หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6

บทสรุปและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงบทสรุปและข้อเสนอแนะเกี่ยวกับงานวิจัย ที่ได้จากการออกแบบและพัฒนาเครื่องมือวัดซอฟต์แวร์ซึ่งใช้เทคนิคการวัดแบบฟังก์ชันพอยต์เชิงวัตถุ ดังรายละเอียดต่อไปนี้

6.1 บทสรุป

ผู้วิจัยได้พัฒนาเครื่องมือวัดซอฟต์แวร์เชิงวัตถุ ซึ่งใช้เทคนิคการวัดแบบฟังก์ชันพอยต์เชิงวัตถุ โดยมีแผนภาพคลาสซึ่งเป็นข้อกำหนดความต้องการของซอฟต์แวร์(Software Requirement Specification) เป็นข้อมูลเข้า ทำให้สามารถวัดขนาดของซอฟต์แวร์เชิงวัตถุได้ตั้งแต่ขั้นตอนการวิเคราะห์และออกแบบระบบ วิธีการคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุ ได้จากการรวบรวมผลงานวิจัยของ Caldiera Antoniol และ Lokan [4] ซึ่งได้เสนอแนะวิธีการวัดขนาดซอฟต์แวร์เชิงวัตถุโดยใช้เทคนิคดังกล่าว การวิเคราะห์และออกแบบเครื่องมือวัดในงานวิจัยนี้ใช้ภาษายูเอ็มแอลเป็นเครื่องมือหลักในการแสดงให้เห็นถึงแนวทางการออกแบบระบบ การทำงานของเครื่องมือวัดใช้สถาปัตยกรรมแบบไคลเอนต์เซอร์ฟเวอร์ โดยที่ส่วนโปรแกรมไคลเอนต์ทำหน้าที่ติดต่อกับผู้ใช้เพื่อรองรับคำสั่งการทำงานทั้งหมด เช่นการทำหน้าที่เป็นโปรแกรมประมวลผลคำ การคอมไพล์และประมวลผลโปรแกรมภาษาจาวา คำสั่งสร้างข้อมูลภาษาเอโอแอล และคำสั่งการคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุ เป็นต้น และส่วนเซอร์ฟเวอร์ได้แก่เครื่องให้บริการเว็บซึ่งทำหน้าที่แปลงไฟล์ต้นฉบับภาษาจาวาให้เป็นภาษาเอโอแอล (Abstract Object Language: AOL) เพื่อใช้ในการคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุ ผู้วิจัยได้พัฒนาเครื่องมือวัดตามที่ได้ออกแบบไว้ซึ่งสามารถคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุซึ่งเป็นเป้าหมายหลัก จากนั้นจึงได้ทำการทดสอบความสามารถของเครื่องมือวัด ซึ่งก็สามารถวัดขนาดของซอฟต์แวร์ได้ตามที่ออกแบบไว้

ผู้วิจัยสามารถสรุปประโยชน์ของเครื่องมือมาตรวัดซอฟต์แวร์ที่พัฒนาขึ้น ได้ดังต่อไปนี้

1) สามารถใช้เป็นเครื่องมือในการวัดขนาดของซอฟต์แวร์เชิงวัตถุ

เมื่อผู้ใช้สร้างแผนภาพคลาสด้วยเครื่องมือช่วยในการออกแบบซอฟต์แวร์เชิงวัตถุและใช้งานร่วมกับเครื่องมือวัดซอฟต์แวร์ที่พัฒนาขึ้นในงานวิจัย เพื่อคำนวณหาจำนวนของฟังก์ชันพอยต์เชิงวัตถุ ผู้ใช้สามารถทราบขนาดของซอฟต์แวร์ที่กำลังจะพัฒนาได้ตั้งแต่ขั้นตอนการวิเคราะห์และออกแบบระบบ ทำให้การวางแผนจัดสรรทรัพยากรที่มีอยู่จำกัดเป็นไปอย่างเหมาะสม

2) สามารถใช้เป็นเครื่องมือติดตามความก้าวหน้าในการพัฒนาซอฟต์แวร์

เครื่องมือวัดสามารถคำนวณหาจำนวนของฟังก์ชันพอยต์เชิงวัตถุจากโค้ดต้นฉบับ ดังนั้นในระหว่างการพัฒนาซอฟต์แวร์ผู้บริหารโครงงานหรือนักพัฒนาสามารถทราบจำนวนฟังก์ชันพอยต์ที่พัฒนาไปแล้วได้ โดยการผ่านโค้ดต้นฉบับที่ได้พัฒนาเสร็จแล้วเป็นข้อมูลเข้าสำหรับเครื่องมือวัด

3) สามารถใช้เป็นเครื่องมือในการเลือกปรับปรุงคลาสที่มีขนาดใหญ่

ถ้าผู้พัฒนาซอฟต์แวร์ได้มีการกำหนดขนาดของคลาส เช่นจำนวนฟังก์ชันพอยต์เชิงวัตถุของคลาสหนึ่งๆ จะต้องมีค่าไม่เกิน 40 ฟังก์ชันพอยต์ เป็นต้น หลังจากใช้เครื่องมือวัดเพื่อดูจำนวนของฟังก์ชันพอยต์แล้วพบว่ามีขนาดเกินกว่าที่ได้กำหนดไว้ ผู้พัฒนาก็สามารถพิจารณาปรับปรุงให้มีขนาดให้เล็กลงได้ซึ่งทำให้ลดความซับซ้อนลง เช่นแยกคลาสหรือโมดูลออกเป็นโมดูลย่อย หรือลดไฟล์ต้นฉบับที่ไม่มีความจำเป็นออกจากโปรแกรม เป็นต้น

6.2 ข้อเสนอแนะ

ผู้วิจัยมีข้อเสนอแนะสำหรับงานวิจัยเกี่ยวกับฟังก์ชันพอยต์เชิงวัตถุ ดังนี้

1) ในงานวิจัยนี้ได้ใช้เพียงแผนภาพคลาสเป็นข้อมูลเข้าของเครื่องมือวัด ซึ่งแผนภาพคลาสเป็นเพียงแผนภาพเชิงสถิตย์ ในภาษายูเอ็มแอล ซึ่งแสดงการมีอยู่ของคลาสต่างๆและความสัมพันธ์ของคลาสในระบบเท่านั้น โดยไม่แสดงถึงกิจกรรมที่เกิดขึ้นในระบบแต่อย่างใด ดังนั้นแนวทางการทำวิจัยต่อไปจึงอาจเป็นการนำแผนภาพเชิงกิจกรรม ในภาษายูเอ็มแอล เช่น แผนภาพแสดงลำดับการทำงาน หรือแผนภาพแสดงกิจกรรม เป็นต้น ใช้เป็นข้อมูลเข้าสำหรับเครื่องมือวัดซอฟต์แวร์ซึ่งใช้เทคนิคการวัดแบบฟังก์ชันพอยต์เชิงวัตถุ ทำให้แนวทางการคำนวณหาจำนวนของฟังก์ชันพอยต์เชิงวัตถุมีการใช้ข้อมูลเข้าที่หลากหลายมากยิ่งขึ้น

2) การทำงานแบบไคลเอนต์เซอร์ฟเวอร์ของโปรแกรม Chula OOFP Counting มีเป้าหมายเพื่อความง่ายในการปรับปรุง ดูแลการทำงานของโมดูลการแปลงไฟล์ต้นฉบับภาษาจาวาเป็นข้อมูลภาษาเอโอแอล ซึ่งข้อด้อยของรูปแบบการทำงานลักษณะนี้คือต้องการผู้ดูแลระบบในการบำรุงรักษาเครื่องให้บริการ โดยแนวคิดที่สามารถลดการทำงานของผู้ดูแลระบบได้คือการเปลี่ยนรูปแบบการทำงานของโปรแกรม Chula OOFP Counting จากไคลเอนต์เซอร์ฟเวอร์ไปสู่โปรแกรมแบบเดสท็อปที่ทำงานบนเครื่องพีซี โดยการรวมโมดูลการแปลงไฟล์ต้นฉบับภาษาจาวาเป็นข้อมูลภาษาเอโอแอลให้เป็นส่วนหนึ่งในโปรแกรมไคลเอนต์ Chula OOFP Counting