

นางสาว เสาวลักษณ์ รัตนธรรมสกุล : การวัดซอฟต์แวร์เชิงแง่มุมโดยการวิเคราะห์ฟังก์ชันพอยต์ (MEASURING AOP-BASED SOFTWARE BY FUNCTION POINT ANALYSIS). อาจารย์ที่ปรึกษา : อาจารย์ ดร. โปรดปราน บุญยพุกกณะ, 109 หน้า. ISBN 974-53-2849-9.

การวัดขนาดซอฟต์แวร์เป็นกิจกรรมที่นำไปสู่การประเมินขนาด และค่าใช้จ่ายในการพัฒนาซอฟต์แวร์ หากสามารถประเมินได้อย่างแม่นยำตั้งแต่ระยะเริ่มแรกของโครงการ จะทำให้การจัดสรรทรัพยากรและการดำเนินงานเป็นไปอย่างมีประสิทธิภาพ การวัดขนาดซอฟต์แวร์โดยใช้เทคนิคการวิเคราะห์ฟังก์ชันพอยต์ (Function Point Analysis) เป็นวิธีการวัดขนาดซอฟต์แวร์วิธีหนึ่งที่มีผู้นิยมใช้มากและมีงานวิจัยทำการทดสอบและพบว่า ไม่ว่าจะทำการพัฒนาซอฟต์แวร์ด้วยเทคโนโลยีแบบใดก็ตาม ค่าฟังก์ชันพอยต์ที่ได้จะไม่มีการเปลี่ยนแปลง เนื่องจากฟังก์ชันพอยต์เป็นการวัดจำนวนฟังก์ชันที่มีอยู่ในซอฟต์แวร์ โดยใช้ข้อกำหนดความต้องการของซอฟต์แวร์ (Software Requirements Specification) เป็นข้อมูลพื้นฐานในการวิเคราะห์คำนวณขนาดของซอฟต์แวร์

เมื่อการเขียนโปรแกรมเชิงแง่มุม (Aspect-Oriented Programming) ได้ถูกนำเสนอขึ้นมา จุดเด่นประการหนึ่งคือทำให้ขนาดซอฟต์แวร์ลดลง ถึงแม้การเขียนโปรแกรมเชิงแง่มุมจะมีแนวความคิดที่เพิ่มเติมขึ้นจากการเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming) แต่ข้อกำหนดความต้องการของซอฟต์แวร์ได้แก่แผนภาพ Use Case ในเชิงแง่มุมอาจจะมีแตกต่างจากแผนภาพ Use Case ในเชิงวัตถุก็ได้ ดังนั้นในวิทยานิพนธ์นี้เป็นการทดสอบค่าฟังก์ชันพอยต์ที่ได้จากการวัดขนาดซอฟต์แวร์ที่พัฒนาด้วยการเขียนโปรแกรมเชิงแง่มุม ว่าสามารถใช้ได้กับการพัฒนาซอฟต์แวร์ภายใต้เทคโนโลยีเชิงแง่มุมได้หรือไม่

จากการวิเคราะห์และทดสอบโดยนำแผนภาพ Use Case ในเชิงวัตถุของระบบงานทางด้านการเงินที่เป็นกรณีศึกษาจำนวน 12 ระบบ มาแปลงเป็นแผนภาพ Use Case ในเชิงแง่มุมด้วยวิธีการของ Jacobson พบว่าค่าฟังก์ชันพอยต์ที่ได้จากการนับจากแผนภาพ Use Case ในเชิงแง่มุมมีค่าสูงกว่าค่าฟังก์ชันพอยต์ที่ได้จากการนับจากแผนภาพ Use Case ในเชิงวัตถุที่ระดับนัยสำคัญทางสถิติ 0.05 ($\alpha=0.05$) นอกจากนี้ในวิทยานิพนธ์ยังทำการทดสอบการวัดขนาดซอฟต์แวร์ด้วยวิธี Use Case Points ก็พบว่าได้ผลเช่นเดียวกัน

4671448921 : MAJOR COMPUTER SCIENCE

KEY WORD: ASPECT-ORIENTED PROGRAMMING / USE CASE DIAGRAM / FUNCTION POINT ANALYSIS

SAOWALUK RATANATAMSKUL : MEASURING AOP-BASED SOFTWARE BY
FUNCTION POINT ANALYSIS. THESIS ADVISOR : PROADPRAN PUNYABUKKANA,
Ph.D., 109 pp. ISBN 974-53-2849-9.

Measuring the size of software leads to an estimate of the effort and the expense of software development activities. We hope to obtain as accurate as possible the expense the estimate and as early as possible in order to effectively management resource allocation and track its progress. Function Point Analysis is a popular means to measure the size of the software that many research found to be independent of technology employed since the inputs for the measurement are derived from software requirements specification.

However, when Aspect-Oriented Programming (AOP) is introduced, one of its major promises includes fewer lines of code or smaller program. AOP may be thought as an extension of Object-Oriented Programming (OOP). The requirements specification such as use case in the AOP paradigm may differ from that of OOP. Therefore, this thesis aims to investigate if function point is still usable to measure software development under AOP environment.

Twelve case studies from a large financial institute were carried out using Jacobson method to convert use case under the OOP environment to AOP. The result shows that the numbers of function point under AOP are significantly higher than those under OOP ($\alpha=0.05$). This thesis further analyzes the counting of use case points and found the results to confirm the findings.