วยโมเดลที่เป็นสแตติก

อย่างไรก็ตาม ด้วยคุณสมบัติที่ได้จากการแปลงให้อยู่ในรูปของยูนิตโมเดลนั้น ทำให้สถาปัตยกรรมของระบบมีภาพที่เป็นระดับสูง สามารถแบ่งเป็นระดับชั้น สามารถแยกส่วนแต่ละคอมโพเนนต์ได้อย่างชัดเจน ช่วยซ่อนรายละเอียดที่ไม่จำเป็นต้องทราบ สิ่งเหล่านี้จะช่วยในเรื่องของความสามารถในการทำความเข้าใจระบบแก่นักพัฒนา รวมทั้งทำให้ผู้ออกแบบสถาปัตยกรรมของซอฟต์แวร์สามารถตรวจสอบความถูกต้องได้ง่ายกว่าการใช้แพ็คเกจ เนื่องจากยูนิตโมเดลมีภาษาที่เป็นทางการและเครื่องมือที่ช่วยในการตรวจสอบความถูกต้อง

นอกจากวิธีการแปลงที่ได้นำเสนอไปแล้วนั้น ผลผลิตที่ได้เพิ่มเติมคือ ได้ยูนิตโมเดลแบบเอไอเอ็มที่สามารถนำกลับไปใช้งานในการพัฒนาระบบอื่นๆได้ ซึ่งยูนิตโมเดลดังกล่าวมีทั้งส่วนประกอบของแพ็คเกจซิกเนเจอร์ อะตอมยูนิตและคอมพาวด์ยูนิต ที่สามารถนำไปใช้ได้ทันทีหรือไปประยุกต์ใช้ด้วยวิธีการออกแบบแบบอื่นๆได้ โดยมียูนิตดังต่อไปนี้ `au_aom.entitytype`, `au_aom.operation`, `au_aom.relationship`, `au_aom.entity` และ `cu_aom.entity` ยูนิตเหล่านี้เมื่อนำไปใช้ ผู้ออกแบบสามารถนำไปเพิ่มความสามารถให้กับเอนทิตีของโดเมนได้โดยการสืบทอด ทำให้แผนภาพคลาสรักษาเอนทิตีที่เป็นตัวแทนของโดเมนไว้ ซึ่งจะช่วยให้นักพัฒนาสามารถทำความเข้าใจได้ง่าย นอกจากนั้นยังช่วยลดปริมาณคลาสจำนวนมากที่อาจจะเกิดขึ้นเมื่อต้องการใช้แพทเทิร์นเดิมซ้ำๆกันหลายครั้งด้วย

จากผลผลิตของยูนิตโมเดลแบบเอไอเอ็ม งานวิจัยได้พัฒนาเครื่องมือที่ช่วยในการออกแบบและสร้างแอปพลิเคชันแบบเอไอเอ็ม ซึ่งภายในเครื่องมือได้เตรียมส่วนประกอบต่างๆของยูนิตโมเดลไว้ให้พร้อมต่อการนำไปใช้งานได้ทันที ซึ่งจะช่วยลดเวลาในการพัฒนาระบบลงไปได้

7.2. ปัญหาและอุปสรรค

- 7.2.1. ภาษาที่ใช้งานการสร้างยูนิตนั้นมีข้อจำกัดที่ไม่สนับสนุนการทำ Reflection ในภาษาจาวา ทำให้การพัฒนาทำได้ยากขึ้น
- 7.2.2. สัญลักษณ์ที่ใช้ในการสร้างยูนิตยังไม่เป็นข้อตกลงที่เป็นมาตรฐานในการแสดง ทำให้แผนภาพยูนิตโมเดลนั้นแตกต่างกันไปตามภาษาโปรแกรมต่างๆ และแม้ว่าโดยหลักการของยูนิตเหมือนกัน แต่ภาษาที่ใช้ในการออกแบบยูนิตและพัฒนาเป็นโปรแกรมขึ้นมาได้นั้น ยังติดอยู่กับภาษาโปรแกรมมิ่ง ทำให้การเปลี่ยนจากภาษายูนิตหนึ่งเป็นอีกภาษาต้องใช้เวลาเรียนรู้มาก

7.3. ข้อจำกัด

- 7.3.1. เนื่องจากปัญหาของภาษายูนิตที่ไม่สนับสนุนการทำ Reflection ทำให้การอิมพลิเมนต์ในส่วนของการเรียกใช้เมทอดในเอไอเอ็มต้องแก้ปัญหาเฉพาะหน้าเป็นกรณีๆไป
- 7.3.2. เครื่องมือที่ใช้สร้างยูนิตโมเดลนั้นยังไม่สนับสนุนให้สามารถแสดงแผนภาพยูนิตโมเดล เหมือนแผนภาพยูเอ็มแอล

7.4. ข้อเสนอแนะ

- 7.4.1. ความซับซ้อนของเอไอเอ็มนั้นไม่ได้อยู่ที่ โครงสร้างของแผนภาพคลาสเท่านั้น แต่ยังมีความซับซ้อนอีกและเป็นปัญหาอีกอย่างหนึ่งก็คือ ความสัมพันธ์ระหว่างอินสแตนซ์ต่างๆ พรอบเพอร์ตี เมทอด ที่จะต้องไม่ซ้ำซ้อนและขัดแย้งกัน ซึ่งยากที่จะตรวจสอบโมเดลตอนรันไทม์ ซึ่งน่าจะมีการศึกษาและหาแนวทางในการช่วยแก้ปัญหาต่อไป