การออกแบบและพัฒนามาตรวัดซอฟต์แวร์โดยใช้เทคนิคฟังก์ชันพอยต์สำหรับโปรแกรมเชิงวัตถุ

นาย สมภพ ตาลสอน

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2544

ISBN 974-03-1215-2

ลิขสิทธิ์ของ จุฬาลงกรณ์มหาวิทยาลัย

19 พ.ย. 2546

I20305929

DESIGN AND IMPLEMENTATION OF A FUNCTION POINT MEASUREMENT TOOL FOR AN OBJECT-ORIENTED PROGRAM

MR. SOMPOP TALSON

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science in Computer Science

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2001

ISBN 974-03-1215-2

หัวข้อวิทยานิพนธ์	การออกแบบและพัฒนามาตรวัดซอฟต์แวร์โดยใช้เทคนิคฟังก์ชันพอยต์สำหรับโปรแกรมเชิงวัตถุ
โดย	นาย สมภพ ตาลสอน
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. พรศิริ หมื่นไชยศรี

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์

(ศาสตราจารย์ ดร.สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(ผู้ช่วยศาสตราจารย์ วิชาญ เลิศวิภาตระกูล)

.....อาจารย์ที่ปรึกษา

(ผู้ช่วยศาสตราจารย์ ดร.พรศิริ หมื่นไชยศรี)

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ วันพร ปิ้นเก่า)

.....กรรมการ

(อาจารย์ จารุมาตร ปิ่นทอง)

สมภพ ตาลสอน: การออกแบบและพัฒนามาตรวัดซอฟต์แวร์โดยใช้เทคนิคฟังก์ชันพอยต์สำหรับโปรแกรมเชิงวัตถุ (DESIGN AND IMPLEMENTATION OF A FUNCTION POINT MEASUREMENT TOOL FOR AN OBJECT-ORIENTED PROGRAM) อาจารย์ที่ปรึกษา: ผู้ช่วยศาสตราจารย์ ดร. พรศิริ หมื่นไชยศรี, 123 หน้า. ISBN 974-03-1215-2

ในการพัฒนาซอฟต์แวร์เชิงวัตถุ การวางแผนเพื่อจัดสรรทรัพยากรที่มีอยู่จำกัดอย่างเหมาะสมเป็นสิ่งสำคัญ ดังนั้นการทราบถึงขนาดของซอฟต์แวร์ตั้งแต่ระยะเริ่มแรกของโครงงานทำให้การจัดสรรทรัพยากรเป็นไปอย่างมีประสิทธิภาพ วิทยานิพนธ์นี้เป็นการออกแบบและพัฒนาเครื่องมือวัดขนาดซอฟต์แวร์สำหรับโปรแกรมเชิงวัตถุ โดยเครื่องมือดังกล่าวมีชื่อว่า Chula OOFP Counting ซึ่งใช้เทคนิคการวัดแบบฟังก์ชันพอยต์เชิงวัตถุ เทคนิคการวัดนี้ดัดแปลงมาจากฟังก์ชันพอยต์แบบดั้งเดิมซึ่งทำงานกับแนวคิดการออกแบบและพัฒนาซอฟต์แวร์แบบโครงสร้าง โดยที่การวัดไม่ขึ้นอยู่กับภาษาคอมพิวเตอร์ที่ใช้ในการพัฒนา ฟังก์ชันพอยต์ใช้ข้อกำหนดความต้องการของซอฟต์แวร์ (Software Requirement Specification) เป็นข้อมูลเข้าในการวิเคราะห์คำนวณขนาดของซอฟต์แวร์ งานวิจัยนี้ได้ใช้แผนภาพคลาสซึ่งเป็นหนึ่งในข้อกำหนดความต้องการของซอฟต์แวร์เป็นข้อมูลเข้าสำหรับเครื่องมือวัดที่พัฒนาขึ้น จำนวนฟังก์ชันพอยต์ของคลาสและของซอฟต์แวร์เป็นค่าวัดที่ได้จากเครื่องมือวัดนี้ ข้อมูลจำนวนฟังก์ชันพอยต์เหล่านี้ทำให้ผู้พัฒนาทราบขนาดของซอฟต์แวร์เชิงวัตถุได้ในระหว่างขั้นตอนการวิเคราะห์และออกแบบระบบ ซึ่งช่วยให้นักพัฒนาหรือผู้บริหารโครงงานสามารถจัดสรรทรัพยากรที่มีอยู่อย่างเหมาะสมกับขนาดซอฟต์แวร์ที่วัดได้จริง

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต สมภพ ตาลสอน
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่ออาจารย์ที่ปรึกษา พรศิริ หมื่นไชยศรี
ปีการศึกษา 2544

##4171501021 : MAJOR COMPUTER SCIENCE
KEY WORD : SIZE / MEASUREMENT / METRICS / OBJECT-ORIENTED PROGRAM / FUNCTION POINTS / OBJECT-ORIENTED FUNCTION POINTS/ OOFP

SOMPOP TALSON : DESIGN AND IMPLEMENTATION OF A FUNCTION POINT MEASUREMENT TOOL FOR AN OBJECT-ORIENTED PROGRAM. THESIS ADVISOR: ASSISTANT PROFESSOR DR. PORNSIRI MUENCHAISRI, 123 pp. ISBN 974-03-1215-2.

In object-oriented software development projects, the resource allocation is an important part of project planning. So, the awareness of the software size since the initial phase makes the project planners allocate the resource effectively. This thesis is the design and implementation of a measurement tool for object-oriented programs. The tool is named Chula OOFP Counting. Object-Oriented Function Points (OOFP) is a technique used in the tool. OOFP is based on an adaptation of classical Function Points (FP) method to enable the measurement of object-oriented analysis and design specifications. Originally, FP has dealt with structural software development practices and Software Requirement Specification (SRS) is used as input. Class diagram is a Software Requirement Specification used as an input for the measurement tool in this research. The tool has been constructed to automate the counting method and the number of function point in classes and entire software would be reported. With This information, developers roughly know the object-oriented software size during design phase. Particularly, the software developers or project managers are able to allocate resources properly to the measured software size

Department Computer Engineering ..... Student's signature Sompop Talson .....

Field of study Computer Science ..... Advisor's signature P. Muenchaisri .....

Academic year 2001 .....

กิตติกรรมประกาศ

ข้าพเจ้าใคร่ขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร. พรศิริ หมื่นไชยศรี อาจารย์ที่ปรึกษาวิทยานิพนธ์ของข้าพเจ้า ที่ท่านเป็นผู้แนะนำให้ความรู้ คำปรึกษา ความช่วยเหลือต่างๆ ตลอดจนคอยดูแลการทำวิจัยของข้าพเจ้าอย่างดียิ่งจนสำเร็จลุล่วงลงได้ด้วยดี

ขอกราบขอบพระคุณผู้ช่วยศาสตราจารย์ วิชาญ เลิศวิภาตระกูล เป็นประธานกรรมการ ผู้ช่วยศาสตราจารย์ วันพร ปั้นเก่า และอาจารย์ จารุมาตร ปิ่นทอง เป็นกรรมการสอบวิทยานิพนธ์ ซึ่งได้สละเวลาและให้คำแนะนำต่างๆ ในการสอบวิทยานิพนธ์ของข้าพเจ้าได้อย่างดียิ่ง

ขอขอบคุณเพื่อนร่วมงาน และเพื่อน ๆ ที่เคยได้ศึกษาด้วยกันมา ที่ได้ให้คำแนะนำและกำลังใจต่าง ๆ แก่ข้าพเจ้า และขอขอบคุณบริษัท รอยเตอร์ (ประเทศไทย) จำกัด ที่ได้ให้มีโอกาสศึกษาต่อในสาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ของภาควิศวกรรมคอมพิวเตอร์แห่งนี้

ท้ายที่สุด ข้าพเจ้าใคร่ขอกราบขอบพระคุณบิดา และมารดา ที่ได้ให้โอกาสและสนับสนุนในด้านการเงินและกำลังใจแก่ข้าพเจ้าเสมอมา

นายสมภพ ตาลสอน

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....ง

บทคัดย่อภาษาอังกฤษ.....จ

กิตติกรรมประกาศ.....ฉ

สารบัญ.....ช

สารบัญตาราง.....ฎ

สารบัญรูป.....ฎ

บทที่ 1.....1

1.1 ความเป็นมาและความสำคัญของปัญหา.....1

1.2 วัตถุประสงค์.....2

1.3 ขอบเขตของงานวิจัย.....2

1.4 ขั้นตอนและวิธีการดำเนินงาน.....2

1.5 ประโยชน์ที่คาดว่าจะได้รับ.....3

บทที่ 2.....4

2.1 ทฤษฎีที่เกี่ยวข้อง.....4

2.1.1 การวิเคราะห์และออกแบบเชิงวัตถุ.....4

2.1.2 การวัดเชิงขนาด (Size-Oriented Metrics).....8

2.1.3 การวัดเชิงหน้าที่ (Function-Oriented Metrics).....9

2.2 งานวิจัยที่เกี่ยวข้อง.....21

2.2.1 Mapping the OO-Jacobson Approach into Function Point Analysis [3].....21

2.2.2 Definition and Experimental Evaluation of Function Points for Object-Oriented Systems [4].....21

บทที่ 3.....22

3.1 สถาปัตยกรรมไคลเอนต์เซอร์ฟเวอร์ที่ใช้ในระบบ.....22

3.2 ภาพรวมในการพัฒนาเครื่องมือวัด.....23

3.3 โมเดลการใช้งาน (Use-Case Diagram) ของเครื่องมือวัดซอฟต์แวร์ซึ่งใช้เทคนิคฟังก์ชันพอยต์เชิงวัตถุ..25

3.4 แผนภาพคลาส (Class Diagram).....27

3.4.1 การสร้าง การเปิด และการปรับปรุงแก้ไขไฟล์ต้นฉบับภาษาจาวา.....29

3.4.2 การสร้าง การเลือก และการจัดการโปรเจ็คไฟล์.....31

3.4.3 การคอมไพล์ไฟล์ต้นฉบับภาษาจาวาและการประมวลผลโปรแกรมที่พัฒนาด้วยภาษาจาวา.....35

สารบัญ (ต่อ)

3.4.4 การสร้างข้อมูลภาษาเอโอแอลจากไฟล์ต้นฉบับภาษาจาวาในโปรเจ็คไฟล์ 37
3.4.5 การนับจำนวนของฟังก์ชันพอยต์เชิงวัตถุจากข้อมูลภาษาเอโอแอล....39
3.5 แผนภาพแสดงลำดับการทำงาน (Sequence Diagram)....41
3.5.1 การสร้าง การเปิด และการปรับปรุงแก้ไขไฟล์ต้นฉบับภาษาจาวา....41
3.5.2 การสร้าง การเลือก และการจัดการโปรเจ็คไฟล์....43
3.5.3 การคอมไพล์ไฟล์ต้นฉบับภาษาจาวา....43
3.5.4 การประมวลผลโปรแกรมที่พัฒนาด้วยภาษาจาวา....46
3.5.5 การสร้างข้อมูลภาษาเอโอแอลจากไฟล์ต้นฉบับภาษาจาวาในโปรเจ็คไฟล์ 46
3.5.6 การนับจำนวนของฟังก์ชันพอยต์เชิงวัตถุจากข้อมูลภาษาเอโอแอล....48
3.6 แผนภาพแสดงกิจกรรม (Activity Diagram)....49
3.6.1 แผนภาพแสดงกิจกรรมสำหรับเมนู File....49
3.6.2 แผนภาพแสดงกิจกรรมสำหรับเมนู Edit....52
3.6.3 แผนภาพแสดงกิจกรรมสำหรับเมนู Project....54
3.6.4 แผนภาพแสดงกิจกรรมสำหรับเมนู Function Point....55
3.6.5 แผนภาพแสดงกิจกรรมสำหรับเมนู Tools....57
3.6.6 แผนภาพแสดงกิจกรรมสำหรับเมนู Window....57
3.6.7 แผนภาพแสดงกิจกรรมสำหรับเมนู Help....59
บทที่ 4....62
4.1 การทำงานแบบไคลเอนต์เซอร์ฟเวอร์ของเครื่องมือวัด....62
4.1.1 เครื่องไคลเอนต์ติดตั้งโปรแกรม Chula OOFP Counting....62
4.1.2 เครื่องให้บริการเว็บพร้อมด้วยชุดคำสั่งภาษาเพิร์ลของโปรแกรม Chula OOFP Counting....63
4.2 การสร้างแพ็กเกจคอมไพเลอร์ภาษาจาวา....64
4.3 การสร้างแพ็กเกจคอมไพเลอร์ภาษาเอโอแอล....65
4.4 การสร้างข้อมูลภาษาเอโอแอลจากไฟล์ต้นฉบับภาษาจาวา....66
4.4.1 ส่วนการอ่านค่ายูอาร์แอลจากไฟล์คอนฟิกกูเรชันและการรับค่าไอพีแอดเดรสจากเครื่องไคลเอนต์ 69
4.4.2 สร้างการเชื่อมต่อระหว่างโปรแกรมไคลเอนต์ Chula OOFP Counting กับเครื่องให้บริการเว็บ....70
4.4.3 การส่งข้อมูลรายละเอียดของโปรเจ็คไฟล์จากโปรแกรมไคลเอนต์ไปยังเครื่องให้บริการเว็บ....71
4.4.4 การเรียกใช้ชุดคำสั่งภาษาเพิร์ลเพื่อทำการแปลงไฟล์ต้นฉบับภาษาจาวาเป็นข้อมูลภาษาเอโอแอล.73
4.5 การนับจำนวนฟังก์ชันพอยต์เชิงวัตถุจากข้อมูลภาษาเอโอแอล....74
4.5.1 การสร้างแพ็กเกจคอมไพเลอร์ภาษาเอโอแอล....74

สารบัญ (ต่อ)

4.5.2 การสร้างไฟล์ต้นฉบับสำหรับอธิบายไวยากรณ์ภาษาเอโอแอล พร้อมกับฟังก์ชันและคลาสต่างๆที่ใช้ประกอบการนับจำนวนของฟังก์ชันพอยต์เชิงวัตถุ 74
บทที่ 5 85
5.1 การเปรียบเทียบผลการทดสอบการคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุที่ได้จากโปรแกรม Chula OOFP Counting กับการนับจำนวนของฟังก์ชันพอยต์เชิงวัตถุด้วยมือวัตถุบนแผนภาพคลาส 85
5.1.1 โปรแกรมทดสอบที่ 1 86
5.1.2 โปรแกรมทดสอบที่ 2 92
5.1.3 โปรแกรมทดสอบที่ 3 94
5.1.4 โปรแกรมทดสอบที่ 4 97
5.2 สรุปผลการทดสอบ 102
บทที่ 6 103
6.1 บทสรุป 103
6.2 ข้อเสนอแนะ 104

รายการอ้างอิง 105
ภาคผนวก 106
ประวัติผู้เขียนวิทยานิพนธ์ 123

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดงตัวอย่างโครงงานพัฒนาซอฟต์แวร์พร้อมกับข้อมูลการวัดเชิงขนาด [1]..... 8

ตารางที่ 2.2 แสดงค่าน้ำหนัก (Weight Factor) สำหรับแต่ละองค์ประกอบพื้นฐานของซอฟต์แวร์ [2].. 10

ตารางที่ 2.3 แสดงจำนวนบรรทัดเฉลี่ยของโค้ดต่อ 1 ฟังก์ชันพอยต์สำหรับภาษาต่างๆ [1]..... 13

ตารางที่ 2.4 แสดงการแปลงค่าความซับซ้อนของเซอร์วิสรีเควสไปสู่ค่าฟังก์ชันพอยต์ [5]..... 19

ตารางที่ 2.5 แสดงการแปลงค่าความซับซ้อนของลอจิคอลไฟล์ไปสู่ค่าฟังก์ชันพอยต์ [5]..... 19

ตารางที่ 2.6 แสดงผลการนับจำนวน DET และ RET ของคลาสจากแผนภาพคลาส..... 20

ตารางที่ 2.7 แสดงผลการคำนวณจำนวนฟังก์ชันพอยต์ของแผนภาพตัวอย่าง..... 20

ตารางที่ 3.1 แสดงผลการวิเคราะห์โมเดลการใช้งานของระบบ..... 27

ตารางที่ 4.1 ตารางแสดงความซับซ้อนของลอจิคอลไฟล์ (Logical Complexity Matrix) [5]..... 80

ตารางที่ 4.2 ตารางแสดงความซับซ้อนของเซอร์วิสรีเควส [5]..... 81

ตารางที่ 5.1 ตารางแสดงผลการนับจำนวน DET และFTR ของเซอร์วิสรีเควส..... 86

ตารางที่ 5.2 ตารางแสดงความซับซ้อนของเซอร์วิสรีเควส [5]..... 87

ตารางที่ 5.3 แสดงการแปลงค่าความซับซ้อนของเซอร์วิสรีเควสไปสู่ค่าฟังก์ชันพอยต์ [5]..... 87

ตารางที่ 5.4 ตารางแสดงความซับซ้อนของลอจิคอลไฟล์ [5]..... 88

ตารางที่ 5.5 ตารางแสดงผลการนับจำนวน DET และ RET สำหรับลอจิคอลไฟล์ของคลาส..... 89

ตารางที่ 5.6 แสดงการแปลงค่าความซับซ้อนของลอจิคอลไฟล์ไปสู่ค่าฟังก์ชันพอยต์ [5]..... 89

ตารางที่ 5.7 ผลการเปรียบเทียบการนับจำนวนฟังก์ชันพอยต์เชิงวัตถุโดยเครื่องมือวัด Chula OOFP Counting และการนับด้วยมือบนแผนภาพคลาสของโปรแกรมทดสอบ CardTest..... 91

ตารางที่ 5.8 ผลการเปรียบเทียบการนับจำนวนฟังก์ชันพอยต์เชิงวัตถุโดยเครื่องมือวัด Chula OOFP Counting และการนับด้วยมือบนแผนภาพคลาสของโปรแกรมทดสอบ ArcTest..... 94

ตารางที่ 5.9 ผลการเปรียบเทียบการนับจำนวนฟังก์ชันพอยต์เชิงวัตถุโดยเครื่องมือวัด Chula OOFP Counting และการนับด้วยมือบนแผนภาพคลาสของโปรแกรมทดสอบ DrawTest..... 95

ตารางที่ 5.10 ผลการเปรียบเทียบการนับจำนวนฟังก์ชันพอยต์เชิงวัตถุโดยเครื่องมือวัด Chula OOFP Counting และการนับด้วยมือบนแผนภาพคลาสของโปรแกรมทดสอบ Metalwork 97

สารบัญรูป

หน้า

รูปที่ 2.1 โมเดลของฟังก์ชันพอยต์ [3]..... 10
รูปที่ 2.2 แผนภาพกิจกรรมการนับของฟังก์ชันพอยต์ [5]..... 11
รูปที่ 2.3 โมเดลกระบวนการนับของฟังก์ชันพอยต์ [4]..... 15
รูปที่ 2.4 ซิงเกิ้ลคลาส..... 17
รูปที่ 2.5 แอกกรีเกชัน..... 17
รูปที่ 2.6 เจนเนอร์รัลไลเซชัน / สเปเชี่ยลไลเซชั่น..... 18
รูปที่ 3.1 รูปแบบการทำงานของโปรแกรม Chula OOFP Counting..... 23
รูปที่ 3.2 แผนภาพแสดงกิจกรรมการคำนวณหาจำนวนของฟังก์ชันพอยต์เชิงวัตถุจากแผนภาพคลาส... 24
รูปที่ 3.3 โมเดลการใช้งานของเครื่องมือวัดซอฟต์แวร์ซึ่งใช้เทคนิคฟังก์ชันพอยต์เชิงวัตถุ..... 25
รูปที่ 3.4 แสดงแผนภาพคลาสหลักของเครื่องมือวัดซอฟต์แวร์ซึ่งใช้เทคนิคฟังก์ชันพอยต์เชิงวัตถุ..... 30
รูปที่ 3.5 แผนภาพคลาสสำหรับยูสเคส การสร้าง การเปิด และการปรับปรุงแก้ไขไฟล์โปรแกรมภาษาจาวา..... 32
รูปที่ 3.6 แผนภาพคลาสสำหรับยูสเคส การสร้าง การเลือก และการจัดการโปรเจ็คไฟล์..... 34
รูปที่ 3.7 แผนภาพคลาสสำหรับยูสเคส การคอมไพล์ไฟล์ต้นฉบับภาษาจาวา..... 36
รูปที่ 3.8 แผนภาพคลาสสำหรับยูสเคส การสร้างข้อมูลภาษาเอโอแอลจากไฟล์ภาษาจาวาในโปรเจ็คไฟล์..... 38
รูปที่ 3.9 แผนภาพคลาสสำหรับยูสเคส การนับจำนวนของฟังก์ชันพอยต์เชิงวัตถุจากข้อมูลภาษาเอโอแอล..... 40
รูปที่ 3.10 แผนภาพแสดงลำดับการทำงานสำหรับยูสเคส การสร้าง การเปิด และการปรับปรุงแก้ไขไฟล์ต้นฉบับภาษาจาวา..... 42
รูปที่ 3.11 แผนภาพแสดงลำดับการทำงานสำหรับยูสเคส การสร้าง การเปิด การเลือก และการจัดการโปรเจ็คไฟล์..... 44
รูปที่ 3.12 แผนภาพแสดงลำดับการทำงานสำหรับยูสเคส การคอมไพล์ไฟล์ต้นฉบับภาษาจาวา..... 45
รูปที่ 3.13 แผนภาพแสดงลำดับการทำงานสำหรับยูสเคส การประมวลผลโปรแกรมที่พัฒนาด้วยภาษาจาวา..... 47
รูปที่ 3.14 แผนภาพแสดงลำดับการทำงานสำหรับยูสเคส การสร้างข้อมูลภาษาเอโอแอล..... 48
รูปที่ 3.15 แผนภาพแสดงลำดับการทำงานสำหรับยูสเคส การนับจำนวนของฟังก์ชันพอยต์เชิงวัตถุจากข้อมูลภาษาเอโอแอล..... 49
รูปที่ 3.16 แผนภาพแสดงกิจกรรมสำหรับเมนู File..... 50

สารบัญรูป (ต่อ)

รูปที่ 3.17 แผนภาพแสดงกิจกรรมสำหรับเมนู Edit..... 53
รูปที่ 3.18 แผนภาพแสดงกิจกรรมสำหรับเมนู Project..... 54
รูปที่ 3.19 แผนภาพแสดงกิจกรรมสำหรับเมนู Function Point..... 56
รูปที่ 3.20 แผนภาพแสดงกิจกรรมสำหรับเมนู Tools..... 58
รูปที่ 3.21 แผนภาพแสดงกิจกรรมสำหรับเมนู Window..... 60
รูปที่ 3.22 แผนภาพแสดงกิจกรรมสำหรับเมนู Help..... 61
รูปที่ 4.1 แผนภาพแสดงขั้นตอนการสร้างแพ็กเกจคอมไพเลอร์ภาษาจาวา..... 64
รูปที่ 4.2 แผนภาพแสดงขั้นตอนการสร้างแพ็กเกจคอมไพเลอร์ภาษาเอโอแอล..... 65
รูปที่ 4.3 แผนภาพแสดงขั้นตอนการสร้างข้อมูลภาษา เอโอแอล จากไฟล์ต้นฉบับภาษาจาวา..... 67
รูปที่ 4.3 แผนภาพแสดงขั้นตอนการสร้างข้อมูลภาษา เอโอแอล จากไฟล์ต้นฉบับภาษาจาวา (ต่อ)..... 68
รูปที่ 4.4 โค้ดการอ่านค่ายูอาร์แอลจากไฟล์คอนฟิกกูเรชันและการรับค่าไอพีแอดเดรสจากเครื่องไคลเอนต์..... 69
รูปที่ 4.5 โค้ดการเชื่อมต่อระหว่างโปรแกรมไคลเอนต์ Chula OOFP Counting กับเครื่องให้บริการเว็บ 70
รูปที่ 4.6 โค้ดตัวอย่างการวิธีการใช้งานคลาส JServConnect..... 71
รูปที่ 4.7 โค้ดการส่งข้อมูลรายละเอียดของโปรเจ็คไฟล์จากโปรแกรมไคลเอนต์ไปยังเครื่องให้บริการเว็บ..... 71
รูปที่ 4.7 โค้ดการส่งข้อมูลรายละเอียดของโปรเจ็คไฟล์จากโปรแกรมไคลเอนต์ไปยังเครื่องให้บริการเว็บ (ต่อ)..... 72
รูปที่ 4.8 โค้ดการเรียกใช้ชุดคำสั่งภาษาเพิร์ลเพื่อวิเคราะห์ไฟล์ต้นฉบับภาษาจาวาและสร้างข้อมูลภาษาเอโอแอล..... 73
รูปที่ 4.9 โค้ดแสดงค่าตัวเลือกซึ่งถูกใช้โดยโปรแกรมจาวาซีซี..... 75
รูปที่ 4.10 รายละเอียดคลาสสำหรับการประมวลผลโทเคนและไวยากรณ์..... 77
รูปที่ 4.10 รายละเอียดคลาสสำหรับการประมวลผลโทเคนและไวยากรณ์ (ต่อ)..... 78
รูปที่ 4.11 โค้ดแสดงคลาสที่ทำหน้าที่เก็บข้อมูลรายละเอียดของคลาสในไฟล์เอโอแอล..... 79
รูปที่ 4.12 แผนภาพคลาสที่ทำหน้าที่เก็บข้อมูลความสัมพันธ์ระหว่างคลาสต่างๆในไฟล์เอโอแอล..... 82
รูปที่ 4.13 โค้ดแสดงคลาสที่ทำหน้าที่เก็บข้อมูลความสัมพันธ์ระหว่างคลาสในไฟล์เอโอแอล..... 82
รูปที่ 4.14 โค้ดแสดงโทเคนที่ประกาศใช้สำหรับไวยากรณ์ของภาษาเอโอแอล..... 83
รูปที่ 4.15 แผนภาพคลาสของโมดูลพาร์เซอร์และโมดูลการคำนวณฟังก์ชันพอยต์เชิงวัตถุบนภาษาเอโอแอล..... 84
รูปที่ 5.1 แผนภาพคลาสสำหรับการออกแบบโปรแกรม CardTest..... 86
รูปที่ 5.2 ผลลัพธ์การคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุจากโปรแกรม Chula OOFP Counting บนแผนภาพคลาส CardTest..... 90

สารบัญรูป (ต่อ)

รูปที่ 5.2 ผลลัพธ์การคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุจากโปรแกรม Chula OOFP Counting บนแผนภาพคลาส CardTest (ต่อ) 91
รูปที่ 5.3 แผนภาพคลาสสำหรับการออกแบบโปรแกรม ArcTest..... 92
รูปที่ 5.4 ผลลัพธ์การคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุจากโปรแกรม Chula OOFP Counting บนแผนภาพคลาส ArcTest..... 93
รูปที่ 5.5 แผนภาพคลาสสำหรับการออกแบบโปรแกรม DrawTest..... 95
รูปที่ 5.6 ผลลัพธ์การคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุจากโปรแกรม Chula OOFP Counting บนแผนภาพคลาส DrawTest..... 96
รูปที่ 5.7 แผนภาพคลาสสำหรับการออกแบบโปรแกรม Metalworks..... 99
รูปที่ 5.8 ผลลัพธ์การคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุจากโปรแกรม Chula OOFP Counting บนแผนภาพคลาส Metalworks..... 100
รูปที่ 5.8 ผลลัพธ์การคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุจากโปรแกรม Chula OOFP Counting บนแผนภาพคลาส Metalworks (ต่อ)..... 101
รูปที่ 5.8 ผลลัพธ์การคำนวณหาจำนวนฟังก์ชันพอยต์เชิงวัตถุจากโปรแกรม Chula OOFP Counting บนแผนภาพคลาส Metalworks (ต่อ)..... 102