การวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบ

นางสาวอลิสรา หินชีระนันท์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2555
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ที่ส่งผ่านทางบัณฑิตวิทยาลัย
The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)
are the thesis authors' files submitted through the Graduate School.

A MEASUREMENT OF SOFTWARE MAINTAINABILITY IN DESIGN PHASE

Miss Alisara Hincheeranan

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science Program in Software Engineering

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2012

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบ
โดย	นางสาวอลิสรา หินชีระนันท์
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(รองศาสตราจารย์ ดร.พรศิริ หมื่นไชยศรี)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.อรรถสิทธิ์ สุรฤกษ์)

.....กรรมการภายนอกมหาวิทยาลัย
(อาจารย์ ดร.สุนีย์ พงษ์พินิจภิญโญ)

อลิสรา หินชีระนันท์ : การวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบ (A MEASUREMENT OF SOFTWARE MAINTAINABILITY IN DESIGN PHASE) อ.ที่ปรึกษาวิทยานิพนธ์หลัก : รศ.ดร.วันชัย ริ้วไพบูลย์, 102 หน้า.

งานวิจัยนี้มีวัตถุประสงค์เพื่อสร้างโมเดลและพัฒนาเครื่องมือเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบด้วยแผนภาพคลาส โดยใช้มาตรวัดเชิงวัตถุที่สามารถคำนวณด้วยแผนภาพคลาส 11 มาตรวัด การคำนวณหาค่าคุณภาพของการออกแบบ และการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ที่สามารถวัดได้ 2 ระดับ คือ ระดับง่าย และระดับยาก จากนั้นทำการทดลองและการเก็บรวบรวมข้อมูลเพื่อนำไปสร้างเป็นโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ด้วยวิธีการวิเคราะห์การจำแนกกลุ่ม ระบบที่นำมาใช้ในการทดลองมีจำนวน 13 ระบบ แบ่งออกเป็นระบบที่ใช้ในการสร้างโมเดล 10 ระบบ และระบบที่ใช้ในการทดสอบความถูกต้องของการทำงานของเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ 3 ระบบ

ผู้วิจัยได้ทำการพัฒนาเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ด้วยภาษาซีชาร์ป ซึ่งข้อมูลนำเข้าสำหรับเครื่องมือ คือ แผนภาพคลาสที่สร้างแผนภาพคลาสและแปลงแผนภาพคลาสให้อยู่ในรูปแบบของเอกสารเอ็กซ์เอ็มแอลด้วยโปรแกมสตาร์ยูเอ็มแอล

ผลการวิเคราะห์จากการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ และผลการทดสอบความถูกต้องของเครื่องมือ พบว่าความสามารถในการยืดหยุ่น และความสามารถในการขยาย ซึ่งเป็นค่าคุณภาพของการออกแบบเป็นตัวแปรอิสระที่มีความเหมาะสมในการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ และเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์สามารถทำงานได้อย่างถูกต้องตรงกับความต้องการในด้านหน้าที่และความต้องการที่ไม่ใช่หน้าที่

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....
สาขาวิชา วิศวกรรมซอฟต์แวร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์.....
ปีการศึกษา 2555

5270819821 : MAJOR SOFTWARE ENGINEERING
KEYWORDS : MAINTAINABILITY / OBJECT ORIENTED METRICS

ALISARA HINCHEERANAN : A MEASUREMENT OF SOFTWARE MAINTAINABILITY IN DESIGN PHASE. ADVISOR: ASSOC. PROF. WANCHAI RIVEPIBOON, Ph.D. 102 pp.

The objective of this thesis is to establish a model and the Maintainability Estimation Tool (MET) for measurement of maintainability in design phase from class diagram. This thesis uses eleven design metrics for class diagram and a design quality attributes in order to construct a maintainability estimation model. The model can identify two levels of maintainability, which are easy level and difficulty level. The data collected for establishing a model from ten systems and three systems for test functional requirements and non functional requirements of tool.

This research constructs an automation tool developed with c# language. This tool are measuring software metrics and estimating a level of maintainability. The input data for this tool, which are an XML documents representing class diagram and exported from the StartUML tool.

The results from constructing a maintainability estimation model found that flexibility and extendibility are independent variables for establishing a maintainability estimation model and the Maintainability Estimation Tool (MET) work correctly meets the functional requirement and non functional requirements.

Department : Computer Engineering Student's Signature .....
Field of Study : Software Engineering Advisor's Signature .....
Academic Year : 2012 .....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความอนุเคราะห์เป็นอย่างยิ่งของ รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ของข้าพเจ้า ซึ่งท่านได้ให้คำแนะนำ และข้อเสนอแนะต่างๆ ตลอดระยะเวลาของการจัดทำวิทยานิพนธ์ของข้าพเจ้าอย่างดียิ่ง จนสำเร็จลุล่วงได้ด้วยดี

ขอขอบพระคุณรองศาสตราจารย์ ดร.พรศิริ หมื่นไชยศรี เป็นประธานกรรมการ ผู้ช่วยศาสตราจารย์ ดร.อรรถสิทธิ์ สุรฤกษ์ และอาจารย์ ดร.สุนีย์ พงษ์พินิจภิญโญ เป็นกรรมการสอบวิทยานิพนธ์ ซึ่งได้สละเวลาและให้คำแนะนำต่างๆ ในการสอบวิทยานิพนธ์ของข้าพเจ้าอย่างดียิ่ง

และท้ายที่สุดนี้ขอขอบคุณเพื่อนทุกๆคนที่ติดตามและให้กำลังใจ รวมถึงท่านอื่นๆ ที่มิได้กล่าวชื่อไว้ ณ ที่นี้ ที่มีส่วนทำให้วิทยานิพนธ์สำเร็จได้ด้วยดี

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง
บทคัดย่อภาษาอังกฤษ ..... จ
กิตติกรรมประกาศ ..... ฉ
สารบัญ ..... ช
สารบัญตาราง ..... ฎ
สารบัญรูป ..... ฌ
บทที่ 1 บทนำ 1
1.1 ความเป็นมาและความสำคัญของปัญหา 1
1.2 วัตถุประสงค์ของการวิจัย 2
1.3 ขอบเขตของการวิจัย 2
1.4 ขั้นตอนและวิธีดำเนินการวิจัย 2
1.5 ประโยชน์ที่คาดว่าจะได้รับ 3
1.6 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์ 4
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 5
2.1 ทฤษฎีที่เกี่ยวข้อง 5
2.2 งานวิจัยที่เกี่ยวข้อง 18
บทที่ 3 การดำเนินการวิจัย 20
3.1 การสร้างแผนภาพคลาสและการแปลงแผนภาพคลาสในรูปแบบเอกสารเอ็กซ์เอ็มแอล 21
3.2 การดำเนินการวิจัย 22
3.3 การพัฒนาเครื่องมือ 27
3.4 การประเมินและสรุปผลการทดลอง 28

หน้า

บทที่ 4 การออกแบบและพัฒนาเครื่องมือ 29
4.1 ความต้องการด้านหน้าที่ 29
4.2 ความต้องการที่ไม่ใช่หน้าที่ 32
4.3 การออกแบบและพัฒนาเครื่องมือ 32
4.4 การอ่านข้อมูลของเอกสารเอ็กซ์เอ็มแอล 40
4.5 สภาพแวดล้อมและเครื่องมือที่ใช้ในการพัฒนา 44
4.6 การติดตั้งซอฟต์แวร์เพื่อใช้ในการพัฒนา 44
4.7 การพัฒนาส่วนต่อประสาน 45
บทที่ 5 ผลการทดลอง 46
5.1 ข้อมูลจากการคำนวณด้วยมาตรวัดทั้ง 11 มาตรวัด 46
5.2 ข้อมูลจากการคำนวณหาค่าคุณภาพของการออกแบบ 46
5.3 การสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ 50
5.4 สรุปผลการทดลองและการทดสอบ 66
บทที่ 6 สรุปผลการวิจัย และข้อเสนอแนะ 67
6.1 สรุปผลการวิจัย 67
6.2 ข้อเสนอแนะ 67
รายการอ้างอิง 69
ภาคผนวก 72
ประวัติผู้เขียนวิทยานิพนธ์ 102

สารบัญตาราง

หน้า

ตารางที่ 1 แสดงความสัญลักษณ์ของความสัมพันธ์ของแผนภาพคลาส 7
ตารางที่ 2 แสดงคุณภาพของการออกแบบ 10
ตารางที่ 3 แสดงคุณสมบัติของการออกแบบเชิงวัตถุ 11
ตารางที่ 4 แสดงมาตรวัดเชิงวัตถุ 12
ตารางที่ 5 แสดงคุณสมบัติของการออกแบบเชิงวัตถุ 13
ตารางที่ 6 แสดงความสัมพันธ์ระหว่างคุณสมบัติและคุณภาพของการออกแบบ.. 14
ตารางที่ 7แสดงรายชื่อระบบและจำนวนคลาสเพื่อใช้สร้างโมเดล 21
ตารางที่ 8 แสดงคุณสมบัติของการออกแบบและมาตรวัดเชิงวัตถุ 23
ตารางที่ 9 มาตรวัดเชิงวัตถุที่ใช้ในงานวิจัย 25
ตารางที่ 10 สมการการคำนวณคุณภาพของการออกแบบ 26
ตารางที่ 11 ความต้องการด้านหน้าที่ 29
ตารางที่ 12 ความต้องการที่ไม่ใช่หน้าที่ 32
ตารางที่ 13 แสดงผลการคำนวณมาตรวัดเชิงวัตถุทั้ง 10 ระบบ 48
ตารางที่ 14 แสดงผลการคำนวณหาค่าคุณภาพของการออกแบบทั้ง 10 ระบบ... 49
ตารางที่ 15 Variables Entered/Removed 50
ตารางที่ 16 Variables in the Analysis 51
ตารางที่ 17 Variable Not in the Analysis 51
ตารางที่ 18 Standardized Canonical Discriminant Function 52
ตารางที่ 19 Structure Matrix 53
ตารางที่ 20 Canonical Discriminant Function 53
ตารางที่ 21 Function at Group Centroids 54
ตารางที่ 22 Classification Function 54
ตารางที่ 23 Classification Results 55
ตารางที่ 24 ผลการทดลองการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ 57
ตารางที่ 25 การทดสอบการนำเอกสารเอ็กซ์เอ็มแอลเข้าสู่ระบบ 58
ตารางที่ 26 การคำนวณด้วยมาตรวัดเชิงวัตถุ 59

หน้า

ตารางที่ 27 การผลการคำนวณด้วยมาตรวัดเชิงวัตถุ 60
ตารางที่ 28 การแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุ 60
ตารางที่ 29 การคำนวณหาค่าคุณภาพของการออกแบบ 61
ตารางที่ 30 การแสดงผลการคำนวณหาค่าคุณภาพของการออกแบบ 61
ตารางที่ 31 การคำนวณการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ 62
ตารางที่ 32 การแสดงผลการคำนวณด้วยสมการจำแนกกลุ่ม 62
ตารางที่ 33 การแสดงผลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ 63
ตารางที่ 34 การนำข้อมูลออก (Export data) 64
ตารางที่ 35 การตั้งชื่อไฟล์เอกสารนำออก 65
ตารางที่ 36 การดาวน์โหลดคู่มือการใช้งาน 65
ตารางที่ 37 การแสดงผลภาษาไทยและภาษาอังกฤษของเครื่องมือ 66
ตารางที่ 38 คำอธิบายยูสเคสการคำนวณของเครื่องมือ 73
ตารางที่ 39 คำอธิบายยูสเคสการคำนวณด้วยมาตรวัดเชิงวัตถุ 74
ตารางที่ 40 คำอธิบายยูสเคสการคำนวณหาค่าคุณภาพของการออกแบบ 75
ตารางที่ 41 คำอธิบายยูสเคสการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ .. 76
ตารางที่ 42 คำอธิบายยูสเคสการสร้างโมเดล 77
ตารางที่ 43 คำอธิบายยูสเคสการนำข้อมูลออก 78
ตารางที่ 44 คำอธิบายยูสเคสการเปลี่ยนภาษา 79
ตารางที่ 45 คำอธิบายยูสเคสการดาวน์โหลดคู่มือการใช้งาน 80

สารบัญรูป

หน้า

รูปที่ 1 แสดงโครงสร้างหลักของแผนภาพคลาส 5
รูปที่ 2 ระดับขั้นของโมเดลคุณภาพการออกแบบเชิงวัตถุ 10
รูปที่ 3 แผนภาพขั้นตอนการดำเนินการวิจัย 20
รูปที่ 4 ภาพรวมของการพัฒนาเครื่องมือ 33
รูปที่ 5 แผนภาพยูสเคสของการพัฒนาเครื่องมือ 35
รูปที่ 6 แผนภาพคลาสของการพัฒนาเครื่องมือ 37
รูปที่ 7 รูปแบบเอกสารเอ็กซ์เอ็มแอล 41
รูปที่ 8 ภาพรวมของงานวิจัยการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ 47
รูปที่ 9 ขั้นตอนการติดตั้งเครื่องมือ 88
รูปที่ 10 หน้าจอ Internet Information Service (IIS) Manager 88
รูปที่ 11 การตั้งค่าของเว็บไซต์ 89
รูปที่ 12 แสดงรายชื่อเว็บไซต์ 90
รูปที่ 13 วิธีการเรียกดูเว็บไซต์ 90
รูปที่ 14 เว็บไซต์ THE MAINTAINABILITY ESTIMATION TOOL (MET). 91
รูปที่ 15 ระบบการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ 92
รูปที่ 16 แสดงส่วนประกอบของเมนู Home (หน้าหลัก) 93
รูปที่ 17 แสดงส่วนประกอบของเมนู Helps (ช่วยเหลือ) 93
รูปที่ 18 แสดงส่วนประกอบของเมนู About Us (เกี่ยวกับเรา) 94
รูปที่ 19 การแสดงผลด้วยภาษาไทย 94
รูปที่ 20 ส่วนแสดงผลการคำนวณของเครื่องมือ 95
รูปที่ 21 ขั้นตอนการนำไฟล์เอกสารเอ็กซ์เอ็มแอล 96
รูปที่ 22 ส่วนแสดงผลการคำนวณของเครื่องมือ 97
รูปที่ 23 ส่วนแสดงผลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ 98
รูปที่ 24 การนำข้อมูลออก (Export Data) 99
รูปที่ 25 แสดงหน้าจอการบันทึกข้อมูล 100
รูปที่ 26 การแสดงผลของข้อมูลของเอกสาร Word 100

หน้า

รูปที่ 27 การแสดงผลของข้อมูลเอกสาร Excel..... 101

บทที่ 1
บทนำ

ในบทนี้กล่าวถึงที่มาและความสำคัญของปัญหา วัตถุประสงค์ของงานวิจัย ขอบเขตของการวิจัย ขั้นตอนและวิธีการดำเนินการวิจัยและประโยชน์ที่คาดว่าจะได้รับ โดยมีรายละเอียดดังนี้

1.1 ความเป็นมาและความสำคัญของปัญหา

ในขั้นตอนของการบำรุงรักษาซอฟต์แวร์ (Software Maintenance Phase) เป็นขั้นตอนที่มีความสำคัญอีกขั้นตอนหนึ่งของกระบวนการในการพัฒนาซอฟต์แวร์ เป็นขั้นตอนที่ใช้ระยะเวลา และเสียค่าใช้จ่ายในการดำเนินการมากที่สุด เพราะว่าซอฟต์แวร์หรือระบบต่างๆเมื่อถึงระยะเวลาหนึ่ง จะต้องมีดำเนินการปรับปรุง แก้ไข เพื่อให้สามารถรองรับกับสภาพแวดล้อมในการทำงานใหม่ และความต้องการใหม่ที่เพิ่มขึ้น แต่ปัญหาหลักในการดำเนินการบำรุงรักษาซอฟต์แวร์นั้นคือ กลุ่มผู้พัฒนาระบบ และกลุ่มผู้ทำการบำรุงรักษาซอฟต์แวร์มักไม่ใช่กลุ่มบุคคลเดียวกัน ดังนั้นจึงเป็นเหตุที่ทำให้ผู้ทำการบำรุงรักษาซอฟต์แวร์ขาดความเข้าใจเกี่ยวกับระบบหรือซอฟต์แวร์ที่จะต้องทำการบำรุงรักษา แต่ถ้าหากมีระบบหรือเครื่องมือเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในช่วงของการเริ่มต้นพัฒนาโครงการ หรือในช่วงของการออกแบบซอฟต์แวร์ โดยใช้แผนภาพคลาสก็จะเป็นการช่วยให้ผู้พัฒนาระบบสามารถทบทวน ปรับปรุง หรือแก้ไขการออกแบบของระบบให้ง่ายต่อการบำรุงรักษา โดยผลลัพธ์ในทางอ้อม คือ เป็นการลดค่าใช้จ่าย ระยะเวลาในการดำเนินการ และลดจำนวนคนที่ต้องใช้ในขั้นตอนของการบำรุงรักษาซอฟต์แวร์

จุดประสงค์ของงานวิจัย เพื่อสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ และพัฒนาเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนการออกแบบด้วยแผนภาพคลาส โดยใช้วิธีการวิเคราะห์การจำแนกกลุ่ม (Discriminant Analysis) มาใช้ในการสร้างโมเดล โดยพิจารณาค่าคุณภาพของการออกแบบ ได้แก่ ความสามารถในการนำกลับมาใช้ใหม่ (Reusability) ความสามารถในการยืดหยุ่น (Flexibility) ความสามารถในการ

ทำงาน (Functionality) ความสามารถในการขยาย (Extendibility) และความสามารถในการมีประสิทธิภาพ (Effectiveness) ด้วยโปรแกรมเอสพีเอสเอส และได้พบว่าความสามารถในการยืดหยุ่น (Flexibility) และความสามารถในการขยาย (Extendibility) นั้นเป็นตัวแปรอิสระที่มีเหมาะสมกับการนำไปสร้างเป็นสมการในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ จากนั้นนำเอาสมการที่ได้จากการวิเคราะห์ไปพัฒนาเป็นฟังก์ชันการวัดความสามารถในการบำรุงรักษาให้กับเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

1.2 วัตถุประสงค์ของการวิจัย

1) เพื่อเป็นการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์
2) พัฒนาเครื่องมือเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ด้วยแผนภาพคลาส

1.3 ขอบเขตของการวิจัย

1) คุณภาพของการออกแบบ โดยงานวิจัยนี้ได้ใช้ความสามารถในการยืดหยุ่น (Flexibility) และความสามารถในการขยาย(Extendibility) เป็นเงื่อนไขในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์
2) แผนภาพยูเอ็มแอลที่ใช้ในงานวิจัย คือ แผนภาพคลาส(Class Diagram) เท่านั้น
3) ข้อมูลนำเข้าสู่ระบบจะต้องเป็นข้อมูลแผนภาพคลาสที่ถูกเปลี่ยนอยู่ในรูปแบบเอกสารเอ็กซ์เอ็มแอล(XML) เท่านั้น สำหรับงานวิจัยนี้ได้ใช้โปรแกรมสตาร์ยูเอ็มแอล(StarUML) [6] มาใช้ในการแปลงแผนคลาสให้อยู่ในรูปแบบเอกสารเอ็กซ์เอ็มแอล
4) ระบบที่ใช้ในงานวิจัยมีจำนวน 10 ระบบ
5) แผนภาพคลาสที่จะใช้ในงานวิจัยนี้ จะต้องประกอบด้วยคลาส 5 คลาสขึ้นไป

1.4 ขั้นตอนและวิธีดำเนินการวิจัย

1) ศึกษางานวิจัยเกี่ยวข้องกับการวัดค่าความสามารถในการบำรุงรักษาซอฟต์แวร์

2) ศึกษาปัจจัยที่มีผลต่อคุณภาพในการออกแบบ เช่น ความสามารถในการยืดหยุ่น(Flexibility)และความสามารถในการขยาย (Extendibility)
3) ศึกษามาตรวัดเชิงวัตถุที่สามารถใช้ในการคำนวณด้วยแผนภาพคลาส
4) ศึกษาวิธีการใช้งานโปรแกรมในการสร้างแผนภาพคลาส
5) ศึกษาโครงสร้างภาษาเอ็กซ์เอ็มแอล
6) ศึกษาวิธีการแปลงแผนภาพคลาสให้อยู่ในรูปแบบเอกสารเอ็กซ์เอ็มแอล
7) ศึกษาวิชาสถิติเพื่อใช้ในการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์
8) สร้างแผนภาพคลาสของระบบที่ใช้ในงานวิจัยและแปลงแผนภาพคลาสให้อยู่ในรูปแบบของเอกสารเอ็กซ์เอ็มแอล
9) สร้างโมเดลเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์
10) ออกแบบและพัฒนาเครื่องมือเพื่อใช้ในการทดลอง
11) ทำการทดลอง
12) เก็บรวบรวมข้อมูลที่ได้จากการทดลอง
13) ประเมินผลการทดลอง และสรุปผลการทดลอง
14) จัดทำรายงานวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1) ประโยชน์ทางตรง

ได้โมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบ และได้เครื่องมือเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ด้วยแผนภาพคลาส

2) ประโยชน์ทางอ้อม

เพื่อเป็นการช่วยลดเวลาในการทำงาน ค่าใช้จ่าย และกำลังคนที่ต้องใช้ในขั้นตอนของการบำรุงรักษาซอฟต์แวร์

1.6 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์

ในวิทยานิพนธ์นี้ได้แบ่งเนื้อหาออกเป็น 6 บท ได้แก่ บทที่ 1 บทนำ ได้กล่าวถึงความเป็นมาและความสำคัญของปัญหา รวมถึงวัตถุประสงค์ของงานวิจัยนี้ บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง บทที่ 3 การดำเนินการวิจัย บทที่ 4 การออกแบบและพัฒนาเครื่องมือ บทที่ 5 ผลการทดลอง บทที่ 6 เป็นบทสรุปที่ได้ทั้งหมดจากการดำเนินการวิจัยและร่วมถึงข้อเสนอแนะต่างๆ

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

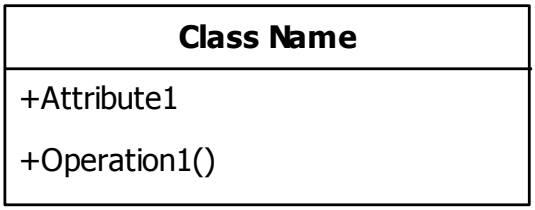
ในบทนี้กล่าวถึงทฤษฎีที่ใช้ในงานวิจัย และงานวิจัยที่เกี่ยวข้องกับการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ซึ่งมีรายละเอียดดังต่อไปนี้

2.1 ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องกับงานวิจัยการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบ ได้แก่ แผนภาพคลาส มาตรวัดเชิงวัตถุ โครงสร้างภาษาเอ็กซ์เอ็มแอล และสถิติสำหรับงานวิจัย มีรายละเอียดดังต่อไปนี้

2.1.1 แผนภาพคลาส [1]

แผนภาพคลาส คือ แผนภาพที่ใช้ในการวิเคราะห์และออกแบบระบบ เป็นแผนภาพเชิงโครงสร้าง เพื่อให้ผู้ใช้งานสามารถมองเห็นองค์ประกอบทั้งหมดของระบบ โดยคลาสจะมีส่วนประกอบ 2 ส่วน ได้แก่ โครงสร้างหลักของแผนภาพคลาส และความสัมพันธ์ของแผนภาพคลาส



รูปที่ 1 แสดงโครงสร้างหลักของแผนภาพคลาส

จากรูปที่ 1แสดงโครงสร้างหลักของแผนภาพคลาส ซึ่งประกอบด้วย

1) ชื่อคลาส (Class Name) คือ ชื่อของคลาส
2) แอตทริบิวต์ (Attribute) คือ คุณลักษณะของคลาส
3) โอเปอเรชัน (Operation) หรือเมธอด (Method) แสดงการทำงานของคลาส

การจำแนกประเภทของโอเปอเรชันตามความสามารถในการมองเห็นและเข้าถึง จะเรียกความสามารถนี้ว่า “Visibility” ซึ่งแบ่งออกเป็น 3 ประเภท มีรายละเอียด ดังนี้

1) ไพรเวทแอตทริบิวต์ (Private Attribute) และโอเปอเรชัน (Operation) คือ แอตทริบิวต์ และโอเปอเรชันที่ไม่สามารถมองเห็นได้จากภายนอก วิธีในการเข้าถึงจะเรียกผ่านโอเปอเรชันที่คลาสมีไว้จะใช้เครื่องหมาย (-) กำกับไว้ที่ไพรเวทแอตทริบิวต์ (Private Attribute) และโอเปอเรชัน (Operation)
2) โพรเทคแอตทริบิวต์ (Protected Attribute) และโอเปอเรชัน (Operation) แอตทริบิวต์ และโอเปอเรชันที่ไม่สามารถเห็นได้จากภายนอกแต่เป็นส่วนที่สามารถส่งต่อให้คลาสที่สืบทอด (Inherited Class) ได้เท่านั้น ใช้เครื่องหมาย (#) กำกับไว้หน้าโพรเทคแอตทริบิวต์ (Protected Attribute) และโอเปอเรชัน (Operation)
3) พลับบลิคแอตทริบิวต์ (Public Attribute) และโอเปอเรชัน (Operation) คือ แอตทริบิวต์ และโอเปอเรชันที่สามารถมองเห็นได้และสามารถเรียกใช้ได้โดยตรงจากภายนอก ใช้เครื่องหมาย (+) กำกับไว้หน้าพลับบลิคแอตทริบิวต์ (Public Attribute) และโอเปอเรชัน (Operation)

ความสัมพันธ์ของแผนภาพคลาส ได้แก่

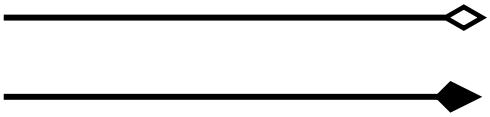
1) ความสัมพันธ์แบบแอสโซซิเอชัน (Association) เพื่อใช้ในการอธิบายความสัมพันธ์ระหว่างคลาสสองคลาส ซึ่งมีอินสแตนท์ (Instance) ของคลาสที่ได้ทำการส่งสาร (Massage) ระหว่างกัน สัญลักษณ์ของความสัมพันธ์แบบแอสโซซิเอชัน แสดงด้วยเส้นทึบไม่มีหัวลูกศร
2) ความสัมพันธ์แบบเจเนอรัลไลเซชัน (Generalization) เป็นความสัมพันธ์ระหว่างคลาสที่มีลักษณะที่เรียกว่า “ซุปเปอร์คลาส (Super class)” หรือ “แพเรนท์คลาส (Parent class) หรือคลาสแม่” และคลาสที่มีลักษณะเฉพาะที่เรียกว่าสับคลาส (Sub class) หรือไชลคลาส (Child class) คือ คลาสลูก ซึ่งมีความสัมพันธ์แบบอีสอะ (Is-a) สัญลักษณ์ของความสัมพันธ์แบบเจเนอรัลไลเซชันจะแสดงด้วยเส้นทึบหัวลูกศรกลวงตรงกลางชี้ไปยังคลาสแม่
3) ความสัมพันธ์แบบแอกกริเกชัน (Aggregation) และคอมโพซิชัน (Composition) เป็นความสัมพันธ์พิเศษชนิดหนึ่งของความสัมพันธ์แบบแอส

โซซิเอชัน แสดงความสัมพันธ์ระหว่างคลาสที่แสดงสิ่งที่ใหญ่กว่า (Whole) ซึ่งประกอบด้วยคลาสที่เล็กกว่า (Parts) หรือจะเรียกว่าเป็นความสัมพันธ์แบบแฮสอะ (Has-a) สัญลักษณ์ของความสัมพันธ์แบบแอก กริเกชัน และคอมโพซิชัน จะแสดงด้วยเส้นทึบหัวรูปขนมเปียกปูนกลวง และทึบตามลำดับ

4) ความสัมพันธ์แบบดีเพนเดนซี (Dependency) เป็นความสัมพันธ์ที่ใช้แสดงในกรณีที่คลาสหนึ่งมีการเปลี่ยนแปลงแล้วมีผลกระทบกับคลาสอื่นด้วย สัญลักษณ์ของความสัมพันธ์แบบดีเพนเดนซี แสดงด้วยเส้นประแบบมีทิศทาง (Directed line)

5) ความสัมพันธ์แบบเรียลไรเซชัน (Realization) ใช้เชื่อมระหว่างส่วนติดต่อผู้ใช้งาน (Interface) และคลาสหรือคอมโพเนนท์ ความสัมพันธ์แบบเรียลไรเซชัน แสดงด้วยสัญลักษณ์เส้นประหัวลูกศรกลวงตรงกลาง

ตารางที่ 1 แสดงความสัญลักษณ์ของความสัมพันธ์ของแผนภาพคลาส

ความสัมพันธ์	สัญลักษณ์
แอสโซซิเอชัน (Association)	
เจเนอรัลไลเซชัน (Generalization)	
แอกกริเกชัน (Aggregation) และ คอมโพซิชัน (Composition)	
ดีเพนเดนซี (Dependency	
เรียลไรเซชัน (Realization)	

2.1.2 มาตรวัดเชิงวัตถุ [10]

ในปัจจุบันการออกแบบเชิงวัตถุได้เป็นที่นิยมแพร่หลาย และนักวิจัยหลายท่านได้นำเสนอมาตรวัดเชิงวัตถุที่สามารถนำมาใช้ในการวัดด้วยแผนภาพคลาส โดยมีรายละเอียดดังนี้

1) มาตรวัด CK metrics

Chidamber และ Kemerer ได้นำเสนอมาตรวัดทั้งหมด 6 มาตรวัด และมี 3 มาตรวัดที่สามารถใช้ในการวัดด้วยแผนภาพคลาส ได้แก่ มาตรวัด Weighted Methods per Class (WMC) มาตรวัด Depth of Inheritance of a class (DIT) และมาตรวัด Number of Children (NOC)

2) มาตรวัด Li and Henry's metrics

Li และ Henry's นำเสนอ ได้แก่ มาตรวัด Number of Attributes in a Class (DAC) มาตรวัด Number of Different Classes (DAC) มาตรวัด Number Of local Methods (NOM) มาตรวัด Number of Attributes รวมกับมาตรวัด Number of Local Methods คือ มาตรวัด SIZE2

3) มาตรวัด MOOD Metrics

Britoe Abreu และ Melo ได้นำเสนอมาตรวัด MOOD ได้แก่ มาตรวัด Method Hiding Factor (MHF) มาตรวัด Attribute Hiding Factor (AHF) มาตรวัด Method Inheritance Factor (MIF) มาตรวัด Attribute Inheritance Factor (AIF) และมาตรวัด Polymorphism Factor (PF)

4) มาตรวัด Lorenz and Kidd's metrics

Lorenz และ Kidd's ได้นำเสนอมาตรวัดเชิงวัตถุ แบ่งออกได้เป็น 3 กลุ่ม ได้แก่ มาตรวัดสำหรับการหาขนาดของคลาส (Class size metrics) มาตรวัดสำหรับการสืบทอดของคลาส (Class inheritance) และมาตรวัดภายในคลาส (Class internals metrics)

5) มาตรวัด Briand et al.’s metrics

Briand et al.’s ได้นำเสนอมาตรวัด ได้แก่ มาตรวัด ACAIC มาตรวัด OCAIC มาตรวัด DCAEC มาตรวัด OCAEC มาตรวัด ACMIC มาตรวัด OCMIC มาตรวัด DCMEC และ มาตรวัด OCMEC

6) มาตรวัด Bansiya et al.’s metrics

Bansiya et al.’s ได้นำเสนอมาตรวัดทั้งหมด 11 มาตรวัด และสามารถนำเอาไปใช้ในการวัดด้วยแผนภาพคลาสได้ ได้แก่ มาตรวัด Data Access Metrics (DAM) มาตรวัด Direct Class Coupling (DCC) มาตรวัด Cohesion Among Method of Class (CAMC) มาตรวัด Measure Of Aggregation (MOA) มาตรวัด Measure of Functional Abstraction (MFA) มาตรวัด Number of Classes in the design (DSC) มาตรวัด Number of Class Hierarchies in the design (NOH) มาตรวัด Average Number of Ancestors (ANA) และ มาตรวัด Number Of Polymorphic methods (NOP)

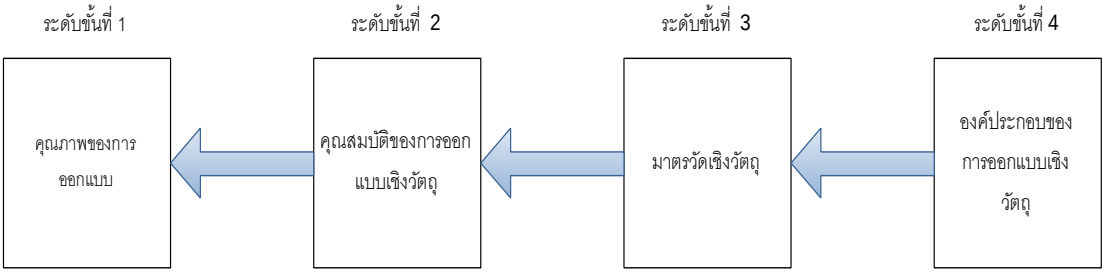
7) มาตรวัด Genero et al.’s metrics

Genero et al.’s ได้นำเสนอมาตรวัดที่ใช้ในการวัดด้วยแผนภาพคลาส โดยแบ่งออกเป็น 2 กลุ่ม คือ Class – scope metrics ได้แก่ มาตรวัด Number of Association metric (NAssoc) มาตรวัด Number of Aggregation metric (NAgg) มาตรวัด Number of Dependencies metric (NDep) มาตรวัด Number of Generalization metric (NGen) มาตรวัด Number of Generalization Hieratchies metric (NGenH) มาตรวัด Number of Aggregation Hierarchies metric (NAggH) มาตรวัด Maximum DIT metric (MaxDIT) มาตรวัด Maximum HAgg metric (MaxHAgg) และกลุ่มของ Class-diagram scope metrics ได้แก่ มาตรวัด Number of Association per Class metric (NAssocC) มาตรวัด Height of a class within an aggregation hierarchy (HAgg) มาตรวัด Number of Direct Parts metric (NODP) มาตรวัด Number of Part metric (NP) มาตรวัด Number of Wholes metric (NW) มาตรวัด Multiple Aggregation metric (MAgg) มาตรวัด Number of

Dependencies In metric (NDepIn) มาตรวัด Number of Dependencies Out metric (NDepOut)

2.1.3 โมเดลคุณภาพสำหรับการออกแบบเชิงวัตถุ (Quality Model for Object-Oriented Design: QMOOD) [4]

Jagdish Bansiya และ Carl G Davis ได้นำเสนอโมเดลคุณภาพสำหรับการออกแบบเชิงวัตถุ (Quality Model for Object Oriented Design – QMOOD) ด้วยการขยายความจากโมเดลของ Dromey's รูปที่ 2 แสดงระดับขั้นของโมเดลคุณภาพสำหรับการออกแบบเชิงวัตถุ



รูปที่ 2 ระดับขั้นของโมเดลคุณภาพการออกแบบเชิงวัตถุ

ระดับขั้นที่ 1 คุณภาพของการออกแบบ (Design Quality Attribute) [4][12]

ตารางที่ 2 แสดงคุณภาพของการออกแบบ

คุณภาพของการออกแบบ	**คำนิยาม**
ความสามารถในการนำกลับมาใช้ใหม่ (Reusability)	ความสามารถที่จะนำเอาการออกแบบของระบบ หรือซอฟต์แวร์เดิมนำกลับมาใช้ในการแก้ปัญหาใหม่ โดยที่ไม่ต้องใช้ความพยายาม (Effort)

คุณภาพของการออกแบบ	คำนิยาม
ความสามารถในการยืดหยุ่น (Flexibility)	ความสามารถในการปรับเปลี่ยนการออกแบบให้เข้ากับฟังก์ชันของการทำงานได้
ความสามารถในการทำงาน (Functionality)	ความรับผิดชอบของคลาสทั้งหมด จะต้องมีลักษณะเป็นที่เป็นสาธารณะ
ความสามารถในการขยาย (Extendibility)	การออกแบบระบบสามารถที่จะรองรับกับการความต้องการใหม่ที่เพิ่มมากขึ้นได้
ความสามารถในการมีประสิทธิภาพ (Effectiveness)	การออกแบบระบบจะต้องได้ผลตรงกับฟังก์ชันการทำงานของการออกแบบเชิงวัตถุและเชิงเทคนิค

ระดับขั้นที่ 2 คุณสมบัติของการออกแบบเชิงวัตถุ (Object Oriented Design Properties) [4][12]

ตารางที่ 3 แสดงคุณสมบัติของการออกแบบเชิงวัตถุ

คุณสมบัติของการออกแบบเชิงวัตถุ
ขนาดของการออกแบบ (Design Size)
ลำดับชั้น (Hierarchy)
นามธรรม (Abstraction)
การห่อหุ้ม (Encapsulation)
การเข้าคู่ (Coupling)
การยึดเหนี่ยว(Cohesion)
ส่วนประกอบ (Composition)
การถ่ายทอดคุณลักษณะ(Inheritance)
การเปลี่ยนรูป (Polymorphism)
การส่งสาร (Messaging)
ความซับซ้อน (Complexity)

ระดับขั้นที่ 3 มาตรวัดของการออกแบบเชิงวัตถุ (Object Oriented Design Metrics) [4][12]

ตารางที่ 4 แสดงมาตรวัดเชิงวัตถุ

มาตรวัด	ชื่อมาตรวัด
DSC	Design Size in Class
NOH	Number of Hierarchies
ANA	Average Number of Ancestors
DAM	Data Access Metric
DCC	Direct Class Coupling
CAM	Cohesion Among Methods of Class
MOA	Measure of Aggregation
MFA	Measure of Functional Abstraction
NOP	Number of Polymorphic Methods
CIS	Class Interface Size
NOM	Number of Methods

ระดับขั้นที่ 4 องค์ประกอบของการออกแบบเชิงวัตถุ (Object Oriented Design Components) [4] [12]

ตารางที่ 5 แสดงคุณสมบัติของการออกแบบเชิงวัตถุ

คุณสมบัติของการออกแบบเชิงวัตถุ	**ชื่อมาตรวัด**
ขนาดของการออกแบบ (Design Size)	Design Size in Class: DSC
ลำดับชั้น (Hierarchy)	Number of Hierarchies: NOH
นามธรรม (Abstraction)	Average Number of Ancestors: ANA
การห่อหุ้ม (Encapsulation)	Data Access Metric: DAM
การเข้าคู่ (Coupling)	Direct Class Coupling: DCC
การยึดเหนี่ยว (Cohesion)	Cohesion Among Methods of Class: CAM
ส่วนประกอบ (Composition)	Measure of Aggregation: MOA
การถ่ายทอดคุณลักษณะ (Inheritance)	Measure of Functional Abstraction: MFA
การเปลี่ยนรูป (Polymorphism)	Number of Polymorphic Methods: NOP
การส่งสาร (Messaging)	Class Interface Size: CIS
ความซับซ้อน (Complexity)	Number of Methods: NOM

ตารางที่ 6 แสดงความสัมพันธ์ระหว่างคุณสมบัติและคุณภาพของการออกแบบ [4] [12]

	การนำกลับมาใช้ใหม่ (Reusability)	การยืดหยุ่น (Flexibility)	การทำงาน (Functionality)	การขยาย (Extendibility)	การมีประสิทธิภาพ (Effectiveness)
ขนาดของการออกแบบ (Design Size)			↑		
ลำดับชั้น (Hierarchy)			↑		
นามธรรม (Abstraction)				↑	↑
การห่อหุ้ม (Encapsulation)		↑			↑
การเข้าคู่ (Coupling)	↓	↓		↓	
การยึดเหนี่ยว (Cohesion)	↑		↑		
ส่วนประกอบ (Composition)		↑			↑
การถ่ายทอดคุณลักษณะ(Inheritance)				↑	↑
การเปลี่ยนรูป (Polymorphism)		↑	↑	↑	↑
การส่งสาร (Messaging)	↑		↑		

โดยสัญลักษณ์ ↑ แสดงความสัมพันธ์เชิงบวก และสัญลักษณ์ ↓ แสดงความสัมพันธ์เชิงลบ

2.1.4 **ภาษาเอ็กซ์เอ็มแอล** (Extensive Markup Language - XML)

ภาษาเอ็กซ์เอ็มแอลถูกออกแบบมาเพื่อใช้ในการอธิบายข้อมูลของวัตถุ และพฤติกรรม (Behavior) โดยเป็นภาษาที่มีพื้นฐานมาจากเอสจีเอ็มแอล (Standard Generalized Markup Language - SGML) โดยประกอบด้วย 2 ส่วน คือ ดีทีดี (Document Type Definition - DTD) เป็นคำอธิบายชนิดของเอกสารเอ็กซ์เอ็มแอลที่ทำหน้าที่กำหนดไวยากรณ์ของเอกสาร ส่วนที่สอง คือ ในส่วนของเนื้อหาของเอกสารเอ็กซ์เอ็มแอล (XML document) ซึ่งเป็นโครงสร้างทางตรรกะ (Logical structure) ใช้อธิบายคุณลักษณะต่างๆให้สอดคล้องกับโครงสร้างของดีทีดี เอ็กซ์เอ็มแอลได้แบ่งส่วนของดีทีดี และเนื้อหาของเอกสารเอ็กซ์เอ็มแอลออกจากส่วน

ของการแสดงผล ทำให้ได้เอกสารที่ดีโครงร่าง (Schema) เป็นโครงสร้างที่ง่ายต่อการนำข้อมูลไปใช้งาน และสามารถนำกลับมาใช้ใหม่ได้หลายๆครั้ง

เอ็กซ์เอ็มแอล (XML) เป็นเอกสารที่มีความยืดหยุ่นสำหรับงานประยุกต์ที่มีพื้นฐานบนเว็บ และมีรูปแบบการนำเสนอในรูปแบบข้อความ จึงไม่ขึ้นกับระบบปฏิบัติการ

2.1.5 **สถิติสำหรับงานวิจัย** (Statistics for Research)

สำหรับงานวิจัยนี้ได้นำเอาวิธีการทางสถิติเข้ามาช่วยในการทำงานวิจัยและวิเคราะห์ข้อมูล ได้แก่ การหาค่าเฉลี่ย (Mean) การวิเคราะห์ความถดถอย และการวิเคราะห์จำแนกกลุ่ม (Discriminant Analysis)

2.1.5.1 **การหาค่าเฉลี่ย** [7]

การหาค่าเฉลี่ย เป็นการวัดแนวโน้นเข้าสู่ส่วนกลาง (Central tendency) ซึ่งเป็นการคำนวณค่ากลางของข้อมูล โดยจะใช้ค่าเฉลี่ยเป็นตัวแทนของข้อมูลที่นำมาคำนวณดังสมการที่ (1)

$$\overline{x} = \frac{\sum_{i=1}^{n} xi}{n} = \frac{x1+x2+\cdots+xn}{n} \qquad (1)$$

โดยที่ $\overline{x}$ คือ ค่าเฉลี่ยของข้อมูลชุดใด ๆ

$X_{i\ คือ}$ ค่าข้อมูลในลำดับที่ I เมื่อ $1 \leq I \leq n$

n คือ จำนวนข้อมูลทั้งหมด

2.1.5.2 การวิเคราะห์การถดถอยเชิงพหุ [7] [8]

การวิเคราะห์ความถดถอยเชิงพหุ เป็นการวิเคราะห์ความสัมพันธ์ของตัวแปรหลายตัวแปร ประกอบด้วย ตัวแปรตาม (Dependent variable) 1 ตัว ต้องเป็นตัวแปรเชิงปริมาณ และตัวแปรอิสระ (Independent variable) จำนวน k ตัว ($k \geq 2$) โดยมีความสัมพันธ์ในรูปเชิงเส้น ดังสมการที่ (2)

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k + e \qquad (2)$$

กำหนดให้ γ เป็นตัวแปรตาม

β_0 เป็นส่วนตัดแกน γ

e คือความคลาดเคลื่อนอย่างสุ่ม

$\beta_{0,\ldots,}, \beta_k$ เป็นสัมประสิทธิ์ความถดถอยเชิงส่วน (Partial Regression Coefficient)

เทคนิคการเลือกตัวแปรอิสระเข้าสมการความถดถอย เพื่อเป็นการเลือกตัวแปรอิสระที่มีความสัมพันธ์กับตัวแปรตามมี 4 เทคนิค ดังนี้

1) วิธีการนำเอาตัวแปรอิสระเข้าสมการทั้งหมด (Enter) เป็นวิธีการเลือกตัวแปรอิสระเข้าสมการความถดถอย โดยผู้วิจัยจะต้องทำการทดสอบสมมติฐานเพื่อตรวจสอบความสัมพันธ์ระหว่างตัวแปรต้นกับตัวแปรตาม โดยใช้ F-testจากตาราง ANOVA และ t-test โดยผู้วิจัยจะต้องสรุปผลการทดสอบว่าควรจะมีตัวแปรอิสระใดบ้างที่มีความสัมพันธ์กับตัวแปรตาม โดยใช้ผลการทดสอบ F-test และ t-testตามลำดับ
2) วิธีการเลือกแบบก้าวหน้า (Forward Selection) เป็นวิธีการพิจารณาว่าตัวแปรอิสระที่ใดบ้างที่มีความสัมพันธ์กับตัวแปรตามโดยจะเลือกตัวแปรอิสระเข้าสมการความถดถอยครั้งละ 1 ตัว ซึ่งจะทำการเลือกหลายๆ ครั้ง ในแต่ละครั้งจะเลือกตัวแปรอิสระเข้าสมการเพียง 1 ตัว โดยเลือกตัวแปรอิสระตัวที่ทำให้ความสัมพันธ์กับตัวแปรตาม Y เพิ่มขึ้นมากที่สุด และจะหยุดนำตัวแปรอิสระเข้าสมการ เมื่อนำเข้ามาเพิ่มแล้วไม่ได้ทำให้ความสัมพันธ์กับตัวแปรตามเพิ่มมากขึ้น

3) วิธีการเลือกแบบถอยหลัง (Backward Selection) เป็นวิธีการเลือกตัวแปรที่ตรงข้ามกับวิธีการเลือกแบบก้าวหน้า โดยจะทำการตัดตัวแปรอิสระที่ไม่มีความสัมพันธ์กับ Y ออกจากสมการความถดถอยครั้งละ 1 ตัว
4) วิธีการเลือกแบบลำดับขั้น (Stepwise Selection) เป็นวิธีการเลือกตัวแปรอิสระเข้าสมการ โดยใช้หลักการทั้งวิธีการเลือกแบบก้าวหน้า และวิธีการเลือกแบบถอยหลัง ใช้หลักการของวิธีการเลือกแบบก้าวหน้า โดยเลือกตัวแปรอิสระที่มีความสัมพันธ์กับตัวแปรตามมากที่สุดและทดสอบแล้วพบว่ามีความสัมพันธ์เป็นจริง ดังนั้นจึงมีตัวแปรอิสระในสมการ 1 ตัว ใช้หลักการของวิธีการเลือกแบบถอยหลังโดยเลือกตัวแปรอิสระตัวที่สองเข้าสมการซึ่งพิจารณาจากตัวแปรที่เหลือทั้งหมดว่าตัวใดมีความสัมพันธ์กับ Y มากที่สุดแล้วนำเข้าสมการ

2.1.5.3 **การวิเคราะห์จำแนกกลุ่ม** (Discriminant Analysis) [7][8]

การวิเคราะห์จำแนกกลุ่ม เป็นเทคนิคที่ทำการแบ่งกลุ่มข้อมูล หรือหน่วยตัวอย่างออกเป็นกลุ่มย่อยๆ หลายๆกลุ่ม ซึ่งใช้กลักเกณฑ์ของการวิเคราะห์ความถดถอยเชิงพหุในรูปเชิงเส้น โดยที่ตัวแปรตามเป็นตัวแปรเชิงกลุ่ม ตัวแปรต้นเป็นตัวแปรเชิงปริมาณ โดยนำเอาตัวแปรเหล่านี้มาศึกษาหาความสัมพันธ์ที่อยู่ในรูปเชิงเส้น จากนั้นนำสมการเชิงเส้นมาพยากรณ์

ตัวแปรอิสระ (Independent) ที่ทำให้กลุ่มแตกต่างกันเรียกว่าตัวแปรจำแนกกลุ่ม (Discriminator variable) ควรเป็นตัวแปรเชิงปริมาณ ซึ่งอาจมีเพียง 1 หรือ 2 ตัวขึ้นไป ความสัมพันธ์ระหว่างตัวแปรตามกับตัวแปรจำแนกกลุ่มจะรูปในรูปเชิงเส้น ดังสมการที่ (3)

$$D = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + e \qquad (3)$$

โดยที่ D คือ ตัวแปรตามและเป็นข้อมูลเชิงกลุ่ม

$x_1, x_2, \ldots, x_n$ คือตัวแปรอิสระหรือตัวแปรจำแนกกลุ่ม เมื่อ $n \geq 1$

$\beta_0, \beta_1, \ldots, \beta_n$ คือ ค่าสัมประสิทธิ์ของตัวแปรอิสระ เมื่อ $n \geq 0$

e คือ ค่าความคลาดเคลื่อน

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 A hierarchical model for object-oriented design quality assessment. [4]

งานวิจัยของ Jagdish Bansiya และ Carl G Davis ได้นำเสนอโมเดลคุณภาพสำหรับการออกแบบเชิงวัตถุ (Quality Model for Object Oriented Design - QMOOD) โดยแบ่งเป็นระดับขั้นมีทั้งหมด 4 ระดับขั้นได้แก่ ระดับขั้นที่ 1 คุณภาพของการออกแบบ (Design Quality Attribute) ระดับขั้นที่ 2 คุณสมบัติของการออกแบบเชิงวัตถุ (Object Oriented Design Properties) ระดับขั้นที่ 3 มาตรวัดของการออกแบบเชิงวัตถุ (Object Oriented Design Metrics) และระดับขั้นที่ 4 องค์ประกอบของการออกแบบเชิงวัตถุ (Object Oriented Design Components)

Jagdish Bansiya และ Carl G Davis ได้นำเสนอคุณภาพของการออกแบบจำนวน 5 ค่า ได้แก่ ความสามารถในการนำกลับมาใช้ใหม่ (Reusability) ความสามารถในการยืดหยุ่น (Flexibility) ความสามารถในการทำงาน (Functionality) ความสามารถในการขยาย (Extendibility) และความสามารถในการมีประสิทธิภาพ (Effectiveness) นำเสนอมาตรวัดเชิงวัตถุ 11 วัด ได้แก่ มาตรวัด Design Size in Class (DSC) มาตรวัด Number of Hierarchies (NOH) มาตรวัด Average Number of Ancestors(ANA) มาตรวัด Data Access Metric (DAM) มาตรวัด Direct Class Coupling (DCC) มาตรวัด Cohesion Among Methods of Class (CAM) มาตรวัด Measure of Aggregation (MOA) มาตรวัด Measure of Functional Abstraction (MFA) มาตรวัด Number of Polymorphic Methods (NOP) มาตรวัด Class Interface Size (CIS) มาตรวัด Number of Methods (NOM)

ผลลัพธ์ที่ได้จากงานวิจัยนี้ ได้โมเดลคุณภาพสำหรับการออกแบบเชิงวัตถุ โดยมีสมการการคำนวณคุณภาพของการออกแบบ และได้มาตรวัดใหม่ 5 มาตรวัด ได้แก่ มาตรวัด DAM มาตรวัด DCC มาตรวัด CAMC มาตรวัด MOA และมาตรวัด MFA

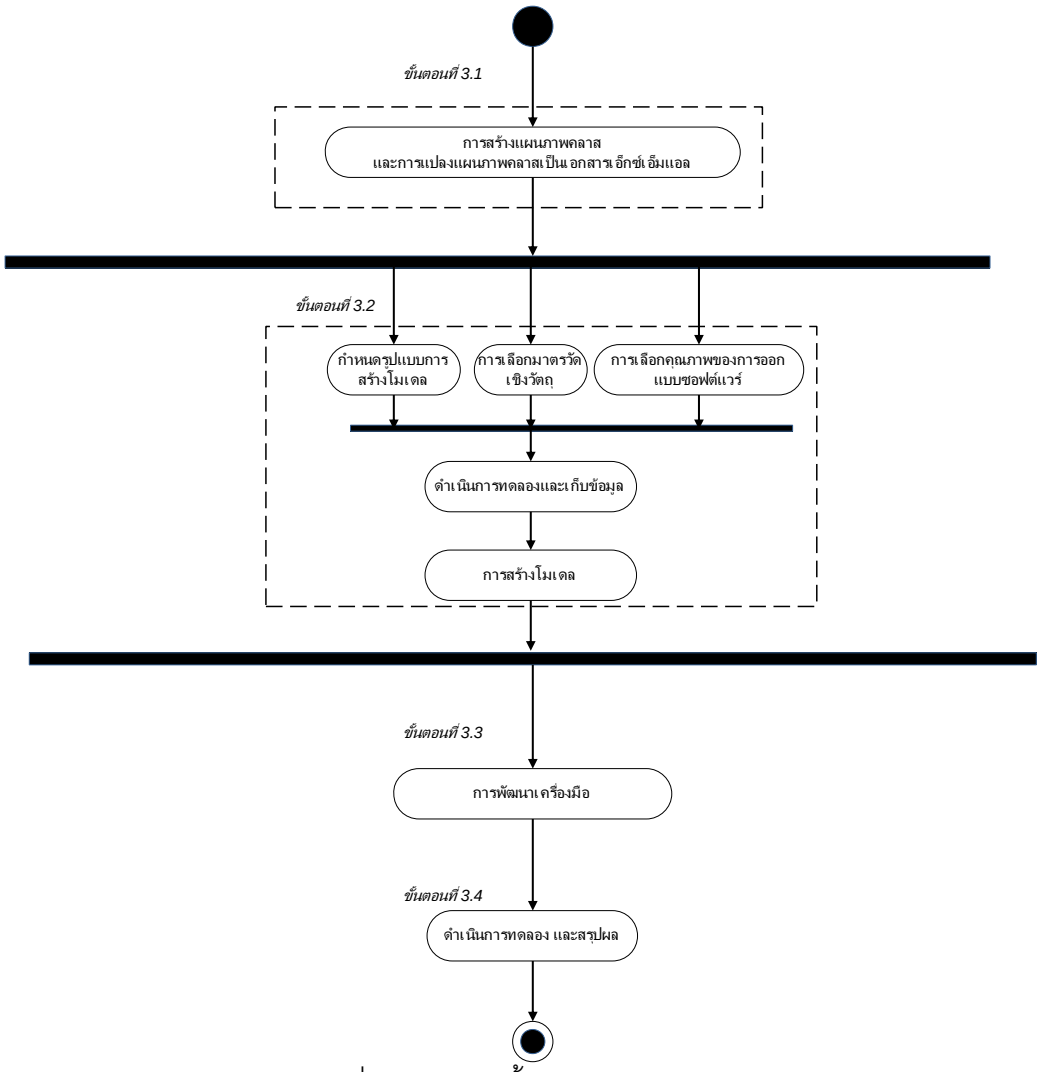
2.2.2 Maintainability Estimation Model for Object-Oriented Software in Design Phase (MEMOOD). [5]

Rizvi และคณะ ได้ทำการวิจัยในเรื่องการประมาณการความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบ โดยสร้างแบบจำลองเชิงเส้นหลายตัวแปร (Multivariate linear model) ใช้ในการประมาณการความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบ (Maintainability Estimation Model for Object-Oriented software in Design phase (MEMOOD)) เพื่อเป็นการช่วยผู้ออกแบบระบบได้ปรับปรุง แก้ไขงานได้ทันก่อนการส่งมอบงานโดยโมเดลการประมาณการนี้จะพิจารณาปัจจัยย่อยจากความสามารถในการเข้าใจ (Understandability) และความสามารถในการปรับเปลี่ยน (Modifiability) มาคำนวณด้วยมาตรตวัดเชิงวัตถุ (Object-Oriented metrics) ซึ่งคำนวณได้จากแผนภาพคลาส (Class diagrams) และสร้างโมเดลด้วยการวิเคราะห์การถดถอยเชิงเส้นแบบพหุ (Multiple Linear Regression) มาใช้ในการคำนวณความสามารถทั้งสามด้าน โดยการประเมินโมเดลจะใช้วิธีโดยนำเอาค่าที่ได้จากการคำนวณจากโมเดลมาเปรียบเทียบกับค่าจริง (Actual)

โดยผลลัพธ์ที่ได้จากวิจัยนี้ คือ ได้วิธีในการประมาณการหาค่าความสามารถในการบำรุงรักษาโดยใช้สมการเชิงเส้นหลายตัวแปร ได้แนวทางวิธีในการคำนวณหาค่าความสามารถในการทำความเข้าใจ และความสามารถในการปรับเปลี่ยน แต่ในงานวิจัยนี้ยังขาดวิธีในการเลือกตัวแปรอิสระ (Independent variables) ที่ใช้ในการสร้างโมเดลและยังขาดการแสดงข้อมูลคุณลักษณะที่สำคัญของการออกแบบเชิงวัตถุ ได้แก่ ขนาดของการออกแบบ (Design Size) ลำดับชั้น (Hierarchy) นามธรรม (Abstraction) การห่อหุ้ม (Encapsulation) การเข้าคู่ (Coupling) การยึดเหนี่ยว (Cohesion) ส่วนประกอบ (Composition) การถ่ายทอดคุณลักษณะ (Inheritance) การเปลี่ยนรูป (Polymorphism) การส่งสาร (Messaging) และความซับซ้อน (Complexity) เป็นต้น

บทที่ 3
การดำเนินการวิจัย

งานวิจัยนี้ได้จัดลำดับขั้นตอนในการดำเนินการวิจัยออกเป็น 4 ขั้นตอน สามารถแสดงด้วยแผนภาพแอคทิวิตีดังแสดงในรูปที่ 3



รูปที่ 3 แผนภาพขั้นตอนการดำเนินการวิจัย

รูปที่ 3 แสดงแผนภาพขั้นตอนการดำเนินการวิจัยนี้ ในขั้นตอนที่ 3.1 การสร้างแผนภาพคลาสและการแปลงแผนภาพคลาสให้อยู่ในรูปแบบเอกสารเอ็กซ์เอ็มแอล ขั้นตอนที่ 3.2 การดำเนินการวิจัย ประกอบด้วยขั้นตอนการกำหนดรูปแบบการสร้างโมเดล การเลือกมาตรวัดเชิงวัตถุ การเลือกคุณภาพของการออกแบบ

ซอฟต์แวร์ การดำเนินการทดลองและเก็บรวบรวมผลการทดลอง และการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ในขั้นตอนที่ 3.3 การพัฒนาเครื่องมือเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ และขั้นตอนที่ 3.4 เป็นการทดลองและสรุปผลที่ได้จากงานวิจัยนี้

3.1 การสร้างแผนภาพคลาสและการแปลงแผนภาพคลาสในรูปแบบเอกสารเอ็กซ์เอ็มแอล

ในขั้นตอนนี้เป็นการสร้างแผนภาพคลาสและแปลงเอกสารจากแผนภาพคลาสให้อยู่ในรูปแบบเอกสารเอ็กซ์เอ็มแอล โดยมีรายชื่อระบบทั้งหมดที่ใช้ในงานวิจัยนี้ตามตารางที่ 7

ตารางที่ 7แสดงรายชื่อระบบและจำนวนคลาสเพื่อใช้สร้างโมเดล

	ระบบ	จำนวนคลาส
1	Airline Reservation System [15]	6
2	ATM System [18]	10
3	Banks System [1]	9
4	E-Ticket [16]	10
5	Hospital Management [19]	9
6	Internet Cafe [17]	20
7	Laboratory System [20]	12
8	Library Management [21]	6
9	Web Base Auction System [14]	10
10	Hotel Management System [13]	8

3.2 การดำเนินการวิจัย

ในขั้นตอนนี้เป็นการดำเนินการวิจัย และกำหนดขั้นตอนในการวิจัย ได้แก่ การกำหนดรูปแบบโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ การเลือกมาตรวัดเชิงวัตถุ การเลือกคุณภาพของการออกแบบและการดำเนินการทดลอง

3.2.1 การกำหนดรูปแบบโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

สำหรับงานวิจัยนี้ได้เลือกวิธีการจำแนกกลุ่ม (Discriminant Analysis) มาใช้เป็นรูปแบบในการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ซึ่งโมเดลนี้จะแบ่งระดับของความสามารถในการบำรุงรักษาซอฟต์แวร์ออกเป็น 2 ระดับ คือ ระดับง่าย (Easy Level) และระดับยาก (Difficult Level) และมีรูปแบบของโมเดลในการทำนาย ดังสมการที่ (1)

$$D = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + e \qquad (1)$$

โดยที่ ค่า D (ตัวแปรตาม) คือ ระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ ได้แก่ ระดับง่าย (Easy Level) และระดับยาก (Difficult Level)

ค่า x_i (ตัวแปรต้น) คือ ค่าคุณภาพของการออกแบบซอฟต์แวร์ เมื่อ $1 \leq i \leq n$

ค่า β_i คือ ค่าสัมประสิทธิ์ของตัวแปรอิสระ เมื่อ $1 \leq i \leq n$

ค่า e คือ ค่าคงที่

3.2.2 การเลือกมาตรวัดเชิงวัตถุ

ในขั้นตอนนี้เป็นการเลือกมาตรวัดเชิงวัตถุที่สามารถนำมาใช้ในการคำนวณด้วยแผนภาพคลาสได้ โดยมาตรวัดเชิงวัตถุที่นำมาใช้ในงานวิจัยนี้ คือ มาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด [4] [9] [10] [12] ได้แก่ มาตรวัด Data Access Metrics (DAM) มาตรวัด Direct Class Coupling (DCC) มาตรวัด Cohesion Among Method of Class (CAMC) มาตรวัด Measure Of Aggregation (MOA) มาตรวัด Measure of Functional Abstraction (MFA) มาตรวัด Number of Classes in the Design (DSC) มาตรวัด Number Of Class Hierarchies in the design (NOH) มาตรวัด Average Number of Ancestors (ANA) และ มาตรวัด Number of Polymorphic methods (NOP) มาใช้ในการคำนวณด้วยแผนภาพคลาส ตารางที่ 8 แสดงคุณสมบัติของการออกแบบและมาตรวัดเชิงวัตถุที่ใช้ในการวัด

ตารางที่ 8 แสดงคุณสมบัติของการออกแบบและมาตรวัดเชิงวัตถุ [2][4][12]

คุณสมบัติการออกแบบ	มาตรวัดเชิงวัตถุ
ขนาดของการออกแบบ(Design Size)	Design Size in Class (DSC)
ลำดับชั้น (Hierarchy)	Number Of Hierarchies(NOH)
นามธรรม (Abstraction)	Average Number of Ancestors (ANA)
การห่อหุ้ม (Encapsulation)	Data Access Metric (DAM)
การเข้าคู่ (Coupling)	Direct Class Coupling (DCC)
การยึดเหนี่ยว (Cohesion)	Cohesion Among Methods in Class (CAM)
ส่วนประกอบ(Composition)	Measure Of Aggregation (MOA)
การถ่ายทอดคุณลักษณะ(Inheritance)	Measure of Functional Abstraction (MFA)
การเปลี่ยนรูป (Polymorphism)	Number Of Polymorphic Methods (NOP)
การส่งสาร (Messaging)	Class Interface Size (CIS)
ความซับซ้อน (Complexity)	Number Of Methods (NOM)

3.2.2.1 คำนิยามของมาตรวัด

1) มาตรวัด Design Size in Classes (DSC) เป็นการนับจำนวนคลาสทั้งหมดที่ใช้ในการออกแบบ

2) มาตรวัด Number Of Hierarchies (NOH) เป็นการนับจำนวนของลำดับชั้นของคลาสทั้งหมดที่ใช้ในการออกแบบ
3) มาตรวัด Average Number of Ancestors (ANA) เป็นการหาค่าเฉลี่ยของจำนวนคลาสที่มีการสืบทอดข้อมูลต่อจำนวนคลาสที่มีการสืบทอดทั้งหมด
4) มาตรวัด Data Access Metric (DAM) เป็นการหาอัตราส่วนของจำนวนคุณลักษณะไพรเวท และคุณลักษณะโพรเทคที่ประกาศในคลาสต่อจำนวนคุณลักษณะที่ประกาศไว้ทั้งหมด
5) มาตรวัด Direct Class Coupling (DCC) เป็นการนับจำนวนการประกาศคุณลักษณะ (Attribute declaration) ทั้งหมดภายในคลาสต่อจำนวนคุณลักษณะที่ประกาศทั้งหมดในระบบ
6) มาตรวัด Cohesion Among Methods of Class (CAMC) โดยอ้างอิงมาจาก KuljitKaur [9]. มาตรวัดนี้เป็นการคำนวณการยึดเหนี่ยวของการเรียกใช้คุณลักษณะภายในคลาสโดยมีรูปแบบสมการคือ

$$\text{CAMC(C)} = \frac{\sigma}{kl} \text{ where } \sigma = \sum_1^{kl} PO[i][j]$$

โดยที่ CAMC คือ มาตรวัด Cohesion among Methods of Class

$\sigma = \sum_1^{kl} PO[i][j]$ คือ แทนค่าผลรวมของเมตริก

i คือ จำนวนคุณลักษณะที่ใช้ภายในคลาส

j คือ เมธอดที่เรียกใช้ภายในคลาส โดยนับจำนวนค่า1 คือค่าคุณลักษณะที่มีการเรียกใช้ด้วยเมธอดของคลาสเอง

7) มาตรวัด Measure Of Aggregation (MOA) มาตรวัดนี้ใช้ในการวัดความสัมพันธ์ part-whole โดยเป็นการนับจำนวนคุณลักษณะทั้งหมดที่ประกาศภายในคลาส
8) มาตรวัด Measure of Functional Abstraction metric (MFA) มาตรวัดอัตราส่วนของจำนวนเมธอดที่ถูกโอเวอร์ไรด์ต่อจำนวนรวมของเมธอดที่สามารถเข้าถึงได้โดยสมาชิกของเมธอดของคลาส

9) มาตรวัด Number Of Polymorphic Methods (NOP) มาตรวัดจำนวนเมธอดที่สามารถเปลี่ยนรูปได้เป็นการนับจำนวนเมธอดที่มีประกาศเป็นโอเวอร์ไรด์เมธอด
10) มาตรวัด Class Interface Size (CIS) มาตรวัดนี้คือการนับจำนวนของพลับบลิคเมธอดทั้งหมดภายในคลาส
11) มาตรวัด Number Of Methods (NOM) นับจำนวนเมธอดทั้งหมดที่กำหนดในคลาส ได้แก่ พลับบลิคเมธอด โพรเทคเมธอดและไพรเวทเมธอด

ตารางที่ 9 มาตรวัดเชิงวัตถุที่ใช้ในงานวิจัย

ชื่อมาตรวัดเชิงวัตถุ	
Design Size in Classes(DSC) [4],[12]	*NumberOfClass*
Number Of Hierarchies(NOH) [10],[12]	*NumberOfHieratchies*
Average Number of Ancestors (ANA) [12]	$\frac{NumberofInheritance}{NumberClassallpath}$
Data Access Metric(DAM) [4],[10]	$\frac{NumberofPrivateAttribute}{NumberofAttribute}$
Direct Class Coupling (DCC) [10]	*NumberofInstability*
Cohesion Among Methods of Class(CAMC) [9]	CAMC(C) = $\frac{\sigma}{kl}$ where $\sigma = \sum_1^{kl} PO[i][j]$
Measure Of Aggregation (MOA) [12]	1- $\frac{numberofOverridenMethods}{NumberofMethodes}$
Number Of Polymorphic Methods (NOP) [4],[10],[12]	*NumberOfOverideMethod*
Class Interface Size (CIS) [4]	*NumberofPublicMethod*
Number of Methods (NOM) [4]	*NumberOfMethod*

3.2.3 การเลือกคุณภาพของการออกแบบ

งานวิจัยนี้ได้เลือกคุณภาพของการออกแบบที่มีผลต่อการบำรุงรักษาซอฟต์แวร์[3][4][11] ได้แก่ ความสามารถในการนำกลับมาใช้ใหม่(Reusability) ความสามารถในการยืดหยุ่น(Flexibility) ความสามารถในการทำงาน (Functionality) ความสามารถในการขยาย(Extendibility) และความสามารถในการมีประสิทธิภาพ (Effectiveness) มาใช้เป็นตัวแปรอิสระเพื่อใช้ในการโมเดลการทำนายโดยมีสมการเพื่อใช้ในการคำนวณหาค่าคุณภาพของการออกแบบ ตามตารางที่ 10 แสดงสมการการคำนวณคุณภาพของการออกแบบ

ตารางที่ 10 สมการการคำนวณคุณภาพของการออกแบบ [3][12]

ค่าคุณภาพ	สมการในการคำนวณ
ความสามารถในการนำกลับมาใช้ใหม่	-0.25*Coupling + 0.25*Cohesion+0.5*Messaging
ความสามารถในการยืดหยุ่น	0.25 * Encapsulation – 0.25 * Coupling +0.5 * Composition + 0.5 * Polymorphism
ความสามารถในการทำงาน	0.12*Cohesion+0.22*Polymorphism+0.22*Messaging+0.22*Design Size+0.22*Hierarchies
ความสามารถในการขยาย	0.5 * Abstraction – 0.5 * Coupling + 0.5 * Inheritance + 0.5 Polymorphism
ความสามารถในการมีประสิทธิภาพ	0.2*Abstraction+0.2*Encapsulation+0.2*Composition+0.2*Inheritance+0.2*Polymorphism

3.2.4 การดำเนินการทดลองและเก็บผลข้อมูล

ขั้นตอนของการดำเนินการทดลองและเก็บผลข้อมูล แบ่งออกเป็น 2 ขั้นตอน ได้แก่

1) การคำนวณด้วยมาตรวัดเชิงวัตถุ คือ เป็นการคำนวณหาค่าคุณสมบัติการออกแบบด้วยมาตรเชิงวัตถุ 11 มาตรวัด และเก็บรวบรวมผลข้อมูลที่ได้จากการคำนวณ เพื่อนำเอาไปแทนค่าในสมการการคำนวณคุณภาพของการ

ออกแบบตามตารางที่ 10 เพื่อใช้ในการคำนวณหาค่าคุณภาพการของออกแบบซอฟต์แวร์

2) การคำนวณหาค่าคุณภาพของการออกแบบ คือ การคำนวณหาค่าคุณภาพของการออกแบบ ได้แก่ ความสามารถในการนำกลับมาใช้ใหม่ ความสามารถในการยืดหยุ่น ความสามารถในการทำงาน ความสามารถในการขยาย และความสามารถในการมีประสิทธิภาพ และเก็บผลที่ได้จากการคุณภาพของการออกแบบตามตารางที่ 11 นำไปใช้ในสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

3.2.5 การสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

การสร้างโมเดลเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ในขั้นตอนนี้จะเป็นการนำเอาผลการคำนวณหาค่าคุณภาพของการออกแบบซอฟต์แวร์ มาวิเคราะห์ด้วยโปรแกรมเอสพีเอสเอสโดยเลือกใช้เทคนิคการวิเคราะห์การจำแนกกลุ่ม (Discriminant Analysis) เลือกใช้วิธีแบบลำดับขั้น (Stepwise Selection) เพื่อหาตัวแปรอิสระ (Independent variable) ที่เหมาะสมมาใช้ในการสร้างสมการการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ สำหรับการแบ่งกลุ่มของความสามารถในการบำรุงรักษาซอฟต์แวร์จะแบ่งออกเป็น 2 กลุ่ม คือ ระดับง่าย (Easy level) และระดับยาก (Difficult level)

3.3 การพัฒนาเครื่องมือ

ขั้นตอนการพัฒนาเครื่องมือจะแบ่งการพัฒนาตามฟังก์ชันของการทำงาน สามารถแบ่งออกได้เป็นฟังก์ชันการทำงานหลัก และฟังก์ชันการทำงานเสริมความสามารถให้แก่เครื่องมือ โดยมีรายละเอียดดังนี้

1) ฟังก์ชันการทำงานหลัก ได้แก่ ฟังก์ชันการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด ฟังก์ชันการคำนวณหาค่าคุณภาพของการออกแบบ และฟังก์ชันการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์
2) ฟังก์ชันการทำงานเพื่อเสริมความสามารถให้แก่เครื่องมือ ได้แก่ ฟังก์ชันการนำข้อมูลออก ฟังก์ชันการเปลี่ยนภาษา และฟังก์ชันการดาวน์โหลดคู่มือการใช้งาน

3.4 การประเมินและสรุปผลการทดลอง

ในขั้นตอนนี้เป็นการประเมินผลการทดลองที่ได้ทั้งหมดตามขั้นตอนที่ได้ระบุไว้ตามขั้นตอนที่ 3.1 ขั้นตอนที่ 3.2 และขั้นตอนที่ 3.4 ทำการสรุปผลการทดลองที่ได้ทั้งหมดเพื่อนำไปปรับปรุงกระบวนการทำงาน และการพัฒนาเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ให้ดียิ่งขึ้น

บทที่ 4

การออกแบบและพัฒนาเครื่องมือ

ในบทนี้จะอธิบายรายละเอียดในการออกแบบและพัฒนาเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ โดยเครื่องมือจะพัฒนาในรูปแบบของเว็บแอพพลิเคชันด้วยเทคโนโลยีดอตเน็ตเฟรมเวิร์ก 4.0 (.NET Framework 4.0) โดยใช้ภาษาซีชาร์ป (C# Programming Language) และภาษาลิ้งค์ (LINQ : Language Integrated Query) ในการพัฒนาและอธิบายรายละเอียดเกี่ยวกับความต้องการด้านหน้าที่(Function Requirements) และไม่ใช่ในหน้าที่(Non Functional Requirements) ซึ่งมีรายละเอียดดังนี้

4.1 ความต้องการด้านหน้าที่ (Functional Requirements)

ในส่วนนี้จะเป็นการสรุปความต้องการด้านหน้าที่ของเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ซึ่งมีรายละเอียดดังต่อไปนี้

ตารางที่ 11 ความต้องการด้านหน้าที่

รหัส	ชื่อ	คำอธิบาย
F01	รองรับเอกสารเอ็กซ์เอ็มแอลจากการแปลงแผนภาพคลาส	เครื่องมือจะต้องรองรับเอกสารเอ็กซ์เอ็มแอลของแผนภาพคลาสที่ถูกแปลงอยู่ในรูปแบบเอกสารเอ็กซ์เอ็มแอลด้วยโปรแกรมสตาร์ยูเอ็มแอลเท่านั้น
F02	การคำนวณมาตรวัดเชิงวัตถุด้วยแผนภาพคลาส	1. เครื่องมือจะต้องสามารถอ่านข้อมูลตามโครงสร้างของเอกสารเอ็กซ์เอ็มแอลได้ เพื่อนำเอาข้อมูลที่ได้ ไปคำนวณด้วยมาตรวัดเชิงวัตถุ 2. ค่าที่ได้จากการคำนวณ จะต้องสามารถส่งผ่านเพื่อไปคำนวณหาค่า

รหัส	ชื่อ	คำอธิบาย
		คุณภาพของการออกแบบได้
F03	ส่วนการแสดงผลข้อมูลของการคำนวณมาตรวัดเชิงวัตถุ	เครื่องมือจะต้องแสดงผลข้อมูลที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุ 11 มาตรวัด ในรูปแบบตารางได้
F04	ส่วนการแสดงผลข้อมูลจากการคำนวณด้วยมาตรวัดเชิงวัตถุด้วยรูปแบบแผนภูมิแท่ง	1.เครื่องมือสามารถแสดงผลการคำนวณมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด ในรูปแบบแผนภูมิแท่ง 2.ค่าที่ได้จากการคำนวณด้วยมาตรวัดในรูปแบบแผนภูมิแท่งจะต้องสามารถจัดเรียงลำดับข้อมูล ด้วยการเลือกลำดับการจัดเรียงได้อย่างถูกต้อง
F05	การคำนวณหาค่าคุณภาพของการออกแบบ	เครื่องมือจะต้องสามารถรับค่าที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัดมาใช้ในการคำนวณค่าคุณภาพของการออกแบบ
F06	ส่วนการแสดงผลของการคำนวณหาค่าคุณภาพของการออกแบบ	เครื่องมือจะต้องแสดงผลที่ได้จากการคำนวณหาค่าคุณภาพของการออกแบบในรูปแบบตารางได้
F07	การคำนวณด้วยสมการการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์	1.เครื่องมือจะต้องคำนวณหาค่าความสามารถในการบำรุงรักษาซอฟต์แวร์ ตามสมการจำแนกกลุ่มทั้ง 2 วิธี ได้แก่ สมการของ Fisher's Liner Discriminate functions และสมการ Canonical Discriminate Function Coefficientsที่ได้มาจากขั้นตอนการสร้างโมเดลด้วยโปรแกรม

รหัส	ชื่อ	คำอธิบาย
		เอสพีเอสเอส
F08	ส่วนการแสดงผลข้อมูลที่ได้จากการคำนวณด้วยสมการจำแนกกลุ่มทั้ง 2 วิธี	เครื่องมือจะต้องแสดงผลข้อมูลที่ได้จากการคำนวณด้วยสมการการจำแนกกลุ่มทั้ง 2 วิธีในรูปแบบของตารางได้
F09	เครื่องมือจะต้องสามารถบอกความสามารถในการบำรุงรักษาซอฟต์แวร์ได้	เครื่องมือจะต้องสามารถบอกความสามารถในการบำรุงรักษาซอฟต์แวร์ ได้แก่ ระดับง่าย (Easy) และระดับยาก (Difficult) ได้
F10	ส่วนของการนำข้อมูลออก (Export data)	เครื่องมือจะต้องรองรับการนำข้อมูลออกในรูปแบบเอกสาร ดังต่อไปนี้ 1.รูปแบบเอกสารเวิร์ด (Word) นามสกุลไฟล์เอกสาร .doc เท่านั้น 2.รูปแบบไฟล์เอกสารเอ็กเซล (Excel) นามสกุลไฟล์เอกสาร .xls เท่านั้น
F011	การตั้งชื่อไฟล์เอกสารของข้อมูลที่ถูกนำออก (Export data)	เครื่องมือมีรูปแบบการตั้งชื่อไฟล์เอกสาร ดังนี้ “ชื่อไฟล์เอกสาร_วันเดือนปี (ddMMyyy)” แล้วเท่านั้น โดยมีคำอธิบาย 1.ชื่อไฟล์เอกสาร คือ ชื่อของไฟล์แผนภาพคลาสที่ถูกแปลงเป็นเอกสารเอ็กซ์เอ็มแอล 2. วันเดือนปี คือ เป็นวันที่ในนำข้อมูลออก (Export data) โดยมีรูปแบบของวันเดือนปี ddMMyyyyเท่านั้น

4.2 ความต้องการที่ไม่ใช่หน้าที่ (Non Functional Requirements)

ในส่วนนี้จะแสดงรายละเอียดของความต้องการที่ไม่ใช่หน้าที่ของเครื่องมือที่ทำการพัฒนาโดยมีรายละเอียดตามตารางที่ 12

ตารางที่ 12 ความต้องการที่ไม่ใช่หน้าที่

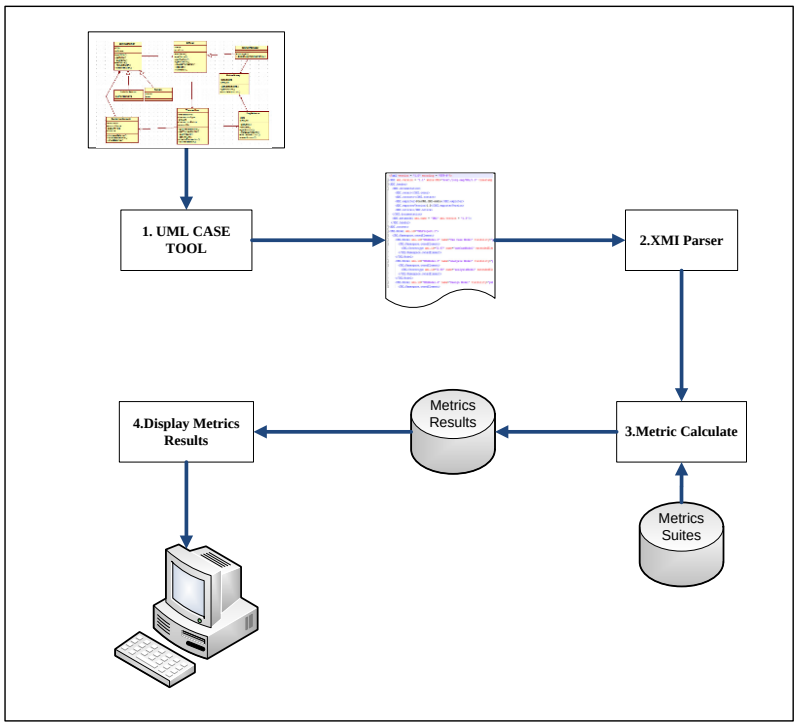
รหัส	ชื่อ	คำอธิบาย
F012	เครื่องมือที่พัฒนาจะต้องอยู่ในรูปแบบเว็บแอพพลิเคชัน	เครื่องมือสามารถรองรับการทำงานร่วมกับเว็บเบราว์เซอร์ คือ อินเทอร์เน็ตเอกซ์พลอเรอร์ 9 ขึ้นไป
F013	เครื่องมือที่พัฒนาจะต้องมีคู่มือการใช้งาน	เครื่องมือจะต้องมีคู่มือการใช้งาน โดยรองรับภาษาไทย และภาษาอังกฤษ
F014	เครื่องมือจะต้องสามารถแสดงผลได้ทั้งภาษา และภาษาอังกฤษ	เครื่องมือจะต้องแสดงผลได้ทั้งภาษาไทยและภาษาอังกฤษ

4.3 การออกแบบและพัฒนาเครื่องมือ

ในการพัฒนาเครื่องมือเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบ โดยจะทำการพัฒนาในรูปแบบของเว็บแอพพลิเคชัน (Web Application) โดยรายละเอียดในส่วนนี้จะนำเสนอแผนภาพยูสเคส แผนภาพคลาส และสภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ ซึ่งมีรายละเอียดดังต่อไปนี้

4.3.1 ภาพรวมของการพัฒนาเครื่องมือ

จากการวิเคราะห์ความต้องการด้านหน้าที่ และความต้องการที่ไม่ใช่หน้าที่ของการพัฒนาเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในหัวข้อที่ 4.1 และ 4.2 สามารถสรุปภาพรวมของการพัฒนาเครื่องมือ และกระบวนการในการทำงานของเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ดังรูปที่ 4



รูปที่ 4 ภาพรวมของการพัฒนาเครื่องมือ

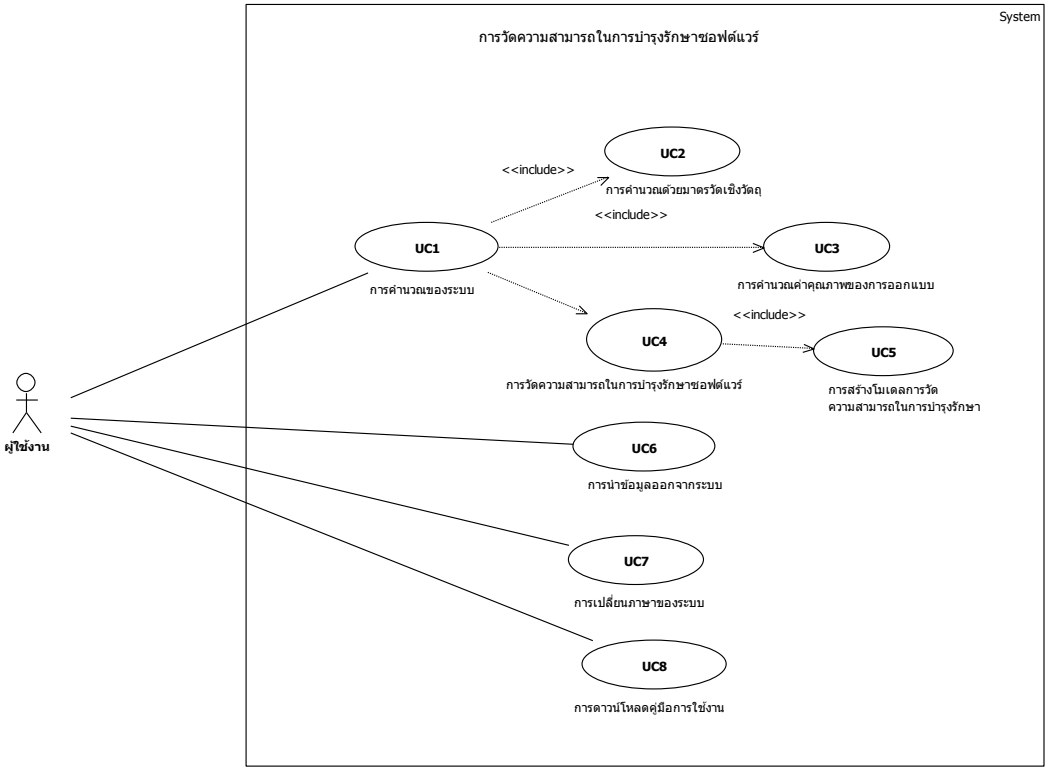
1) การสร้างแผนภาพคลาสและการแปลงแผนภาพคลาสที่ให้อยู่ในรูปเอกสารเอ็กซ์เอ็มแอล โดยใช้โปรแกรมสตาร์ยูเอ็มแอลเท่านั้น และนำเอกสารเอ็กซ์เอ็มแอลเข้าสู่ระบบ เพื่อทำการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์
2) กระบวนการในการอ่านค่าคุณลักษณะต่างๆภายในเอกสารเอ็กซ์เอ็มแอล และส่งผ่านไปยังฟังก์ชันการทำงานหลัก เพื่อใช้ในการคำนวณค่าการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนต่อไป
3) ฟังก์ชันการคำนวณด้วยมาตรวัดเชิงวัตถุจะเป็นกระบวนการทำงานแรกที่รับค่าคุณลักษณะต่างๆของเอกสารเอ็กซ์เอ็มแอลมาทำการคำนวณด้วยมาตรวัดเชิงวัตถุตามแต่ละมาตรวัด และคำนวณหาค่าคุณภาพของการออกแบบตามฟังก์ชันการคำนวณหาคุณภาพของการออกแบบ โดยผลการคำนวณจะใช้ในการคำนวณด้วยฟังก์ชันการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ซึ่งเป็นกระบวนการทำงานสุดท้ายของฟังก์ชันการทำงานหลักของเครื่องมือ

4) ค่าที่ได้จากการคำนวณด้วยฟังก์ชันการทำงานหลักของระบบทั้ง 3 ฟังก์ชันการทำงาน จะแสดงผลที่หน้าจอคอมพิวเตอร์ ซึ่งมีรายละเอียดดังนี้

- แสดงค่าที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุ 11 มาตรวัด
- แสดงค่าที่ได้จากการคำนวณหาค่าคุณภาพของการออกแบบซอฟต์แวร์
- แสดงค่าที่ได้จากการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

4.3.2 แผนภาพยูสเคสของการพัฒนาเครื่องมือ

จากหัวข้อที่ 4.3.1 ได้กล่าวถึงภาพรวมของการพัฒนาเครื่องมือและกระบวนการในการทำงานของเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ สามารถวิเคราะห์กระบวนการในการทำงานของเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ โดยแบ่งกลุ่มตามฟังก์ชันการทำงานของระบบออกเป็น 2 กลุ่ม ได้แก่ ฟังก์ชันการทำงานหลัก และฟังก์ชันการทำงานเสริมความสามารถให้แก่เครื่องมือ



รูปที่ 5 แผนภาพยูสเคสของการพัฒนาเครื่องมือ

รูปที่ 5 แสดงแผนภาพยูสเคสของเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

โดยมีกลุ่มผู้ใช้งานเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ได้แก่ ผู้พัฒนาระบบ ผู้ออกแบบระบบ และผู้ที่มีหน้าที่เกี่ยวกับการพัฒนาซอฟต์แวร์ หรือผู้ที่มีความสนใจเกี่ยวกับการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ เป็นต้น

4.3.2.1 **ฟังก์ชันการทำงานหลัก**

1) ฟังก์ชันการคำนวณด้วยมาตรวัดเชิงวัตถุ เป็นการคำนวณค่ามาตรวัดเชิงวัตถุทั้ง 11 มาตรวัดด้วยแผนภาพคลาส และผลการคำนวณจะไปใช้ในการคำนวณหาค่าคุณภาพของการออกแบบซอฟต์แวร์

2) ฟังก์ชันการคำนวณหาคุณภาพของการออกแบบ เป็นการคำนวณหาค่าคุณภาพของการออกแบบซอฟต์แวร์ทั้ง 5 ค่า ได้แก่ ความสามารถในการนำกลับมาใช้ใหม่ (Reusability) ความสามารถในการยืดหยุ่น (Flexibility) ความสามารถในการทำงาน (Functionality) ความสามารถในการขยาย (Extendibility) และความสามารถในการมีประสิทธิภาพ (Effectiveness) โดยผลการคำนวณจะนำมาใช้ในการสร้างโมเดลการวัดความสามารถในการบำรุงรักษารักษาซอฟต์แวร์
3) ฟังก์ชันการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ในฟังก์ชันการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์นี้ จะรวมขั้นตอนในการสร้างโมเดลเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ โดยจะใช้ผลการคำนวณหาค่าคุณภาพการออกแบบของระบบทั้ง 10 ระบบมาใช้ในการสร้างโมเดล และใช้โปรแกรมเอสพีเอสเอสในการวิเคราะห์ข้อมูลด้วยวิธีการจำแนกกลุ่ม โดยจำแนกกลุ่มของความสามารถในการบำรุงรักษาซอฟต์แวร์ออกเป็น 2 กลุ่ม ได้แก่ ระดับง่าย และระดับยาก ผลที่ได้จากการวิเคราะห์ด้วยโปรแกรมเอสพีเอสเอส คือ สมการจำแนกกลุ่มการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ และนำเอาสมการจำแนกกลุ่มการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ไปใช้สร้างเป็นฟังก์ชันการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ให้แก่เครื่องมือ

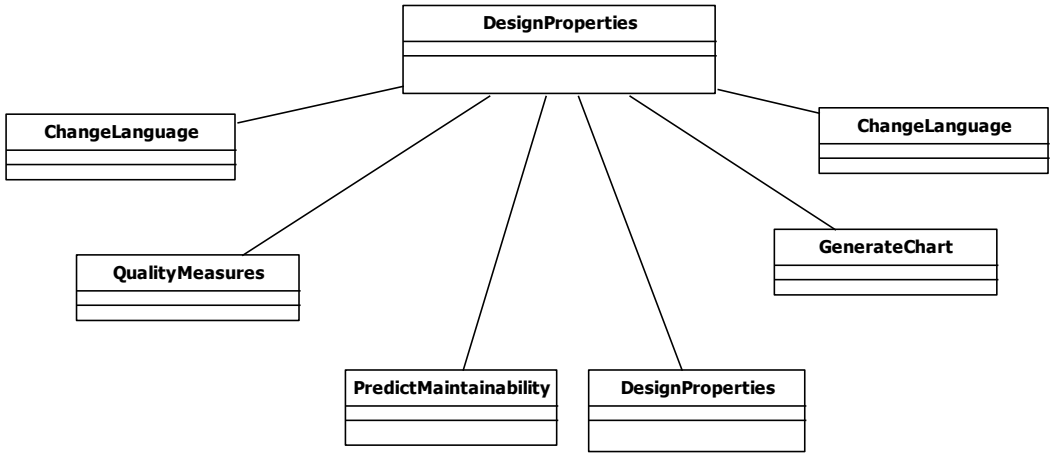
4.3.2.2 **ฟังก์ชันการทำงานเสริมความสามารถให้แก่เครื่องมือ**

1) ฟังก์ชันการนำข้อมูลออก คือ ฟังก์ชันการทำงานเสริมความสามารถให้แก่เครื่องมือ โดยเครื่องมือจะต้องสามารถรองรับการทำงานในการนำเอาข้อมูลออกในรูปแบบของเอกสารเวิร์ด(Word) และเอกสารเอ็กเซล(Excel)
2) ฟังก์ชันการเปลี่ยนภาษา คือ เครื่องมือจะต้องสามารถรองรับการแสดงผลภาษาไทย และภาษาอังกฤษ ด้วยการเลือกเปลี่ยนภาษา
3) ฟังก์ชันการดาวน์โหลดคู่มือการใช้งาน คือ เครื่องมือได้มีการเตรียมคู่มือการใช้งานในรูปแบบเอกสารพีดีเอฟ (PDF) ซึ่งรองรับภาษาไทยและภาษาอังกฤษ โดยให้ผู้ใช้งานสามารถเลือกดาวน์โหลดเอกสารนี้ได้จากเมนู “ช่วยเหลือ(Helps)”

รายละเอียดเกี่ยวกับคำอธิบายของแผนภาพยูเคสจะอยู่ในภาคผนวก ก

4.3.3 แผนภาพคลาส

ในส่วนนี้แสดงแผนภาพคลาสทั้งหมดที่ใช้ในการในการพัฒนาเครื่องมือ มีทั้งหมด 7 คลาสหลัก และ 11 คลาสย่อยสำหรับใช้ในการคำนวณด้วยมาตรวัดเชิงวัตถุ ดังรูปที่ 6



รูปที่ 6 แผนภาพคลาสของการพัฒนาเครื่องมือ

- **คลาส** Default

คลาส Default เป็นคลาสหลักในใช้ในการแสดงผลข้อมูล และเป็นส่วนต่อประสานกับผู้ใช้งาน นั้นคือ เป็นส่วนแสดงผลข้อมูล และเป็นส่วนนำข้อมูลเข้าและออก โดยข้อมูลนำเข้าสู่ระบบ คือ แผนภาพคลาสที่แปลงอยู่ในรูปแบบเอกสารเอ็กซ์เอ็มแอล ส่วนการแสดงผลของข้อมูลแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด แสดงผลการคำนวณหาค่าคุณภาพของการออกแบบ และแสดงผลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ในส่วนของการนำข้อมูลออกจากระบบรองรับการนำข้อมูลออกในรูปแบบเอกสารเวิร์ด (Word) และเอกสารเอ็กเซล (Excel) เท่านั้น

- **คลาส** DesignProperties

คลาส DesignProperties เป็นคลาสกลางสำหรับเรียกใช้คลาสย่อยอีก 11 คลาสเพื่อใช้ในการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด คลาสย่อย 11 คลาส มีรายละเอียดดังนี้

- **คลาส** DesignSizeMeasure

คลาส DesignSizeMeasure เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Design Size in Class (DSC) ด้วยแผนภาพคลาส

- **คลาส** HieratchiesMeasure

คลาส HieratchiesMeasure เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Number of Hierarchies (NOH) ด้วยแผนภาพคลาส

- **คลาส** AbstractionMeasure

คลาส AbstractionMeasure เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Average Number of Ancestors (ANA) ด้วยแผนภาพคลาส

- **คลาส** EncapsulationMeasure

คลาส EncapsulationMeasur เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Data Access Metric (DAM) ด้วยแผนภาพคลาส

- **คลาส** CouplingMeasure

คลาส CouplingMeasure เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Direct Class Coupling (DCC) ด้วยแผนภาพคลาส

- **คลาส** CohesionMeasure

คลาส CohesionMeasure เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Cohesion Among Methods of Class (CAM) ด้วยแผนภาพคลาส

- **คลาส CompositionMeasure**

คลาส CompositionMeasure เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Measure of Functional Abstraction (MFA) ด้วยแผนภาพคลาส

- **คลาส InheritanceMeasure**

คลาส InheritanceMeasure เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Measure of Functional Abstraction (MFA) ด้วยแผนภาพคลาส

- **คลาส PolymorphismMeasure**

คลาส PolymorphismMeasure เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Number of Polymorphic Methods (NOP) ด้วยแผนภาพคลาส

- **คลาส MessagingMeasure**

คลาส MessagingMeasure เป็นคลาสที่ใช้ในการคำนวณหาค่ามาตรวัด Class Interface Size (CIS) ด้วยแผนภาพคลาส

- **คลาส ComplexityMeasure**

คลาส ComplexityMeasure เป็นคลาสที่ใช้ในการคำนวณหามาตรวัด Number of Methods (NOM) ด้วยแผนภาพคลาส

- **คลาส QualityMeasures**

คลาส QualityMeasures เป็นคลาสที่ใช้ในการคำนวณหาค่าคุณภาพของการออกแบบทั้ง 5 ค่า ได้แก่ ความสามารถในการนำกลับมาใช้ใหม่ (Reusability) ความสามารถในการยืดหยุ่น(Flexibility) ความสามารถในการทำงาน (Functionality) ความสามารถในการขยาย(Extendibility) และความสามารถในการมีประสิทธิภาพ(Effectiveness) โดยรับข้อมูลที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัดเข้ามาแทนที่ในสมการเพื่อคำนวณหาคุณภาพของการออกแบบทั้ง 5 ค่า

- **คลาส** PredictMaintainability

คลาส PredictMaintainability เป็นคลาสที่ใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ โดยรับข้อมูลที่ได้จากการคำนวณหาค่าคุณภาพของการออกแบบมาเข้าสู่สมการการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

- **คลาส** GenerateChart

คลาส GenerateChart เป็นคลาสที่ใช้ในการแสดงผลในรูปแบบแผนภูมิแท่งที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด

- **คลาส** ExportData

คลาส ExportData เป็นคลาสที่ใช้ในการนำข้อมูลออก โดยข้อมูลจะอยู่ในรูปแบบเอกสารเวิร์ด (Word) และเอกสารเอ็กเซล (Excel) โดยจะมีนามสกุลของไฟล์เอกสารเป็น .doc และ .xls ตามลำดับ

- **คลาส** ChangeLanguage

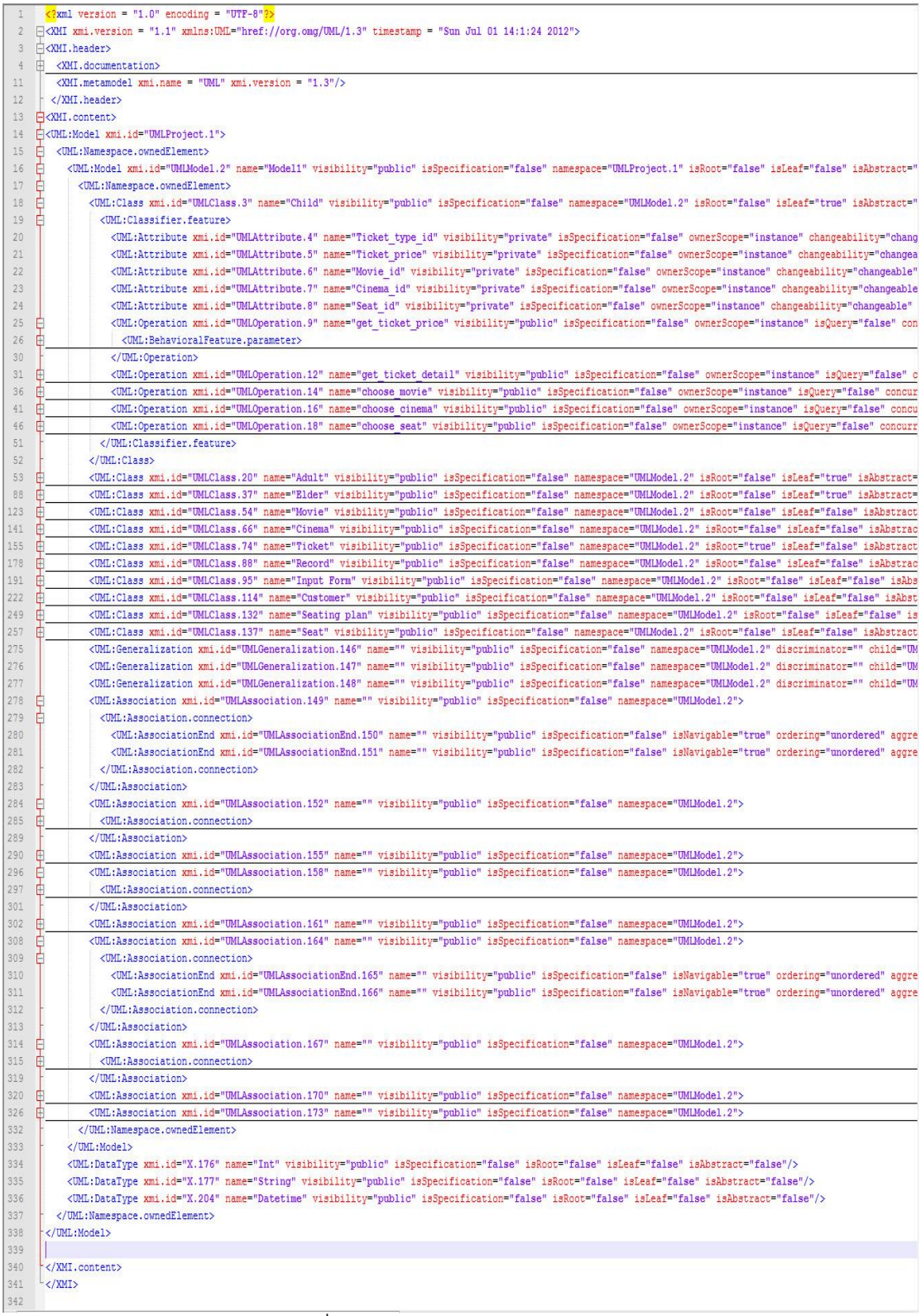
คลาส ChangeLanguage เป็นคลาสสำหรับเปลี่ยนภาษาที่ใช้ในการแสดงผลข้อมูล โดยรองรับกับการแสดงผลเป็นภาษาไทยและภาษาอังกฤษเท่านั้น

รายละเอียดเกี่ยวกับคำอธิบายจะอยู่ในภาคผนวก ข

4.4 การอ่านข้อมูลของเอกสารเอ็กซ์เอ็มแอล

แผนภาพคลาสที่ใช้ในการคำนวณจะต้องอยู่ในรูปเอกสารเอ็กซ์เอ็มแอล สำหรับงานวิจัยนี้ใช้โปรแกรมสตาร์ยูเอ็มแอล (StarUML) มาใช้ในการแปลงแผนภาพคลาส

โครงสร้างของเอกสารเอ็กซ์เอ็มแอล มีส่วนประกอบหลัก 2 ส่วน ได้แก่ ส่วนหัว (Head) และส่วนของเนื้อหา (Content) ดังแสดงในรูปที่ 7



```
<?xml version = "1.0" encoding = "UTF-8"?>
<XMI xmi.version = "1.1" xmlns:UML="href://org.omg/UML/1.3" timestamp = "Sun Jul 01 14:1:24 2012">
<XMI.header>
  <XMI.documentation>
  <XMI.metamodel xmi.name = "UML" xmi.version = "1.3"/>
 </XMI.header>
<XMI.content>
<UML:Model xmi.id="UMLProject.1">
  <UML:Namespace.ownedElement>
    <UML:Model xmi.id="UMLModel.2" name="Model1" visibility="public" isSpecification="false" namespace="UMLProject.1" isRoot="false" isLeaf="false" isAbstract="
      <UML:Namespace.ownedElement>
        <UML:Class xmi.id="UMLClass.3" name="Child" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="true" isAbstract="
          <UML:Classifier.feature>
            <UML:Attribute xmi.id="UMLAttribute.4" name="Ticket_type_id" visibility="private" isSpecification="false" ownerScope="instance" changeability="chang
            <UML:Attribute xmi.id="UMLAttribute.5" name="Ticket_price" visibility="private" isSpecification="false" ownerScope="instance" changeability="changea
            <UML:Attribute xmi.id="UMLAttribute.6" name="Movie_id" visibility="private" isSpecification="false" ownerScope="instance" changeability="changeable"
            <UML:Attribute xmi.id="UMLAttribute.7" name="Cinema_id" visibility="private" isSpecification="false" ownerScope="instance" changeability="changeable
            <UML:Attribute xmi.id="UMLAttribute.8" name="Seat_id" visibility="private" isSpecification="false" ownerScope="instance" changeability="changeable"
            <UML:Operation xmi.id="UMLOperation.9" name="get_ticket_price" visibility="public" isSpecification="false" ownerScope="instance" isQuery="false" con
              <UML:BehavioralFeature.parameter>
            </UML:Operation>
            <UML:Operation xmi.id="UMLOperation.12" name="get_ticket_detail" visibility="public" isSpecification="false" ownerScope="instance" isQuery="false" c
            <UML:Operation xmi.id="UMLOperation.14" name="choose_movie" visibility="public" isSpecification="false" ownerScope="instance" isQuery="false" concur
            <UML:Operation xmi.id="UMLOperation.16" name="choose_cinema" visibility="public" isSpecification="false" ownerScope="instance" isQuery="false" concu
            <UML:Operation xmi.id="UMLOperation.18" name="choose_seat" visibility="public" isSpecification="false" ownerScope="instance" isQuery="false" concurr
          </UML:Classifier.feature>
        </UML:Class>
        <UML:Class xmi.id="UMLClass.20" name="Adult" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="true" isAbstract=
        <UML:Class xmi.id="UMLClass.37" name="Elder" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="true" isAbstract=
        <UML:Class xmi.id="UMLClass.54" name="Movie" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="false" isAbstract
        <UML:Class xmi.id="UMLClass.66" name="Cinema" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="false" isAbstrac
        <UML:Class xmi.id="UMLClass.74" name="Ticket" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="true" isLeaf="false" isAbstract
        <UML:Class xmi.id="UMLClass.88" name="Record" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="false" isAbstrac
        <UML:Class xmi.id="UMLClass.95" name="Input Form" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="false" isAbs
        <UML:Class xmi.id="UMLClass.114" name="Customer" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="false" isAbst
        <UML:Class xmi.id="UMLClass.132" name="Seating plan" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="false" is
        <UML:Class xmi.id="UMLClass.137" name="Seat" visibility="public" isSpecification="false" namespace="UMLModel.2" isRoot="false" isLeaf="false" isAbstract
        <UML:Generalization xmi.id="UMLGeneralization.146" name="" visibility="public" isSpecification="false" namespace="UMLModel.2" discriminator="" child="UM
        <UML:Generalization xmi.id="UMLGeneralization.147" name="" visibility="public" isSpecification="false" namespace="UMLModel.2" discriminator="" child="UM
        <UML:Generalization xmi.id="UMLGeneralization.148" name="" visibility="public" isSpecification="false" namespace="UMLModel.2" discriminator="" child="UM
        <UML:Association xmi.id="UMLAssociation.149" name="" visibility="public" isSpecification="false" namespace="UMLModel.2">
          <UML:Association.connection>
            <UML:AssociationEnd xmi.id="UMLAssociationEnd.150" name="" visibility="public" isSpecification="false" isNavigable="true" ordering="unordered" aggre
            <UML:AssociationEnd xmi.id="UMLAssociationEnd.151" name="" visibility="public" isSpecification="false" isNavigable="true" ordering="unordered" aggre
          </UML:Association.connection>
        </UML:Association>
        <UML:Association xmi.id="UMLAssociation.152" name="" visibility="public" isSpecification="false" namespace="UMLModel.2">
          <UML:Association.connection>
        </UML:Association>
        <UML:Association xmi.id="UMLAssociation.155" name="" visibility="public" isSpecification="false" namespace="UMLModel.2">
        <UML:Association xmi.id="UMLAssociation.158" name="" visibility="public" isSpecification="false" namespace="UMLModel.2">
          <UML:Association.connection>
        </UML:Association>
        <UML:Association xmi.id="UMLAssociation.161" name="" visibility="public" isSpecification="false" namespace="UMLModel.2">
        <UML:Association xmi.id="UMLAssociation.164" name="" visibility="public" isSpecification="false" namespace="UMLModel.2">
          <UML:Association.connection>
            <UML:AssociationEnd xmi.id="UMLAssociationEnd.165" name="" visibility="public" isSpecification="false" isNavigable="true" ordering="unordered" aggre
            <UML:AssociationEnd xmi.id="UMLAssociationEnd.166" name="" visibility="public" isSpecification="false" isNavigable="true" ordering="unordered" aggre
          </UML:Association.connection>
        </UML:Association>
        <UML:Association xmi.id="UMLAssociation.167" name="" visibility="public" isSpecification="false" namespace="UMLModel.2">
          <UML:Association.connection>
        </UML:Association>
        <UML:Association xmi.id="UMLAssociation.170" name="" visibility="public" isSpecification="false" namespace="UMLModel.2">
        <UML:Association xmi.id="UMLAssociation.173" name="" visibility="public" isSpecification="false" namespace="UMLModel.2">
      </UML:Namespace.ownedElement>
    </UML:Model>
    <UML:DataType xmi.id="X.176" name="Int" visibility="public" isSpecification="false" isRoot="false" isLeaf="false" isAbstract="false"/>
    <UML:DataType xmi.id="X.177" name="String" visibility="public" isSpecification="false" isRoot="false" isLeaf="false" isAbstract="false"/>
    <UML:DataType xmi.id="X.204" name="Datetime" visibility="public" isSpecification="false" isRoot="false" isLeaf="false" isAbstract="false"/>
  </UML:Namespace.ownedElement>
</UML:Model>

</XMI.content>
</XMI>
```

รูปที่ 7 รูปแบบเอกสารเอ็กซ์เอ็มแอล

ในการคำนวณด้วยมาตรวัดเชิงวัตถุจะพิจารณาข้อมูลของเอกสารเอ็กซ์เอ็มแอลในส่วนของเนื้อหา (Content) เท่านั้น ซึ่งมีรายละเอียดของแท็กอิลิเมนต์ที่ใช้ในการพิจารณาดังนี้

1) คลาส ซึ่งแสดงด้วยแท็กอิลิเมนต์ UML: Class ประกอบด้วยคุณลักษณะที่สำคัญ ได้แก่ xmi.id ใช้ในการแสดงลำดับอ้างอิงของคลาส เพื่อใช้ในการอ้างถึงภายในโครงสร้างของเอกสารเอ็กซ์เอ็มแอล คุณลักษณะname แสดงชื่อของคลาส คุณลักษณะ visibility แสดงระดับความสามารถในการเข้าถึงของคลาส คือ พับลิค โพรเทค และไพรเวท คุณลักษณะ isRoot คือ ใช้ในการแสดงว่าคลาสนี้เป็นรูท (Root) หรือไม่ คุณลักษณะ isLeaf ใช้ในการแสดงว่าคลาสนี้เป็นลีฟ (Leaf) หรือไม่ คุณลักษณะ isAbstract ใช้ในการแสดงว่าคลาสนี้เป็น Abstract หรือไม่
2) คุณลักษณะภายในคลาส แสดงด้วยแท็กอิลิเมนต์ UML: Attribute ประกอบด้วยคุณลักษณะที่สำคัญ ได้แก่ xmi.id ใช้ในการแสดงลำดับอ้างอิงของคุณลักษณะ คุณลักษณะname คือ ชื่อของคุณลักษณะภายในคลาส คุณลักษณะ visibilityแสดงระดับความสามารถในการเข้าถึงของคลาสของคลาส คือ พับลิค โพรเทค และไพรเวท คุณลักษณะ type คือ บอกชนิดของคุณลักษณะภายในคลาส ได้แก่ Integer String Float และDouble เป็นต้น และคุณลักษณะ owner คือ คุณลักษณะนี้อยู่ภายใต้คลาสอะไร
3) โอเปอร์เรชัน หรือเมธอด แสดงด้วยแท็กอิลิเมนต์ UML:Operation ประกอบด้วยคุณลักษณะที่สำคัญ ได้แก่ คุณลักษณะ xmi.id ใช้ในการแสดงลำดับอ้างอิงของโอเปอร์เรชัน คุณลักษณะ name คือ ชื่อของโอเปอร์เรชัน คุณลักษณะ visibility แสดงระดับความสามารถในการเข้าถึงของคลาส คือ พับลิค โพรเทค และไพรเวท คุณลักษณะ isAbstract ใช้ในการแสดงว่าโอเปอร์เรชันนี้เป็น Abstract หรือไม่ และคุณลักษณะ owner คือ โอเปอร์เรชันนี้อยู่ภายในคลาสอะไรและภายในโอเปอร์เรชั่นจะมีแท็กอิลิเมนต์ย่อยภายในคือ UML:Parameter เพื่อแสดงให้เห็นว่าภายในโอเปอร์เรชันรับค่าพารามิเตอร์อะไรบ้าง โดยมีคุณลักษณะที่สำคัญ ได้แก่ xmi.id แสดงลำดับของพารามิเตอร์ของเอกสารเอ็กซ์เอ็มแอล คุณลักษณะ name คือ ชื่อของพารามิเตอร์ คุณลักษณะ visibility แสดงระดับความสามารถในการ

เข้าถึงของคลาสคือ พับลิค โพรเทค และไพรเวท คุณลักษณะ behavioralFeature คือ พารามิเตอร์นี้เป็นของโอเปอร์เรชันอะไร และคุณลักษณะ type แสดงชนิดข้อมูลของพามิเตอร์ ได้แก่ Integer String Float และDouble เป็นต้น

ความสัมพันธ์ ในเอกสารเอ็กซ์เอ็มแอลมีความสัมพันธ์ 2 แบบ ได้แก่

1) ความสัมพันธ์แบบแอสโซซิเอชัน (Association) ซึ่งแสดงด้วยแท็กอิลิเมนต์ UML:Association ประกอบด้วยคุณลักษณะที่สำคัญ ได้แก่ คุณลักษณะ xmi.id ใช้ในการแสดงลำดับอ้างอิงของความสัมพันธ์แบบ Association คุณลักษณะ name คือ ชื่อของความสัมพันธ์แบบแอสโซซิเอชัน คุณลักษณะ visibility แสดงระดับความสามารถในการเข้าถึงของคลาส คือ พับลิค โพรเทค และไพรเวท และภายในความสัมพันธ์แบบแอสโซซิเอชันจะมีแท็กอิลิเมนต์ย่อยภายใน คือ UML:AssociationEnd ถ้ามีความสัมพันธ์แบบแอกกริเกชันคุณลักษณะแอกกริเกชัน (Aggregation) จะเปลี่ยนจากค่า “none” เป็น “aggregate” แต่ถ้ามีความสัมพันธ์แบบคอมโพซิชัน (Composition) คุณลักษณะ aggregation จะเปลี่ยนเป็นค่า “composite”
2) ความสัมพันธ์แบบเจเนอรัลไลเซชัน (Generalization) ซึ่งแสดงด้วยแท็กอิลิเมนต์ UML: Generalization ประกอบด้วยคุณลักษณะที่สำคัญ ได้แก่ xmi: id ใช้ในการแสดงลำดับอ้างอิงของความสัมพันธ์แบบเจเนอรัลไลเซชัน คุณลักษณะ name คือ ชื่อของความสัมพันธ์แบบเจเนอรัลไลเซชัน คุณลักษณะ visibility แสดงระดับความสามารถในการเข้าถึงของคลาส คือ พับลิค โพรเทค และไพรเวท คุณลักษณะไชล (child) คือ ใช้บ่งบอกความสัมพันธ์ของกับคลาสลูก และคุณลักษณะแพเรนท์ (parent) คือ ใช้บ่งบอกความสัมพันธ์กับคลาสแม่

4.5 สภาพแวดล้อมและเครื่องมือที่ใช้ในการพัฒนา

สภาพแวดล้อม และเครื่องมือที่ใช้ในการพัฒนาเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ มีรายละเอียดดังนี้

ฮาร์ดแวร์

1) หน่วยประมวลผลอินเทลเพนเทียมเอ็ม 2.40 กิกะเฮิร์ต (Core i5 M 2.40 GHz.)
2) หน่วยความจำ (RAM) 4000 เมกกะไบต์ (4GB)
3) ฮาร์ดดิสก์ (Hard disk) 500 กิกะไบต์ (500GB)

ซอฟต์แวร์

1) ระบบปฏิบัติการวินโดวส์ 7 (Windows 7 Ultimate)
2) เครื่องมือพัฒนาโปรแกรมไมโครซอฟต์วิชวลสตูดิโอ 2010 (Microsoft Visual Studio 2010)
3) ชุดพัฒนาดอตเน็ตเฟรมเวิร์ก 4.0 (.NET Framework 4.0)
4) ไอไอเอส 7.0 (IIS 7.0)
5) โปรแกรมสตาร์ยูเอ็มแอลเพื่อใช้การสร้างแผนภาพคลาส และแปลงแผนคลาสในรูปแบบเอกสารเอ็กซ์เอ็มแอล

4.6 การติดตั้งซอฟต์แวร์เพื่อใช้ในการพัฒนา

เมื่อเตรียมสภาพแวดล้อมและเครื่องมือสำหรับการพัฒนาระบบเรียบร้อยแล้ว จึงทำการติดตั้งเครื่องมือทั้งหมดลงในเครื่องคอมพิวเตอร์ที่ใช้พัฒนาเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ โดยมีลำดับการติดตั้งเป็นไปตามขั้นตอนต่อไปนี้

1) ติดตั้งระบบปฏิบัติการวินโดวส์ 7
2) ติดตั้งเครื่องมือพัฒนาโปรแกรมไมโครซอฟต์วิชวลสตูดิโอ 2010 (Visual Studio 2010)
3) ติดตั้งไอไอเอส(IIS) เป็นเว็บเซิร์ฟเวอร์
4) ติดตั้งโปรแกรมสตาร์ยูเอ็มแอล(StarUML) เพื่อใช้ในการออกแบบระบบด้วยแผนภาพคลาส และใช้ในการแปลงแผนภาพคลาสเป็นเอกสารเอ็กซ์เอ็มแอล

4.7 การพัฒนาส่วนต่อประสาน

ในการพัฒนาส่วนต่อประสานผู้ใช้งานสามารถดูรายละเอียดได้จากภาคผนวก ค

บทที่ 5

ผลการทดลอง

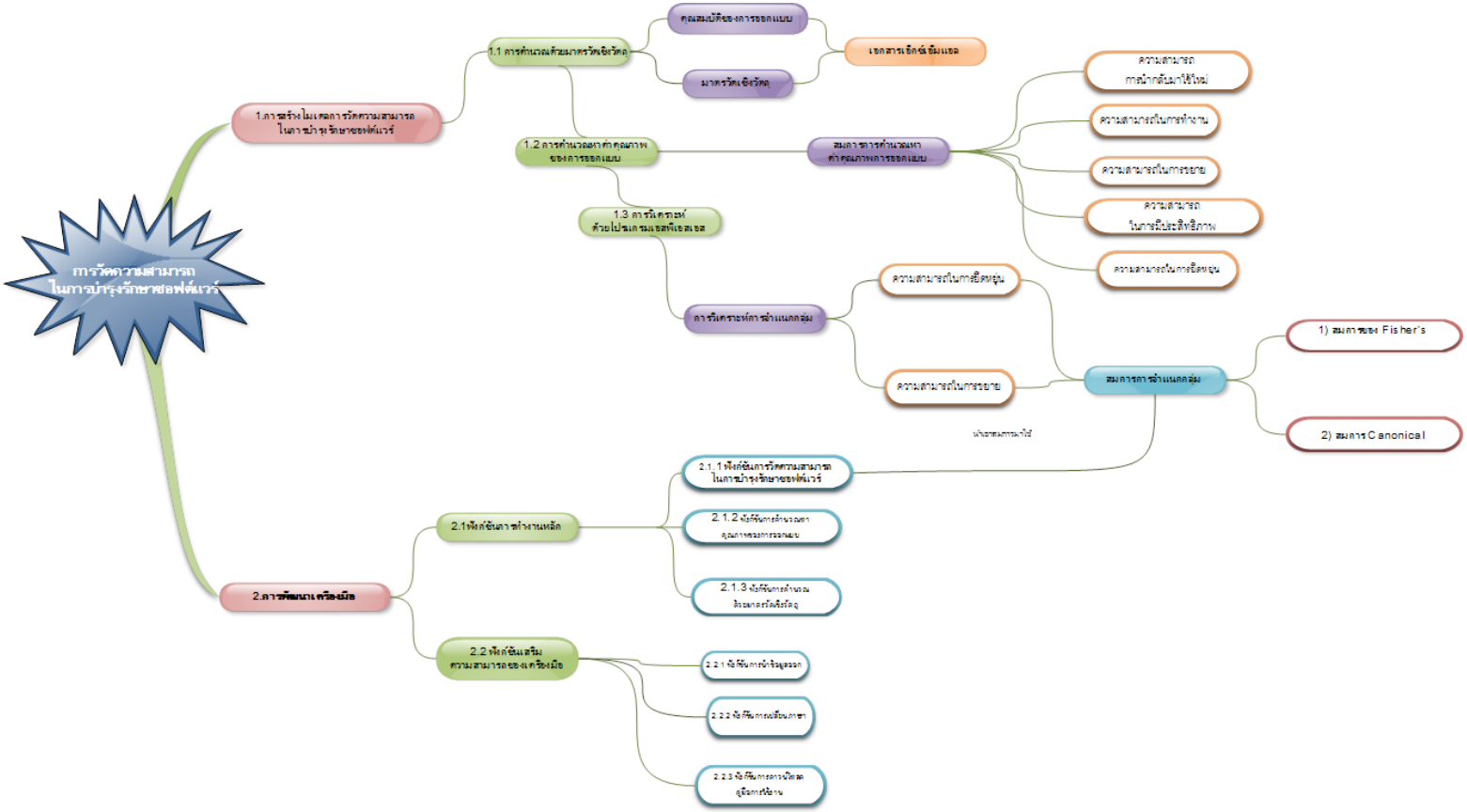
ในบทนี้จะกล่าวถึงผลที่ได้จากการทดลองตามที่กล่าวไว้ในบทที่ 3 ได้แก่ ผลของจากการคำนวณด้วยมาตรวัดเชิงวัตถุของระบบ 10 ระบบ และการคำนวณหาค่าคุณภาพของการออกแบบ เพื่อใช้ในการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ และผลการทดสอบการทำงานของเครื่องมือตรงตามกับฟังก์ชันการทำงานตามที่กำหนดในบทที่ 4 ในรูปที่ 8 แสดงภาพรวมของขั้นตอนทั้งหมดของงานวิจัยการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

5.1 ข้อมูลจากการคำนวณด้วยมาตรวัดทั้ง 11 มาตรวัด

ในส่วนนี้แสดงผลการคำนวณด้วยมาตรวัดทั้ง 11 มาตรวัดเพื่อใช้ในการคำนวณหาค่าคุณสมบัติการออกแบบของระบบทั้ง 10 ระบบที่ใช้ในงานวิจัยนี้เพื่อนำเอาผลการคำนวณที่ได้ไปคำนวณหาค่าคุณภาพของการออกแบบ โดยตารางที่ 13 แสดงข้อมูลที่ได้จากการคำนวณด้วยมาตรวัดทั้ง 11 มาตรวัดของระบบทั้ง 10 ระบบ

5.2 ข้อมูลจากการคำนวณหาค่าคุณภาพของการออกแบบ

ในส่วนนี้แสดงผลการคำนวณหาค่าคุณภาพของการออกแบบทั้ง 5 ค่า เพื่อใช้ในการสร้างโมเดลการวัดระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ โดยแสดงผลการคำนวณหาค่าคุณภาพของการออกแบบดังตารางที่ 14



รูปที่ 8 ภาพรวมของงานวิจัยการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

ตารางที่ 13 แสดงผลการคำนวณมาตรวัดเชิงวัตถุทั้ง 10 ระบบ

	ระบบ	การทำงานของระบบงาน	ความซับซ้อน	ความสามารถ	การนำกลับมาใช้	ปัจจัยด้านพฤติกรรม	การเชื่อมโยงกับระบบอื่น	ความสามารถในการขยาย	ความปลอดภัย	ความต้องการ	ความสามารถในการบำรุงรักษา
1	Airline Reservation System [15]	6.00	3.00	0.67	1.00	1.00	0.83	3.00	1.00	1.00	10.00	17.00
2	ATM System [18]	10.00	10.00	0.75	0.88	0.60	1.00	5.00	1.00	0.00	8.00	23.00
3	Banks System [1]	9.00	9.00	0.80	1.00	0.56	1.83	2.00	1.00	0.00	6.00	33.00
4	E-Ticket [16]	10.00	4.00	0.75	0.97	0.70	0.46	3.00	0.71	2.00	10.00	31.00
5	Hospital Management [19]	9.00	0.00	0.00	1.00	0.67	0.75	5.00	1.00	0.00	7.00	27.00
6	Internet Cafe [17]	20.00	4.00	0.75	0.94	1.21	0.4	1.00	1.00	0.00	4.00	114.00
7	Laboratory System [20]	12.00	3.00	0.67	0.92	0.50	1.13	1.00	0.91	2.00	2.00	25.00
8	Library Management [21]	6.00	2.00	0.50	1.00	1.17	1.42	4.00	0.77	3.00	10.00	13.00
9	Web Base Auction System [14]	10.00	7.00	0.86	1.00	0.30	1.14	2.00	0.82	1.00	5.00	11.00
10	Hotel Management System [13	8.00	0.00	0.00	0.97	1.35	2.07	1.00	1.00	0.00	2.00	18.00

ตารางที่ 14 แสดงผลการคำนวณหาค่าคุณภาพของการออกแบบทั้ง 10 ระบบ

	ระบบ	การใช้ซ้ำ (5)	ความยาว (5)	การนำกลับมาใช้ใหม่ (5)	ความเข้าใจ (5)	ความยืดหยุ่น(5)	ผลรวม (25)	ระดับ
1	Airline Reservation System [15]	2.00	0.84	4.96	3.40	1.33	12.53	Difficult
2	ATM System [18]	2.57	0.58	4.10	2.50	1.53	11.28	Difficult
3	Banks System [1]	1.11	0.62	3.32	2.98	0.96	8.99	Difficult
4	E-Ticket [16]	3.70	2.63	5.94	3.28	2.56	18.11	Easy
5	Hospital Management [19]	2.58	0.17	3.52	2.78	0.40	9.45	Difficult
6	Internet Cafe [17]	0.74	0.88	1.80	1.93	1.74	7.09	Difficult
7	Laboratory System [20]	1.60	1.54	1.16	1.79	1.10	7.19	Difficult
8	Library Management [21]	3.46	2.55	4.06	3.73	2.85	16.65	Easy
9	Web Base Auction System [14]	1.68	1.19	2.71	3.06	1.14	9.78	Difficult
10	Hotel Management System [13]	0.74	0.50	1.18	0.47	0.59	3.48	Difficult
					***	ค่า Mean		10.45
						ค่า SD		4.42

จากตารางที่ 14 สามารถจำแนกกลุ่มระดับความสามารถในการบำรุงรักษาซอฟต์แวร์เป็น 2 กลุ่ม ได้แก่ กลุ่มระดับง่าย (Easy Level) และกลุ่มระดับยาก (Difficult Level) โดยกำหนดเกณฑ์ในการแบ่งกลุ่มระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ ดังนี้

ถ้า ผลรวมของคะแนนกลุ่ม ≥ 14.87 อยู่ในกลุ่มระดับง่าย (Easy)

ผลรวมของคะแนนกลุ่ม < 14.87 อยู่ในกลุ่มระดับยาก (Difficult)

5.3 การสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

งานวิจัยนี้ได้สร้างโมเดลเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ เป็นไปตามขั้นตอนที่ได้อธิบายไว้ในบทที่ 3 ในการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ โดยการนำเอาข้อมูลที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุทั้งหมด 11 มาตรวัด มาคำนวณหาค่าคุณภาพของการออกแบบ โดยนำเอาผลการคำนวณที่ได้มาทำการวิเคราะห์การจำแนกกลุ่มด้วยโปรแกรมเอสพีเอสเอส โดยเลือกใช้วิธีแบบลำดับขั้นตอน(Stepwise Selection) มาใช้ในการเลือกตัวแปรที่เหมาะสมเพื่อใช้ในการสร้างสมการจำแนกกลุ่ม โดยผลที่ได้จากการวิเคราะห์ด้วยโปรแกรมเอสพีเอสเอส มีรายละเอียดดังนี้

5.3.1 การวิเคราะห์ข้อมูลที่ได้จากโปรแกรมเอสพีเอสเอส

ตารางที่ 15 Variables Entered/Removed แสดงค่าของตัวแปรอิสระที่มีความเหมาะสมนำไปใช้ในการสร้างเป็นสมการการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

ตารางที่ 15 Variables Entered/Removed

		Wilks' Lambda							
						Exact F			
Step	Entered	Statistic	df1	df2	df3	Statistic	df1	df2	Sig.
1	Extendibility	.198	1	1	8.000	32.456	1	8.000	.000
2	Flexibility	.123	2	1	8.000	25.059	2	7.000	.001

At each step, the variable that minimizes the overall Wilks' Lambda is entered.

a. Maximum number of steps is 10.

b. Minimum partial F to enter is 3.84.

c. Maximum partial F to remove is 2.71.

d. F level, tolerance, or VIN insufficient for further computation.

จากตารางที่ 15 สามารถสรุปได้ว่าตัวแปรอิสระ Extendibility ได้ถูกนำเข้าสู่สมการความถดถอยในลำดับขั้นที่ 1 (Step 1) โดยมีค่าสถิติ F เท่ากับ 32.456 และในลำดับขั้นที่ 2 (Step 2) ตัวแปรอิสระ Flexibility ได้ถูกนำเข้าสู่สมการความถดถอยโดยมีค่าสถิติ F เท่ากับ 25.059

ตารางที่ 16 Variables in the Analysis แสดงขั้นตอนการนำตัวแปรอิสระเข้าสู่สมการความถดถอย โดยนำตัวแปรอิสระทั้ง 5 ตัว แต่นำเข้าสมการความถดถอยเพียง 2 ขั้นๆ ละ 1 ตัว

ตารางที่ 16 Variables in the Analysis

Step		Tolerance	F to Remove	Wilks' Lambda
1	Extendibility	1.000	32.456	
2	Extendibility	.950	14.960	.384
	Flexibility	.950	4.295	.198

จากตารางที่ 16 สามารถสรุปได้ว่า

ในขั้นที่ 1 นำตัวแปรอิสระ Extendibility มีความสัมพันธ์กับการแบ่งกลุ่มมากที่สุดเข้าสู่สมการความถดถอย โดยมีค่า F เท่ากับ 32.456 และในขั้นที่ 2 นำตัวแปรอิสระ Flexibility ที่มีความสัมพันธ์กับการแบ่งกลุ่มมากที่สุดในขั้นที่ 2 โดยมีค่า F เท่ากับ 4.295 โดยตารางที่ 16 จะมีความสัมพันธ์กับตารางที่ 17

ตารางที่ 17 Variables Not in the Analysis แสดงค่าตัวแปรที่ไม่ได้เข้าสู่สมการความถดถอย โดยมีค่า F to Enter ที่กำหนด เท่ากับ 3.84

ตารางที่ 17 Variable Not in the Analysis

Step		Tolerance	Min. Tolerance	F to Enter	Wilks' Lambda
0	Flexibility	1.000	1.000	12.808	.384
	Extendibility	1.000	1.000	32.456	.198
	Reusability	1.000	1.000	3.921	.671
	Functionality	1.000	1.000	2.632	.752
	Effectiveness	1.000	1.000	22.608	.261
1	Flexibility	.950	.950	4.295	.123
	Reusability	.882	.882	3.036	.138
	Functionality	1.000	1.000	.455	.186
	Effectiveness	.814	.814	1.123	.170
2	Reusability	.522	.522	.157	.119
	Functionality	.692	.657	.174	.119
	Effectiveness	.814	.782	.624	.111

จากตารางที่ 17 สามารถวิเคราะห์ข้อมูลได้ดังนี้

ในขั้นที่ 0 ยังไม่มีการนำตัวแปรอิสระทั้ง 5 ตัวแปรเข้าสมการความถดถอย ดังนั้น ตัวแปรที่ไม่ได้อยู่ในสมการความถดถอย คือ ตัวแปรอิสระทั้ง 5 ตัวแปร

ในขั้นที่ 1 ตัวแปรอิสระ Extendibility ได้เข้าสู่สมการความถดถอย โดยพิจารณาค่า F to Enter ในขั้นที่ 0 (Step 0) มีค่ามากที่สุดคือ 32.456 และมีค่ามากกว่า 3.84 (F To Enter ที่กำหนด) ซึ่งจะมีความสอดคล้องกับตารางที่ 16 ในขั้นที่ 1

ในขั้นที่ 2 ตัวแปรอิสระ Flexibility ได้เข้าสู่สมการความถดถอย โดยพิจารณาค่า F to Enter ในขั้นที่ 1 (Step 1) มีค่ามากที่สุดคือ 4.295 และมีค่ามากกว่า 3.84 (F to Enter ที่กำหนด) ซึ่งมีความสอดคล้องกับตารางที่ 18 ในขั้นที่ 2

ในขั้นที่ 3 แสดงตัวแปรอิสระที่ไม่ได้นำเข้าสู่สมการความถดถอย โดยพิจารณาค่า F to Enter ของตัวแปรอิสระมีค่าน้อยกว่า 3.84 (F to Enter ที่กำหนด) โดยมีรายละเอียดดังนี้

ตัวแปรอิสระ Resuability มีค่า F to Enter 0.157 < 3.84 (F to Enter ที่กำหนด)

ตัวแปรอิสระ Functionality มีค่า F to Enter 0.174 < 3.84 (F to Enter ที่กำหนด)

ตัวแปรอิสระ Effectiveness มีค่า F to Enter 0.624 < 3.84 (F to Enter ที่กำหนด)

ตารางที่ 18 Standardized Canonical Discriminant Function Coefficients แสดงค่าสัมประสิทธิ์ที่อยู่ในรูปมาตรฐาน (Standardized) ของตัวแปรอิสระที่มีความสำคัญต่อการแบ่งกลุ่ม

ตารางที่ 18 Standardized Canonical Discriminant Function

	Function
	1
Flexibility	.675
Extendibility	.904

จากตารางที่ 18 สามารถวิเคราะห์ได้ว่าตัวแปรอิสระ Extendibility มีความสำคัญต่อการแบ่งกลุ่มระดับความสามารถในการบำรุงรักษาซอฟต์แวร์มากที่สุด โดยมีค่าสัมประสิทธิ์ในรูปมาตรฐานเท่ากับ 0.904 และตัวแปรอิสระ Flexibility มีความสำคัญต่อการแบ่งกลุ่มในลำดับที่สอง โดยมีค่าสัมประสิทธิ์ในรูปมาตรฐานเท่ากับ 0.675

ตารางที่ 19 Structure Matrix เป็นการแสดงค่าสหสัมพันธ์ (Correlation)ระหว่างตัวแปรอิสระแต่ละตัวกับตัวแปรคาโนนิคอล (Canonical)

ตารางที่ 19 Structure Matrix

Structure Matrix

	Function
	1
Extendibility	.753
Flexibility	.473
Functionality[a]	.365
Effectiveness[a]	.318
Reusability[a]	.137

จากตารางที่ 19 สามารถวิเคราะห์ได้ว่า ตัวแปรอิสระ Extendibility มีความสัมพันธ์กับตัวแปรโคโนนิคอลมากที่สุด โดยมีค่าสหสัมพันธ์ เท่ากับ 0.753 และตัวแปรอิสระ Flexibility โดยมีค่าสหสัมพันธ์ เท่ากับ 0.473

5.3.2 การสร้างสมการจำแนกกลุ่ม

ตารางที่ 20 Canonical Discriminant Function แสดงค่าสัมประสิทธิ์ของตัวแปรสมการจำแนกกลุ่ม ซึ่งอยู่ในรูปคะแนนดิบ โดยแสดงค่าสัมประสิทธิ์ของแต่ละตัวแปร และค่าคงที่ (Constant)

ตารางที่ 20 Canonical Discriminant Function

	Function
	1
Flexibility	.979
Extendibility	2.262
(Constant)	-4.577

จากตารางที่ 20 สามารถนำมาเขียนเป็นสมการ (2) ในรูปคะแนนดิบได้ดังนี้

Y' = -4.577+0.979(Flexibility) +2.262(Extendibility) (2)

ตารางที่ 21 Function at Group Centroids แสดงค่ากลางของกลุ่ม (Group Centroid) เป็นค่าที่ใช้ในการประเมินสมการจำแนกคาโนนิคอลด้วยค่าเฉลี่ยของกลุ่ม (Canonical Discriminate Functions Evaluated at Group Means) ซึ่งผลที่ได้จะอยู่ในรูปคะแนนดิบ (Unstandardized)

ตารางที่ 21 Function at Group Centroids

	Function
Level	1
Easy	4.787
Difficult	-1.197

จากตารางที่ 21 สามารถวิเคราะห์ผลได้ดังนี้

กลุ่ม Easy มีค่ากลางของกลุ่มเท่ากับ 4.787

กลุ่ม Difficult มีค่ากลางเท่ากับ -1.197

ตารางที่ 22 Classification Function แสดงผลการวิเคราะห์โดยนำเสนอค่าสัมประสิทธิ์ และค่าคงที่ของสมการจำแนกโดยแยกออกเป็นกลุ่ม ตามวิธีของ Fisher's (Fisher's linear discrimination function) โดยสมการที่ได้จะมีจำนวนเท่ากับจำนวนกลุ่มที่ใช้ในการวิเคราะห์

ตารางที่ 22 Classification Function

	Level	
	Easy	Difficult
Flexibility	10.131	4.274
Extendibility	20.135	6.600
(Constant)	-44.902	-6.778

Fisher's linear discriminant functions

จากตารางที่ 22 สามารถเขียนรูปแบบของสมการ (3) (4) ดังนี้

Y'_{Easy} = -44.902+10.131(Flexibility)+20.135(Extendibility) (3)

$Y'_{Difficult}$= -6.778+4.274(Flexibility)+6.600(Extendibility) (4)

5.3.3 การตรวจสอบและพิจารณาความน่าเชื่อถือของสมการกลุ่ม

ผลการวิเคราะห์จากตารางที่ 23 Classification Results แสดงเปอร์เซ็นต์ความถูกต้องในการพยากรณ์ โดยใช้ตัวแปรอิสระ 2 ตัว ในการทำนายระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ ได้พบว่าโดยวิธีการแบ่งกลุ่มเดิม(Original) สามารถแบ่งกลุ่มระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ได้ถูกต้อง 100% (จาก Footnote b) และถ้าใช้วิธีครอสวาลิเดชัน Cross-validation สามารถทำนายได้อย่างถูกต้อง 100% (จาก Footnote c)

ตารางที่ 23 Classification Results

Classification Results[b,c]

			Predicted Group Membership		
		Level	Easy	Difficult	Total
Original	Count	Easy	2	0	2
		Difficult	0	8	8
	%	Easy	100.0	.0	100.0
		Difficult	.0	100.0	100.0
Cross-validated[a]	Count	Easy	2	0	2
		Difficult	0	8	8
	%	Easy	100.0	.0	100.0
		Difficult	.0	100.0	100.0

a. Cross validation is done only for those cases in the analysis. In cross validation, each case is classified by the functions derived from all cases other than that case.

b. 100.0% of original grouped cases correctly classified.

c. 100.0% of cross-validated grouped cases correctly classified.

5.3.4 การสรุปผลการทดลอง

จากการนำเอาสมการจำแนกโคโนนิคอล (Canonical Discriminant Function) สมการที่ (2) และวิธีของฟิชเชอร์(Fisher's linear discrimination function) สมการที่ (3)และ(4) มาใช้ในการคำนวณการจำแนกกลุ่มของระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ โดยแสดงผลการคำนวณด้วยสมการจำแนกกลุ่มทั้ง 2 วิธี เปรียบเทียบกับการจำแนกกลุ่มตามแบบการทดลองวิธีเดิม ซึ่งผลที่ได้จากการทดลองสามารถสรุปได้ว่า สมการจำแนกกลุ่มทั้ง 2 วิธีสามารถจำแนกกลุ่มระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ได้สอดคล้องกับการจำแนกของกลุ่มเดิม

ตารางที่ 24 ผลการทดลองการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

ลำดับ	การทดลองวิธีเดิม							การทดลองที่ได้จากการคำนวณ					ผลการทดลอง
								ฟิชเชอร์		คาโนนิคอล			
	ความซับซ้อน (5)	ความสอดคล้อง (5)	ความสามารถในการเข้าใจ (5)	ความง่าย (5)	ความยืดหยุ่น(5)	ผลรวม (25)	ระดับ	ง่าย	ยาก	ง่าย (4.787)	ยาก (-1.197)	ระดับ	
1	2.00	0.84	4.96	3.40	1.33	12.53	Difficult	-7.7266	7.314	-0.71892		Difficult	ถูกต้อง
2	2.57	0.58	4.10	2.50	1.53	11.28	Difficult	-7.18703	8.03418	-0.74901		Difficult	ถูกต้อง
3	1.11	0.62	3.32	2.98	0.96	8.99	Difficult	-21.1729	2.05814	-2.08787		Difficult	ถูกต้อง
4	3.70	2.63	5.94	3.28	2.56	18.11	Easy	45.53775	26.3938	4.99436		Easy	ถูกต้อง
5	2.58	0.17	3.52	2.78	0.40	9.45	Difficult	-15.3411	5.37092	-1.66664		Difficult	ถูกต้อง
6	0.74	0.88	1.80	1.93	1.74	7.09	Difficult	-19.6863	2.19276	-1.86198		Difficult	ถูกต้อง
7	1.60	1.54	1.16	1.79	1.10	7.19	Difficult	2.3155	10.2244	0.47288		Difficult	ถูกต้อง
8	3.46	2.55	4.06	3.73	2.85	16.65	Easy	41.49551	24.84004	4.57844		Easy	ถูกต้อง
9	1.68	1.19	2.71	3.06	1.14	9.78	Difficult	-3.92127	8.25632	-0.2405		Difficult	ถูกต้อง
10	0.74	0.50	1.18	0.47	0.59	3.48	Difficult	-27.3376	-0.31524	-2.72154		Difficult	ถูกต้อง

5.3.5 การทดสอบความถูกต้องฟังก์ชันการทำงานของเครื่องมือ

การทดสอบความถูกต้องฟังก์ชันการทำงานของเครื่องมือเป็นการทดสอบหน้าที่การทำงานในส่วนต่างๆ ตามที่ได้ระบุไว้ในหัวข้อความต้องการด้านฟังก์ชันการทำงานในบทที่ 4 ซึ่งในการทดสอบได้อาศัยวิธีการทดสอบหน้าที่การทำงาน (Black Box Testing) มาใช้ในการทดสอบตามกรณีทดสอบ และผลการทดสอบที่ได้มีรายละเอียดดังนี้

ตารางที่ 25 การทดสอบการนำเอกสารเอ็กซ์เอ็มแอลเข้าสู่ระบบ

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC01	การนำเอกสารเอ็กซ์เอ็มแอลเข้าสู่ระบบ	เพื่อเป็นการทดสอบความถูกต้องของการทำงานของเครื่องมือในการนำเอกสารเอ็กซ์เอ็มแอลเข้าสู่ระบบ โดยการทดสอบนำเอกสารชนิดอื่นๆที่ไม่ใช่เอกสารเอ็กซ์เอ็มแอลเข้าสู่ระบบ	กรณี 1 นำเอาเอกสารอย่างอื่นที่ไม่ใช่เอกสารเอ็กซ์เอ็มแอลเข้าสู่ระบบ 1. เครื่องมือจะต้องแจ้งเตือนให้ผู้ใช้งานทราบว่าไม่สามารถนำเอาเอกสารอื่นๆ ที่ไม่ใช่เอกสารเอ็กซ์เอ็มแอลเข้าสู่ระบบได้ กรณี 2 นำเอาเอกสารเอ็กซ์เอ็มแอลที่แปลงด้วยโปรแกรมสตาร์ยูเอ็มแอลเท่านั้น เข้าสู่ระบบ	ถูกต้อง

ตารางที่ 26 การคำนวณด้วยมาตรวัดเชิงวัตถุ

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC02	การคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด	เพื่อเป็นการทดสอบการคำนวณมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัดด้วยแผนภาพคลาส	1. เครื่องมือสามารถคำนวณค่ามาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด โดยไม่เกิดปัญหาใดๆเกิดขึ้น 2. เครื่องมือสามารถแสดงผลของการคำนวณทั้ง 11 มาตรวัดที่ส่วนการแสดงผล Design Tab ได้ทั้งหมด 3.กรณีที่เกิดข้อผิดพลาด หรือไม่สามารถคำนวณค่าด้วยมาตรวัดได้ เครื่องมือจะต้องแสดงข้อความบอกถึงข้อผิดพลาดได้	ถูกต้อง

ตารางที่ 27 การผลการคำนวณด้วยมาตรวัดเชิงวัตถุ

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC03	การแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด	เพื่อเป็นการทดสอบการแสดงผลของการคำนวณมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัดได้	เครื่องมือจะต้องแสดงผลการคำนวณมาตรวัดเชิงวัตถุในรูปแบบตารางโดยมีรายละเอียด ดังนี้ 1.แสดงชื่อมาตรวัดที่ใช้ในการวัด 2. แสดงผลที่ได้จากการคำนวณตามแต่ละมาตรวัด	ถูกต้อง

ตารางที่ 28 การแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุ

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC04	การแสดงผลข้อมูลจากการคำนวณด้วยมาตรวัดเชิงวัตถุด้วยรูปแบบแผนภูมิแท่ง	เพื่อเป็นการทดสอบการแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด ในรูปแบบแผนภูมิแท่งได้	เครื่องมือจะต้องแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด ในรูปแบบแผนภูมิแท่งได้	ถูกต้อง

ตารางที่ 29 การคำนวณหาค่าคุณภาพของการออกแบบ

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC05	การคำนวณหาค่าคุณภาพของการออกแบบ	เพื่อเป็นการทดสอบเครื่องมือสามารถคำนวณหาค่าคุณภาพของการออกแบบได้	1. เครื่องมือสามารถคำนวณค่าคุณภาพของการออกแบบได้โดยไม่เกิดปัญหาใดๆเกิดขึ้น 2.เครื่องมือสามารถแสดงผลของการคำนวณที่ส่วนการแสดงผล Maintainability tab 3. กรณีที่เกิดข้อผิดพลาด หรือไม่สามารถคำนวณค่าได้ เครื่องมือจะต้องแสดงข้อความบอกข้อผิดพลาดได้	ถูกต้อง

ตารางที่ 30 การแสดงผลการคำนวณหาค่าคุณภาพของการออกแบบ

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC06	การแสดงผลของการคำนวณหาค่าคุณภาพของการออกแบบ	เพื่อเป็นการทดสอบการแสดงผลการคำนวณหาค่าคุณภาพของการออกแบบ	เครื่องมือสามารถแสดงผลการคำนวณหาค่าคุณภาพของการออกแบบในรูปแบบตารางได้	ถูกต้อง

ตารางที่ 31 การคำนวณการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC07	การคำนวณการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์	ทดสอบเครื่องมือสามารถคำนวณค่าการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ได้	1. เครื่องมือจะต้องคำนวณหาค่าระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ด้วยสมการการจำแนกกลุ่มได้ 2. กรณีที่เกิดข้อผิดพลาด หรือไม่สามารถคำนวณค่าได้ เครื่องมือจะต้องแสดงข้อความบอกข้อผิดพลาดได้	ถูกต้อง

ตารางที่ 32 การแสดงผลการคำนวณด้วยสมการจำแนกกลุ่ม

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC08	การแสดงผลข้อมูลการคำนวณด้วยสมการจำแนกกลุ่ม	ทดสอบความสามารถของเครื่องมือในการแสดงผลการคำนวณด้วยสมการจำแนกกลุ่มทั้ง 2 วิธี	เครื่องมือจะต้องแสดงผลการคำนวณการจำแนกกลุ่มได้อย่างถูกต้อง	ถูกต้อง

ตารางที่ 33 การแสดงผลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC09	เครื่องมือจะต้องสามารถบอกกลุ่มของระดับการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ได้	เพื่อเป็นการทดสอบการจำแนกกลุ่มของการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์	เครื่องมือจะต้องสามารถจำแนกกลุ่มของการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ด้วยข้อความต่อไปนี้ “This system is easy to maintain.”ใช้สำหรับบอกกลุ่มของความสามารถในการบำรุงรักษาซอฟต์แวร์ในระดับง่ายและ “This system is difficult to maintain.” ใช้สำหรับบอกกลุ่มของความสามารถในการบำรุงรักษาซอฟต์แวร์ในระดับยาก	ถูกต้อง

ตารางที่ 34 การนำข้อมูลออก (Export data)

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC10	การนำข้อมูลออก (Export data)	เพื่อเป็นการทดสอบเครื่องมือในความสามารถการนำข้อมูลออก (Export Files) ในรูปแบบของเอกสารเวิร์ด (Word) และเอกสารเอ็กเซล (Excel)ได้	เครื่องมือจะต้องนำข้อมูลออก (Export data)ในรูปแบบเอกสารดังต่อไปนี้ เอกสารเอ็กเซล (Excel) โดยจะต้องมีนามสกุลไฟล์เป็น .xls เท่านั้น เอกสารเวิร์ด(Word) โดยจะต้องมีนามสกุลเป็น .doc เท่านั้น ถ้ามีความผิดพลาดในขั้นตอนนี้ เครื่องมือจะต้องสามารถแสดงข้อความบอกข้อผิดพลาดให้แก่ผู้ใช้งานได้รับทราบปัญหาได้	ถูกต้อง

ตารางที่ 35 การตั้งชื่อไฟล์เอกสารนำออก

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC11	การตั้งชื่อไฟล์เอกสารของข้อมูลที่ถูกนำออก (Export data)	ทดสอบความถูกต้องของชื่อเอกสารการนำข้อมูลออก (Export data)	เครื่องมือจะต้องสามารถตั้งชื่อไฟล์เอกสารในรูปแบบ “ชื่อไฟล์เอกสาร_วันเดือนปี.นามสกุลของไฟล์”ได้อย่างถูกต้อง	ถูกต้อง

ตารางที่ 36 การดาวน์โหลดคู่มือการใช้งาน

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC12	การดาวน์โหลดคู่มือการใช้งาน	ทดสอบความสามารถของของเครื่องมือ ในการดาวน์โหลดเอกสารการใช้งาน	เครื่องมือจะต้องสามารถทำการดาวน์โหลดเอกสารคู่มือการใช้งานได้	ถูกต้อง

ตารางที่ 37 การแสดงผลภาษาไทยและภาษาอังกฤษของเครื่องมือ

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC13	การทดสอบการแสดงผลภาษาไทยและภาษาอังกฤษของเครื่องมือ	ทดสอบความสามารถของเครื่องมือ ในการแสดงผลข้อมูลทั้งภาษาไทย และภาษาอังกฤษ	เครื่องมือจะต้องสามารถแสดงผลภาษาตามที่ผู้ใช้งานเลือกได้อย่างถูกต้อง	ถูกต้อง

5.4 สรุปผลการทดลองและการทดสอบ

จากผลการลองการจำแนกกลุ่มของระบบ 10 ระบบ และผลการทดสอบเครื่องมือตามกรณีทดสอบข้างต้นสรุปได้ว่า การจำแนกกลุ่มระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ที่คำนวณด้วยสมการจำแนกกลุ่มสามารถจำแนกกลุ่มระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ได้สอดคล้องกับการจำแนกกลุ่มแบบเดิม และผลการทดสอบของเครื่องมือสามารถสรุปได้ว่าเครื่องมือสามารถทำงานได้อย่างถูกต้องตามกรณีทดสอบทั้งหมด

บทที่ 6

สรุปผลการวิจัย และข้อเสนอแนะ

ในบทนี้จะกล่าวถึงบทสรุปที่ได้ทั้งหมดจากการดำเนินการวิจัย และร่วมถึงข้อเสนอแนะต่างๆ

6.1 สรุปผลการวิจัย

ผลการวิจัยการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ในขั้นตอนของการออกแบบเป็นไปตามที่ได้คาดหวังไว้ นั้นคือ ได้โมเดลการวัดความสามารถในการบำรุงรักษา และได้เครื่องมือเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ โดยสามารถสรุปผลการวิจัยที่ได้ดังต่อไปนี้

6.1.1 โมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

ในขั้นตอนของการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ได้สมการการจำแนกกลุ่มระดับความสามารถในการบำรุงรักษาซอฟต์แวร์จากวิธี 2 วิธี คือ สมการที่อยู่รูปแบบคะแนนดิบ (Canonical Discriminate Function Coefficients) หรือสมการจำแนกคาโนนิคอล และสมการจำแนกกลุ่มของ Fisher's (Fisher's linear discrimination function) สามารถจำแนกกลุ่มระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ได้ถูกต้องสอดคล้องกับการจำแนกกลุ่มแบบเดิม

6.1.2 เครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

ผลการทดสอบของเครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์สามารถทำงานได้ถูกต้องตรงตามกับความต้องการด้านหน้าที่ (Functional Requirements) และไม่ใช่หน้าที่(Non Functional Requirements) ตามที่กำหนดไว้ในบทที่ 4

สรุปได้ว่างานวิจัยนี้เป็นไปตามที่ได้คาดหวังไว้ ได้โมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ และได้เครื่องมือเพื่อใช้ในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ เพื่อช่วยให้ผู้ออกแบบระบบสามารถทำ

การทวบทวน ปรับปรุงการออกแบบซอฟต์แวร์ให้มีความง่ายต่อการทำการบำรุงรักษาซอฟต์แวร์

6.2 ข้อเสนอแนะ

1) กลุ่มของการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ของงานวิจัยนี้ มีเพียง 2 ระดับ คือ ระดับง่าย และระดับยาก ซึ่งสามารถบอกค่าความสามารถในการบำรุงรักษาซอฟต์แวร์ได้เพียงคราวๆ เท่านั้น ถ้าหากสามารถบอกค่าระดับความสามารถในการบำรุงรักษาซอฟต์แวร์ได้ละเอียดขึ้น ก็จะเป็นการช่วยเพิ่มความสามารถในการตัดสินใจของผู้ออกแบบระบบ
2) วิธีการจำแนกกลุ่มของการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์นอกจากวิธีการจำแนกด้วยค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน ยังมีวิธีอื่นที่สามารถนำมาให้ในการจำแนกกลุ่ม เช่น สถิติที่ไม่ใช้พารามิเตอร์ (Nonparametric Statistics ได้แก่ ไคสแควร์, Median Test, Sign test สามารถนำมาประยุกต์ใช้ในการจำแนกกลุ่มของการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ได้
3) เครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ของงานวิจัยนี้ สามารถวัดค่าความสามารถในการบำรุงรักษาซอฟต์แวร์ได้ และสามารถบอกรายละเอียดเกี่ยวกับคุณสมบัติการออกแบบ ค่าคุณภาพของการออกแบบซอฟต์แวร์ เพื่อเป็นประโยชน์ต่อผู้ออกแบบระบบใช้ในการวิเคราะห์และแก้ไขการออกแบบ แต่ถ้าเครื่องมือสามารถบอกแนวทางในการแก้ไขปัญหาของการออกแบบได้ ก็จะเป็นการช่วยการทำงานให้แก่ผู้ออกแบบระบบได้ดียิ่งขึ้น

รายการอ้างอิง

[1] นงเยาว์ จินดาสวัสดิ์. การทำนายความสามารถในการบำรุงรักษาซอฟต์แวร์โดยใช้มาตรวัดการวิเคราะห์และออกแบบเชิงวัตถุแบบยูเอ็มแอล. มหาบัณฑิต วิทยาศาตร์คอมพิวเตอร์ วิศวกรรมศาตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2546.

[2] Mazhar Khaliq, RiyaZ A. Khan and M.H. Khan. Significance of Design Properties in Object Oriented Software Product Quality Assessment. International Journal of Computing Science and Communication Technologies, 3(2): 622-625, Jan. 2011.

[3] Rimmi Saini, Sanjay Kumar Dubey and Ajay Rana. ANALYTICAL STUDY OF MAINTAINABILITY MODELS FOR QUALITY EVALUATION. Indian Journal of Computer Science and Engineering 2(3): 449-454,2011.

[4] Bansiya, J. and C. G. Davis. A hierarchical model for object-oriented design quality assessment. IEEE Transaction on software engineering, 28: 4-17, 2002.

[5] Rizvi, S. W. A. and R. A. Khan. Maintainability Estimation Model for Object-Oriented Software in Design Phase (MEMOOD). Journal of Computing, 2(4): 26-32, April 2010.

[6] GPL. StarUML [Online]. Available from: www.staruml.sourceforge.net/en/index.php:[2012, October].

[7] กัลยา วินิชย์บัญชา. การวิเคราะห์สถิติขั้นสูงด้วย SPSS for Windows. กรุงเทพ: ศูนย์หนังสือแห่งจุฬาลงกรณ์มหาวิทยาลัย, 2554.

[8] กัลยา วินิชย์บัญชา. สถิติสำหรับงานวิจัยหลักการเลือกใช้เทคนิคทางสถิติในงานวิจัย พร้อมทั้งอธิบายผลลัพธ์ที่ได้จาก SPSS.กรุงเทพ: ศูนย์หนังสือแห่งจุฬาลงกรณ์มหาวิทยาลัย, 2552.

[9] Kuljit Kaur and Hardeep Singh, Exploring Design Level Class Cohesion Metrics. Journal of Software Engineering & Application, 3:384-390, 2010.

[10] Marcela Genero, Mario Piattini, Coral Calero. A Survey of Metrics for UML Class Diagrams. Journal of Object Technology, 4(9):59-92, 2005.

[11] Matinlassi Mari and Niemela Eila. The Impact of Maintainability on Component-based Software Systems. Proceedings of the 29th Conference on EUROMICRO, IEEE Computer Society, 25-32, 2003.

[12] Andreas Jetter. Assessing Software Quality Attributes with Source Code Metrics. University of Zurich, 2006.

[13] Cristina Cachero and Jaime Gomez. Advanced Conceptual Modeling of Web Application: Embedding Operation Interfaces in Navigation Design. International Conference on Conceptual Modeling, 2002.

[14] Frederick T. Sheldon and Young-Jik Kwon. Case Study: Implementing a Web Based Auction System using UML and Component-Based Programming. International Computer Software and Applications Conference, England, 26-29 August, 2002.

[15] Kaavya Kuppa. Component Design Airline Reservation System. Kansas State University. 2008.

[16] Lai Chi Wa and Chung Wang Leong. E-ticket System.2010.

[17] Nathan Joyes. Internation Internet Café [Online]. Available from: www.cafesource.sourceforge.net. 2011.

[18] Rajni Pamnani,Pramila Chawan and Satish Salunkhe. Object Oriented UML Modeling for ATM Systems. International Conference on Recent Trends in Computer Engineering, 2010.

[19] Program & Note for MCA.UML Diagrams for Hospital Management [Online]. Available from: www.programsformca.com.2012.

[20] Taysir Hassan A. Soliman, Adel El-Swesy and Saddam Hussein Ahmed.Utilizing CK Metrics Suite to UML Models: a Case Study of Microarray MIDAS Software. Informatics and Systems (INFOS), 1-6, 28-30 March 2010.

[21] Asad siddiqi. Object Oriented Design – A Class Diagram Walkthrough [Online]. Available from: www.asadsiddiqi.wordpress.com/2008/02/21/object-oriented-design-a-class-diagram-walkhrough.2012.

ภาคผนวก

ภาคผนวก ก.
คำอธิบายยูสเคส

ตารางที่ 38 คำอธิบายยูสเคสการคำนวณของเครื่องมือ

หมายเลขยูสเคส : UC1	**ชื่อยูสเคส** :การคำนวณของเครื่องมือ
ผู้เกี่ยวข้องหลัก :ผู้ใช้งาน	
รายละเอียด : เป็นส่วนการคำนวณของเครื่องมือ ประกอบด้วย 3 ฟังก์ชันหลักของการทำงาน ได้แก่ ฟังก์ชันการคำนวณด้วยมาตรวัดเชิงวัตถุ 11 มาตรวัด ฟังก์ชันการคำนวณหาค่าคุณภาพของการออกแบบ และฟังก์ชันการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์	
ความสัมพันธ์ : Association : ผู้ใช้งาน Use : - Extend : - Generalization : - Include: -	
กระแสเหตุการณ์ปกติ: 1. เลือกเอกสารเอ็กซ์เอ็มแอลของแผนภาพคลาส เพื่อนำเข้าสู่ระบบ 2. เลือก “คำนวณ” เครื่องมือจะแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด 3. เครื่องมือแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด ในรูปแบบของแผนภูมิแท่ง 4. เครื่องมือแสดงผลการคำนวณค่าคุณภาพของการออกแบบทั้ง 5 ค่า 5. เครื่องมือจะแสดงผลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์	
กระแสเหตุการณ์ทางเลือก:-	

ตารางที่ 39 คำอธิบายยูสเคสการคำนวณด้วยมาตรวัดเชิงวัตถุ

หมายเลขยูสเคส : UC2	**ชื่อยูสเคส** :การคำนวณด้วยมาตรวัดเชิงวัตถุ
ผู้เกี่ยวข้องหลัก : การคำนวณของเครื่องมือ	
รายละเอียด : เป็นฟังก์ชันในการคำนวณด้วยมาตรวัดเชิงวัตถุ 11 มาตรวัด	
ความสัมพันธ์ : Association : การคำนวณของเครื่องมือ Use : - Extend : - Generalization : - Include: การคำนวณด้วยมาตรวัดเชิงวัตถุ การคำนวณหาค่าคุณภาพของการออกแบบ และการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์	
กระแสเหตุการณ์ปกติ: 1. เลือกไฟล์เอกสารเอ็กซ์เอ็มแอลของแผนภาพคลาส เพื่อนำเข้าสู่ระบบ 2. เลือก “คำนวณ” 3. เครื่องมือจะทำการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด 4. เครื่องมือแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัดในรูปของแผนภูมิแท่ง	
กระแสเหตุการณ์ทางเลือก:-	

ตารางที่ **40** คำอธิบายยูสเคสการคำนวณหาค่าคุณภาพของการออกแบบ

หมายเลขยูสเคส : UC3	**ชื่อยูสเคส**: การคำนวณหาค่าคุณภาพของการออกแบบ
ผู้เกี่ยวข้องหลัก :การคำนวณของเครื่องมือ	
รายละเอียด : เป็นฟังกชันในการคำนวณหาค่าคุณภาพของการออกแบบซอฟต์ทั้ง 5 ค่า ได้แก่ ความสามารถในการนำกลับมาใช้ใหม่ (Reusability) ความสามารถในการยืดหยุ่น (Flexibility) ความสามารถในการทำงาน (Functionality) ความสามารถในการขยาย (Extendibility) และความสามารถในการมีประสิทธิภาพ (Effectiveness) โดยนำเอาผลลัพธ์ที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุ 11 มาตรวัด มาแทนที่ให้สมการในการคำนวณหาค่าคุณภาพของการออกแบบ	
ความสัมพันธ์ : Association : การคำนวณของเครื่องมือ Use : - Extend : - Generalization : - Include : -	
กระแสเหตุการณ์ปกติ: 1. เลือกไฟล์เอกสารเอ็กซ์เอ็มแอลของแผนภาพคลาส เพื่อนำเข้าสู่ระบบ 2. เลือก “คำนวณ” 3. เครื่องมือจะแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด 4. ผลลัพธ์ที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุ 11 มาตรวัด ถูกนำมาแทนที่ในสมการในการคำนวณหาค่าคุณภาพของการออกแบบ 5. แสดงผลลัพธ์ที่ได้จากการคำนวณหาค่าคุณภาพของการออกแบบ	
กระแสเหตุการณ์ทางเลือก:-	

ตารางที่ 41 คำอธิบายยูสเคสการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

หมายเลขยูสเคส : UC4	**ชื่อยูสเคส**: การวัดความสามารถในการบำรุงรักษาซอฟต์แวร์
ผู้เกี่ยวข้องหลัก :การคำนวณของเครื่องมือ	
รายละเอียด : เป็นฟังก์ชันการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์	
ความสัมพันธ์ : Association : การคำนวณของเครื่องมือ Use : - Extend : - Generalization : - Include : การสร้างโมเดลในการทำนาย	
กระแสเหตุการณ์ปกติ: 1. เลือกไฟล์เอกสารเอ็กซ์เอ็มแอลของแผนภาพคลาส เพื่อนำเข้าสู่ระบบ 2. เลือก “คำนวณ” 3. เครื่องมือจะแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุทั้ง 11 มาตรวัด 4. ผลลัพธ์ที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุ 11 มาตรวัด ถูกนำมาแทนที่ในสมการในการคำนวณหาค่าคุณภาพของการออกแบบ 5. แสดงผลลัพธ์ที่ได้จากการคำนวณหาค่าคุณภาพของการออกแบบ ผลลัพธ์ที่ได้จะถูกนำไปคำนวณด้วยสมการการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ 6. แสดงผลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์	
กระแสเหตุการณ์ทางเลือก:-	

ตารางที่ 42 คำอธิบายยูสเคสการสร้างโมเดล

หมายเลขยูสเคส : UC5	**ชื่อยูสเคส** :การสร้างโมเดลการวัดความสามารถในการบำรุงรักษา
ผู้เกี่ยวข้องหลัก :การวัดความสามารถในการบำรุงรักษาซอฟต์แวร์	
รายละเอียด : เป็นขั้นตอนการสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ด้วยการวิเคราะห์จากโปรแกรมเอสพีเอสเอส โดยนำเอาสมการที่ได้จากการวิเคราะห์มาใช้เป็นฟังก์ชันในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ให้แก่เครื่องมือ	
ความสัมพันธ์ : Association : การวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ Use : - Extend : - Generalization : - Include : -	
กระแสเหตุการณ์ปกติ: 1. เป็นการวิเคราะห์ข้อมูลเพื่อสร้างโมเดลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ด้วยโปรแกรมเอสพีเอสเอส 2. นำเอาสมการที่ได้จากการวิเคราะห์ด้วยโปรแกรมเอสพีเอสเอสมาใช้เป็นฟังก์ชันในการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ให้แก่เครื่องมือ	
กระแสเหตุการณ์ทางเลือก:-	

ตารางที่ 43 คำอธิบายยูสเคสการนำข้อมูลออก

หมายเลขยูสเคส : UC6	**ชื่อยูสเคส** :การนำข้อมูลออก
ผู้เกี่ยวข้องหลัก :ผู้ใช้งาน	
รายละเอียด : เป็นขั้นตอนของการนำข้อมูลออกจากระบบ ในรูปแบบของเอกสารเวิร์ด (Word) และเอกสารเอ็กเซล (Excel)	
ความสัมพันธ์ : Association : ผู้ใช้งาน Use : - Extend : - Generalization : - Include: -	
กระแสเหตุการณ์ปกติ: 1. ผู้ใช้งานทำการเลือกชนิดไฟล์เอกสารที่ต้องการนำเอาข้อมูลออก 2. เลือก “นำเอาข้อมูลออก” 3. ข้อมูลจะถูกบันทึกตามไฟล์เอกสารที่ผู้ใช้งานเลือก	
กระแสเหตุการณ์ทางเลือก:-	

ตารางที่ 44 คำอธิบายยูสเคสการเปลี่ยนภาษา

หมายเลขยูสเคส : UC7	**ชื่อยูสเคส** :การเปลี่ยนภาษา
ผู้เกี่ยวข้องหลัก :ผู้ใช้งาน	
รายละเอียด : เป็นขั้นตอนของการเปลี่ยนภาษาของเครื่องมือ	
ความสัมพันธ์ : Association : ผู้ใช้งาน Use : - Extend : - Generalization : - Include: -	
กระแสเหตุการณ์ปกติ: 1. เครื่องมือจะแสดงภาษาตามแต่ละเครื่องคอมพิวเตอร์ของผู้ใช้งานตั้งค่าไว้ ตัวอย่างเช่น ผู้ใช้งานใช้ระบบปฏิบัติการวินโดว์ 7 เวอร์ชันภาษาไทย ระบบการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ก็จะแสดงผลเป็นภาษาไทยเป็นภาษาหลัก 2. ผู้ใช้งานสามารถที่จะเลือกเปลี่ยนภาษาที่ใช้ในการแสดงผลได้ ได้แก่ ภาษาไทย และภาษาอังกฤษ	
กระแสเหตุการณ์ทางเลือก:-	

ตารางที่ 45 คำอธิบายยูสเคสการดาวน์โหลดคู่มือการใช้งาน

หมายเลขยูสเคส : UC8	**ชื่อยูสเคส** :การดาวน์โหลดคู่มือการใช้งาน
ผู้เกี่ยวข้องหลัก :ผู้ใช้งาน	
รายละเอียด : เป็นขั้นตอนของการดาวน์โหลดคู่มือการใช้งาน	
ความสัมพันธ์ : Association : ผู้ใช้งาน Use : - Extend : - Generalization : - Include: -	
กระแสเหตุการณ์ปกติ: 1. เข้าไปที่เมนูช่วยเหลือ (Helps) 2. ในเมนูช่วยเหลือจะแสดงเอกสารคู่มือการใช้งานของเครื่องมือ โดยจะแบ่งออกเป็น 2 ภาษา คือ คู่มือการใช้งานภาษาไทย และคู่มือการใช้งานภาษาอังกฤษ 3. เลือกคู่มือการใช้งานจะปรากฏหน้าจอให้เลือกบันทึกเอกสารเครื่องมือจะทำการบันทึกเอกสารไว้ที่เครื่องคอมพิวเตอร์ของผู้ใช้งาน 4. เปิดเอกสารคู่มือการใช้งานเครื่องมือ	
กระแสเหตุการณ์ทางเลือก:-	

ภาคผนวก ข.
รายละเอียดของแผนภาพคลาส

1. คลาส Default

ชื่อคลาส	Default
คุณลักษณะ	XDocument xdoc; XNamespace UML;
เมธอด	public void DisplayDesignMetric (XDocument xdoc, XNamespace UML) public void DisplayQualityMeric() public void DisplayChart()

2. คลาส DesignSizeMeasure

ชื่อคลาส	DesignSizeMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String DSC;
เมธอด	public DesingSizeMesure(XDocument xdoc XNamespace UML); public void DesignSizeInClass()

3. คลาส HieratchiesMeasure

ชื่อคลาส	HieratchiesMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String NOH;
เมธอด	public HieratchiesMeasure(XDocument xdoc,XNamespace UML) public void NumberofHieretchies()

4. คลาส AbstractionMeasure

ชื่อคลาส	AbstractionMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String VANA;
เมธอด	public AbstractionMeasure(XDocument xdoc, XNamespace UML) public AverageNumberOfAncestors()

5. คลาส EncapsulationMeasure

ชื่อคลาส	EncapsulationMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String VDAM;
เมธอด	public EncapsulationMeasure(XDocument xdoc, XNamespace UML) public void DataAccessMetric()

6. คลาส CouplingMeasure

ชื่อคลาส	CouplingMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String VDCC;
เมธอด	public CouplingMeasure(XDocument xdoc, XNamespace UML) public void DirectClassCoupling()

7. คลาส CohesionMeasure

ชื่อคลาส	CohesionMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String VCAM;
เมธอด	public CohesionMeasure(XDocument xdoc, XNamespace UML) public void CohesionAmongMethodsOfClass()

8. คลาส CompositionMeasure

ชื่อคลาส	CompositionMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String VMOA;
เมธอด	public CompositionMeasure(XDocument xdoc, XNamespace UML) public void MeasureOfAggregation()

9. คลาส InheritanceMeasure

ชื่อคลาส	InheritanceMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String VMAF;
เมธอด	public InheritanceMeasure(XDocument xdoc, XNamespace UML) public void MeasureOfFunctionalAbstraction()

10. คลาส PolymorphismMeasure

ชื่อคลาส	PolymorphismMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String VNOP;
เมธอด	public PolymorphismMeasure(XDocument xdoc, XNamespace UML) public void NumberOfPolymorphicMethods()

11. คลาส MessagingMeasure

ชื่อคลาส	MessagingMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String VCIS;
เมธอด	public MessagingMeasure(XDocument xdoc, XNamespace UML) public void ClassInterfaceSize()

12. คลาส ComplexityMeasure

ชื่อคลาส	ComplexityMeasure
คุณลักษณะ	XDocument xdoc; XNamespace UML; String VNOM;
เมธอด	public ComplexityMeasure(XDocument xdoc, XNamespace UML) public void NumberOfMethods()

13. คลาส QualityMeasures

ชื่อคลาส	QualityMeasures
คุณลักษณะ	
เมธอด	public void CalculateReusability() public void CalculateFlexibility() public void CalculateFunctionality() public void CalculateExtendibility() public void CalculatEffectiveness()

14. คลาส PredictMaintainability

ชื่อคลาส	PredictMaintainability
คุณลักษณะ	double dFlexibility; double dExtendibility;
เมธอด	public void CalculatePredictMaintainabilityFishers1(Double dFlexibility, Double dExtendibility) public void CalculatePredictMaintainabilityFishers2(Double dFlexibility, Double dExtendibility) public void CalculatePredictCanonical(Double dFlexibility, Double dExtendibility)

15. คลาส GenerateChart

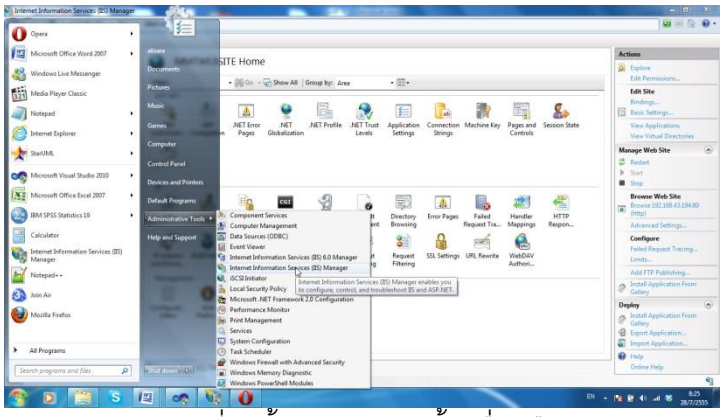
ชื่อคลาส	GenerateChart
คุณลักษณะ	
เมธอด	public void ChartDesignQuality ()

16. คลาส ExportData

ชื่อคลาส	ExportData
คุณลักษณะ	
เมธอด	protected void ExportFileSelection (string FileName); Public static void PrepareControlForExport (Control control)

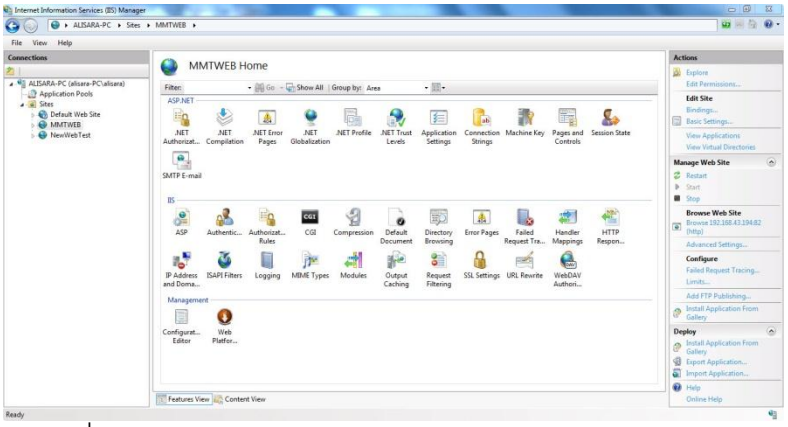
ภาคผนวก ค.
การติดตั้ง และการใช้งานเครื่องมือ

1 การติดตั้งเครื่องมือ



รูปที่ 9 ขั้นตอนการติดตั้งเครื่องมือ

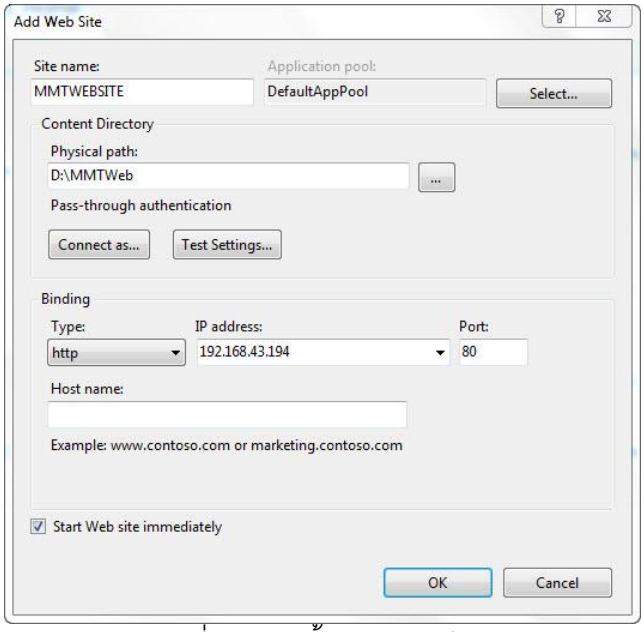
1) คลิก Start Menu เลือก Administrative Tools เลือก Internet Information Service (IIS) Manager. ตามรูปที่ 9 ขั้นตอนการติดตั้งเครื่องมือ



รูปที่ 10 หน้าจอ Internet Information Service (IIS) Manager

รูปที่ 10 แสดงหน้าจอ Internet Information Service (IIS) Manager สำหรับการติดตั้งเว็บไซต์

2) คลิกขวาที่ Site ในช่อง Connection และเลือก Add Web Site จะปรากฏหน้าจอ ดังรูปที่ 11 แสดงหน้าจอการตั้งค่าของเว็บไซต์



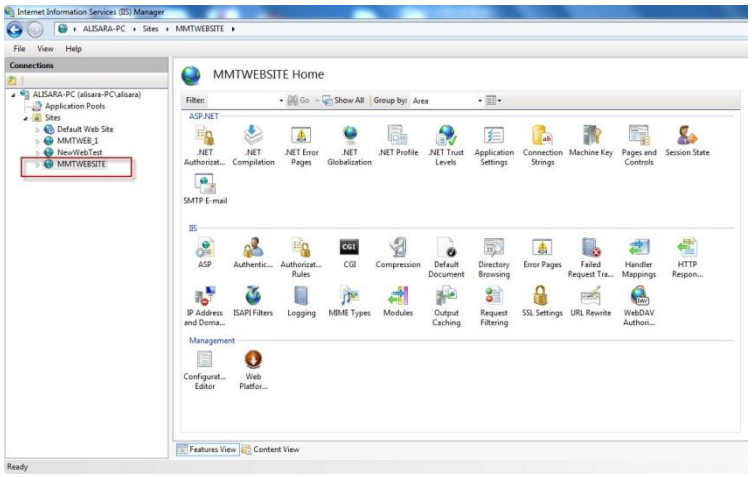
รูปที่ 11 การตั้งค่าของเว็บไซต์

ในการตั้งค่าต่างๆของเว็บไซต์ ประกอบด้วย

- *Site name* คือ ชื่อของเว็บไซต์ (ในตัวอย่างใช้เป็นชื่อ MMTWEBSITE)
- *Application pool* เลือกเป็น DefaultAppPool ให้ตรงกับเวอร์ชั่นที่ใช้งาน
- *Physical path* เลือกพาธที่ใช้ในการเก็บเว็บไซต์
- *IP Address* ระบุค่าไอพีของเครื่องคอมพิวเตอร์ที่ใช้

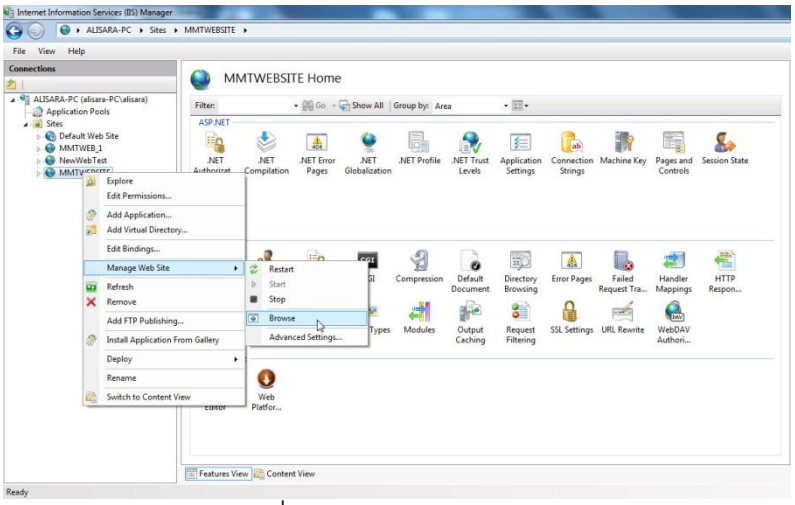
เมื่อระบุค่าต่างๆเสร็จสิ้นแล้วให้คลิก OK

ในรูปที่ 12 แสดงรายการเว็บไซต์ใหม่ที่ได้ทำการสร้างขึ้นมาจากขั้นตอนที่แล้ว



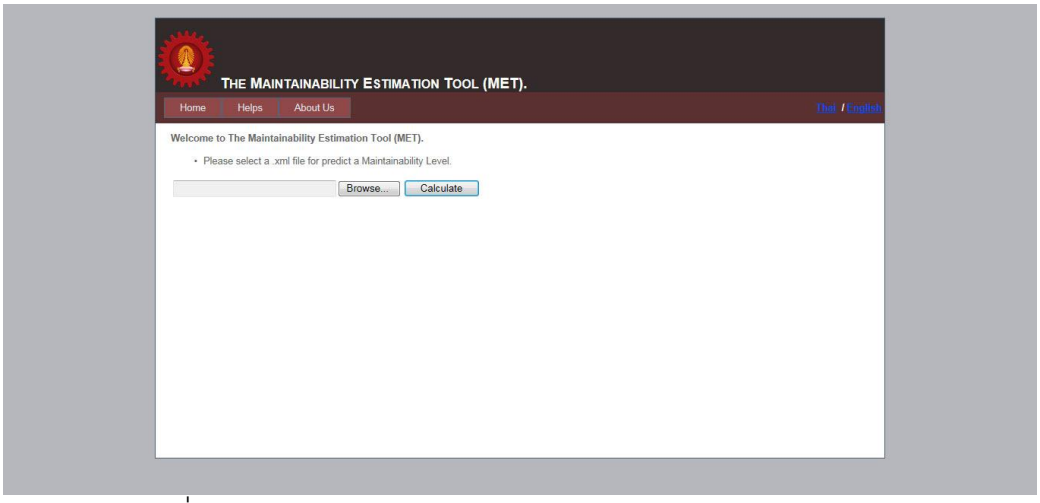
รูปที่ 12 แสดงรายชื่อเว็บไซต์

การเรียกดูเว็บไซต์ที่สร้างขึ้น รูปที่ 13 แสดงวิธีการเรียกดูเว็บไซต์



รูปที่ 13 วิธีการเรียกดูเว็บไซต์

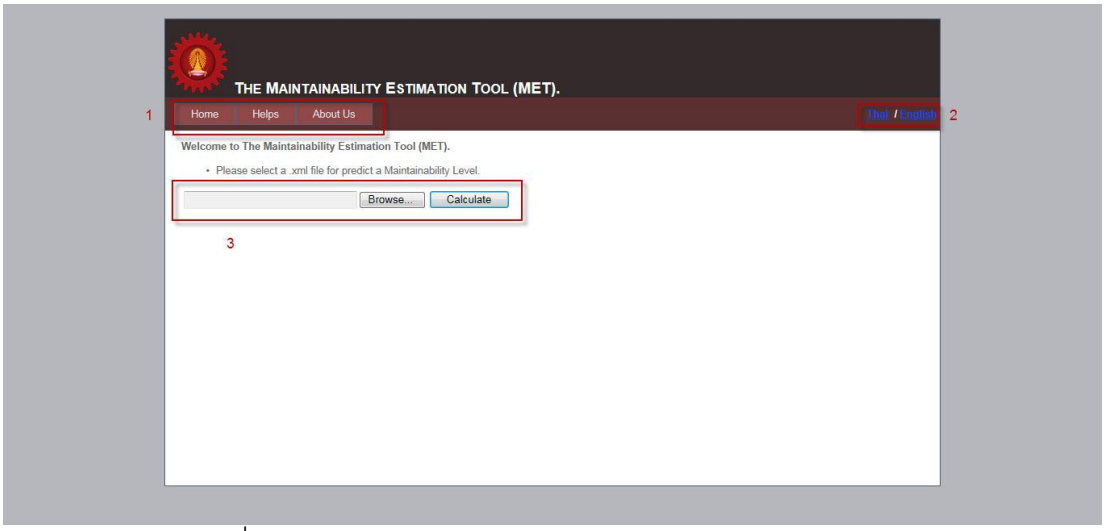
- คลิกขวาที่ MMTWEBSITE เลือก Manage Web Site และเลือก Browse หรือ จะใช้วิธีในการระบุค่า IP Address ที่ Internet Explore URL: http://192.168.43.194/จะปรากฏเว็บไซต์ ดังรูปที่ 14



รูปที่ 14 เว็บไซต์ THE MAINTAINABILITY ESTIMATION TOOL (MET).

2.1 คำอธิบายส่วนประกอบต่าง ๆของเครื่องมือ

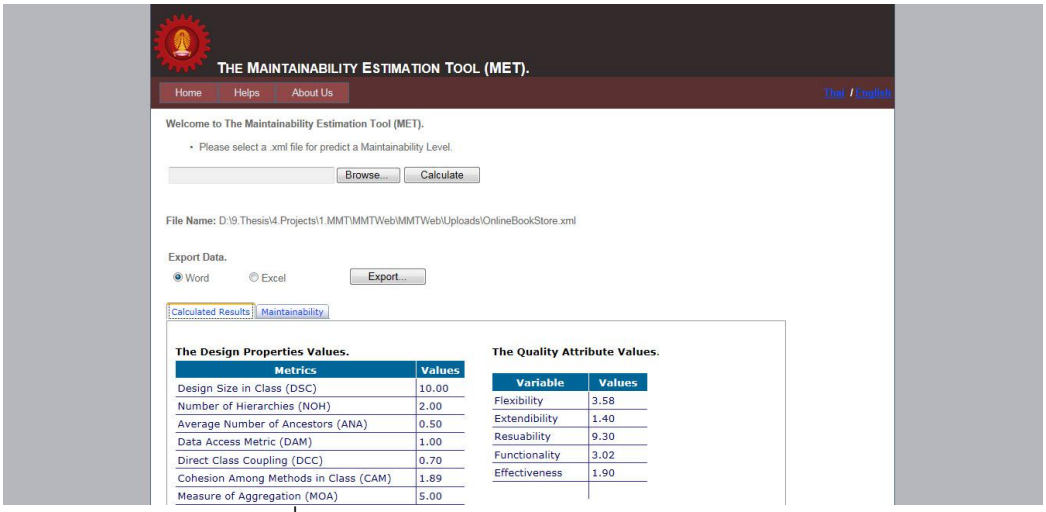
เครื่องมือการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ (THE MAINTAINABILITY ESTMATION TOOL: MET) เป็นระบบที่พัฒนาขึ้นในรูปแบบของเว็บแอพพริเคชัน โดยมีส่วนประกอบต่างๆ ดังรูปที่15 แสดงหน้าจอของระบบการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์



รูปที่ 15 ระบบการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

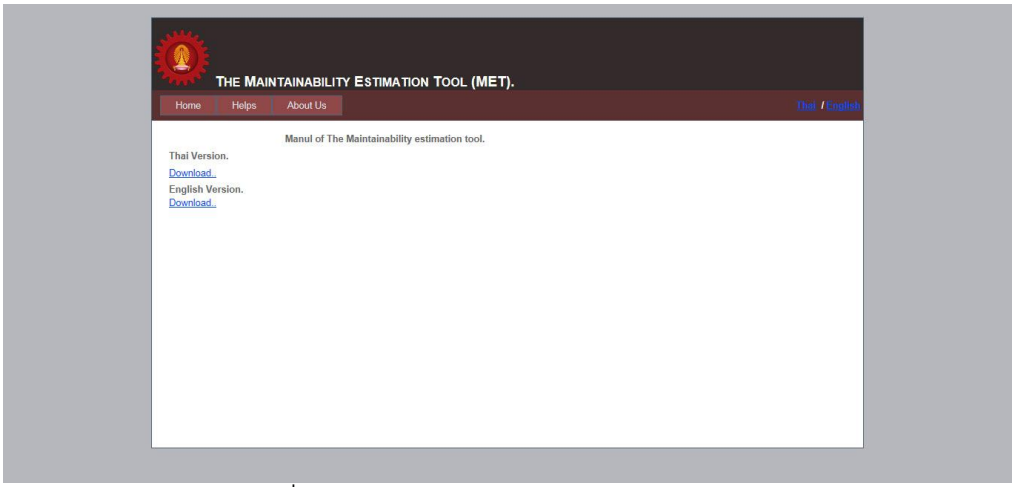
1) เมนูประกอบด้วย 3 เมนู ได้แก่ Home (หน้าหลัก), Helps (ช่วยเหลือ), About Us (เกี่ยวกับเรา)

- เมนู Home (หน้าหลัก) จะเป็นเมนูหลักที่ใช้แสดงผลข้อมูลที่ได้จากการคำนวณด้วยมาตรวัดเชิงวัตถุ คำนวณหาคุณภาพของการออกแบบซอฟต์แวร์ และการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ ดังรูปที่ 14 แสดงส่วนประกอบของหน้า Home (หน้าหลัก)



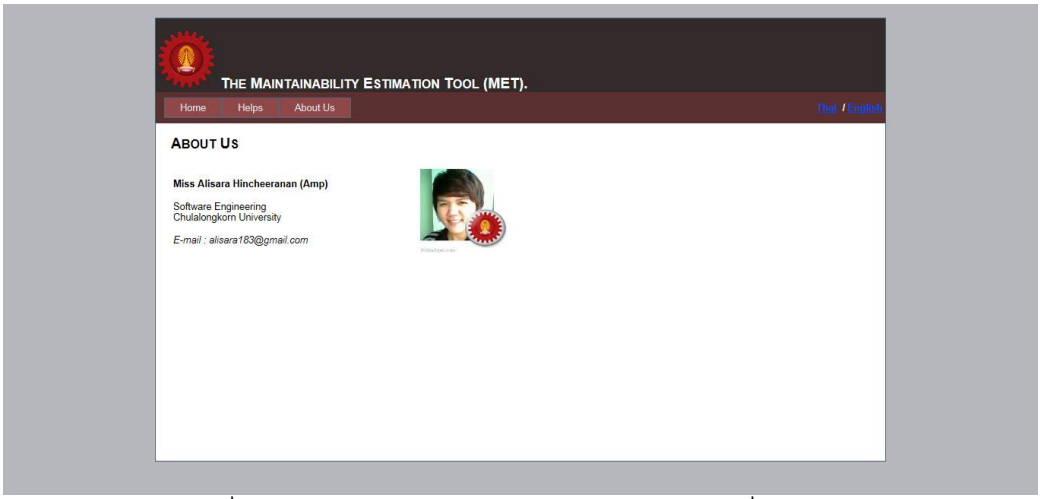
รูปที่ 16 แสดงส่วนประกอบของเมนู Home (หน้าหลัก)

- เมนู Helps (ช่วยเหลือ) เพื่อแสดงคู่มือการใช้งานระบบ โดยทางผู้พัฒนาระบบ ได้จัดทำคู่มือการใช้งานทั้งภาษาไทย และภาษาอังกฤษ ดังรูปที่ 17 แสดงส่วนประกอบของเมนู Helps (ช่วยเหลือ)



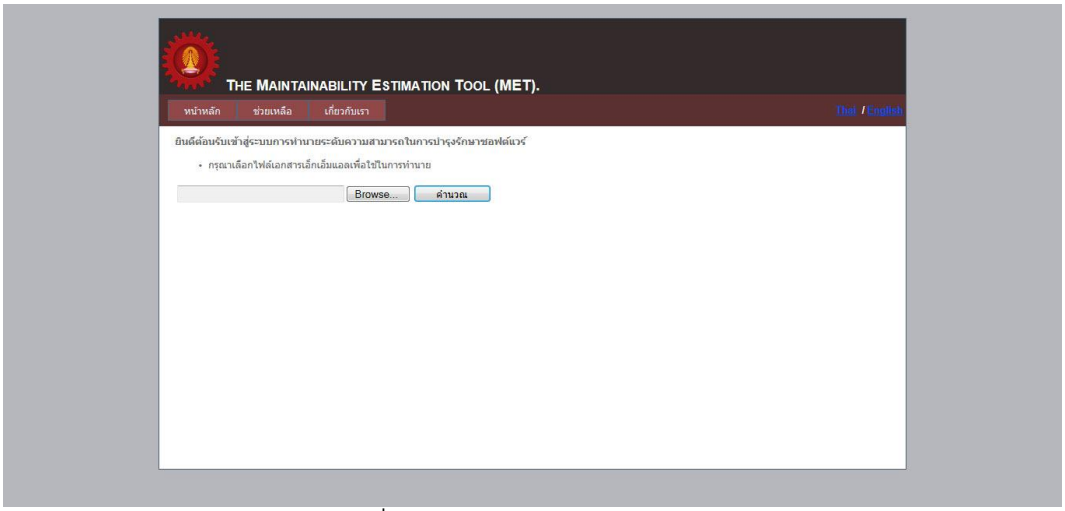
รูปที่ 17 แสดงส่วนประกอบของเมนู Helps (ช่วยเหลือ)

- เมนู About Us (เกี่ยวกับเรา) เป็นเมนูแสดงข้อมูลของผู้พัฒนาระบบ ดังรูปที่ 18 แสดงส่วนประกอบของเมนู About Us (เกี่ยวกับเรา)



รูปที่ 18 แสดงส่วนประกอบของเมนู About Us (เกี่ยวกับเรา)

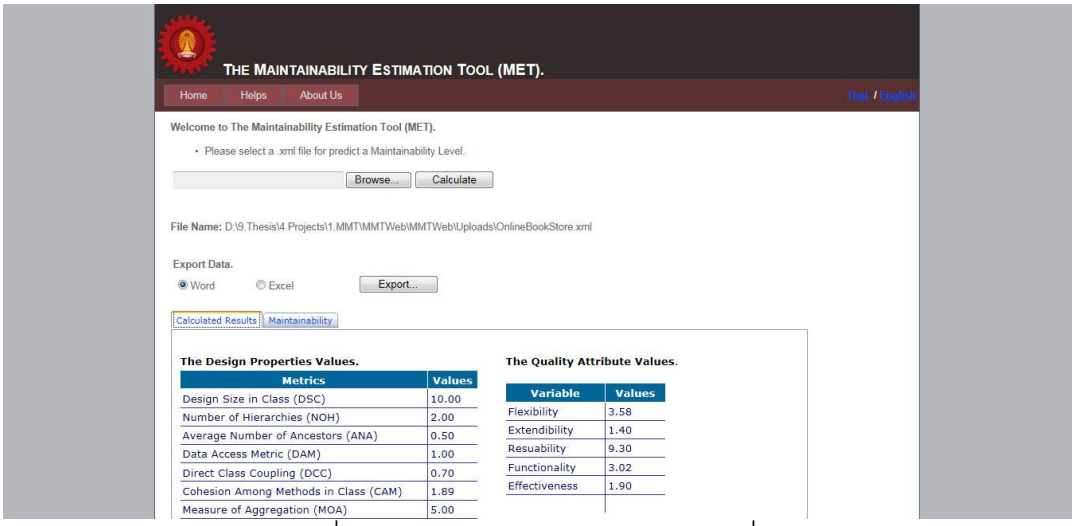
2) ส่วนการแสดงผลทางภาษาของเครื่องมือสามารถรองรับการใช้งานได้ 2 ภาษา ได้แก่ ภาษาไทย และภาษาอังกฤษ ดังรูปที่ 19 เป็นการแสดงผลภาษาไทย จากการเลือกเปลี่ยนภาษา



รูปที่ 19 การแสดงผลด้วยภาษาไทย

3) ส่วนการแสดงผลการคำนวณของเครื่องมือ

ในรูปที่ 20 เป็นส่วนแสดงผลการคำนวณของเครื่องมือ

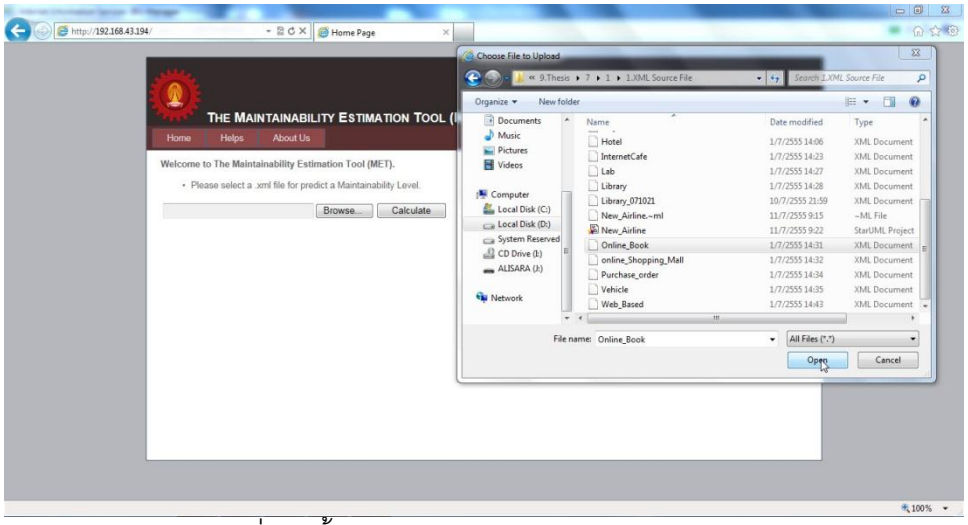


รูปที่ 20 ส่วนแสดงผลการคำนวณของเครื่องมือ

2.2 ขั้นตอนการใช้งาน

ในส่วนนี้จะแสดงขั้นตอนการใช้งานเครื่องมือโดยผู้ใช้งานจะต้องใช้โปรแกรมสตาร์ยูเอ็มแอลในการสร้างแผนภาพคลาส และใช้ในการแปลงแผนภาพคลาสให้อยู่ในรูปแบบของเอกสารเอ็กซ์เอ็มแอลเท่านั้น

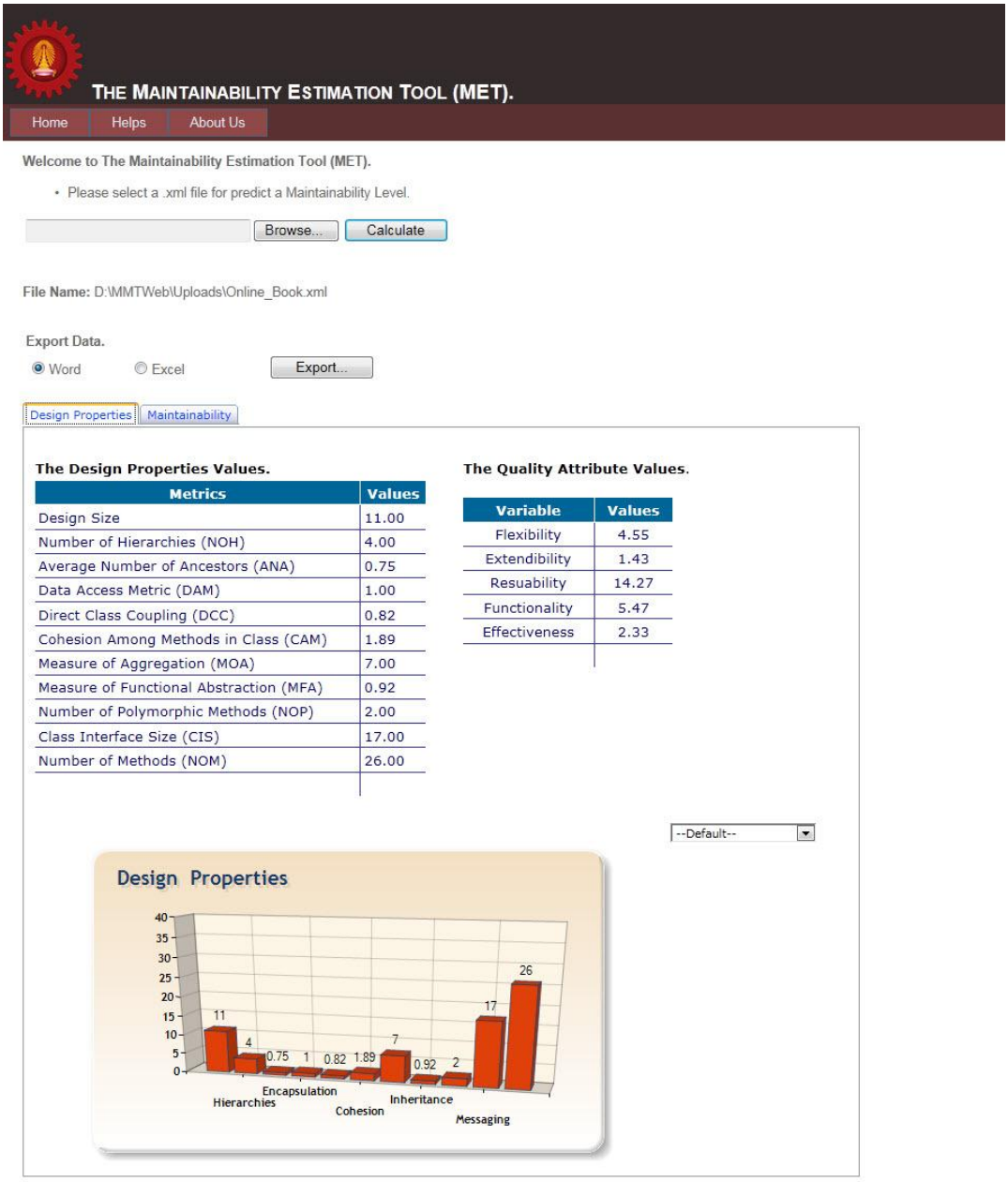
2.1.1 การเลือกเอกสารเอ็กซ์เอ็มแอลของแผนภาพคลาสเข้าสู่เครื่องมือ



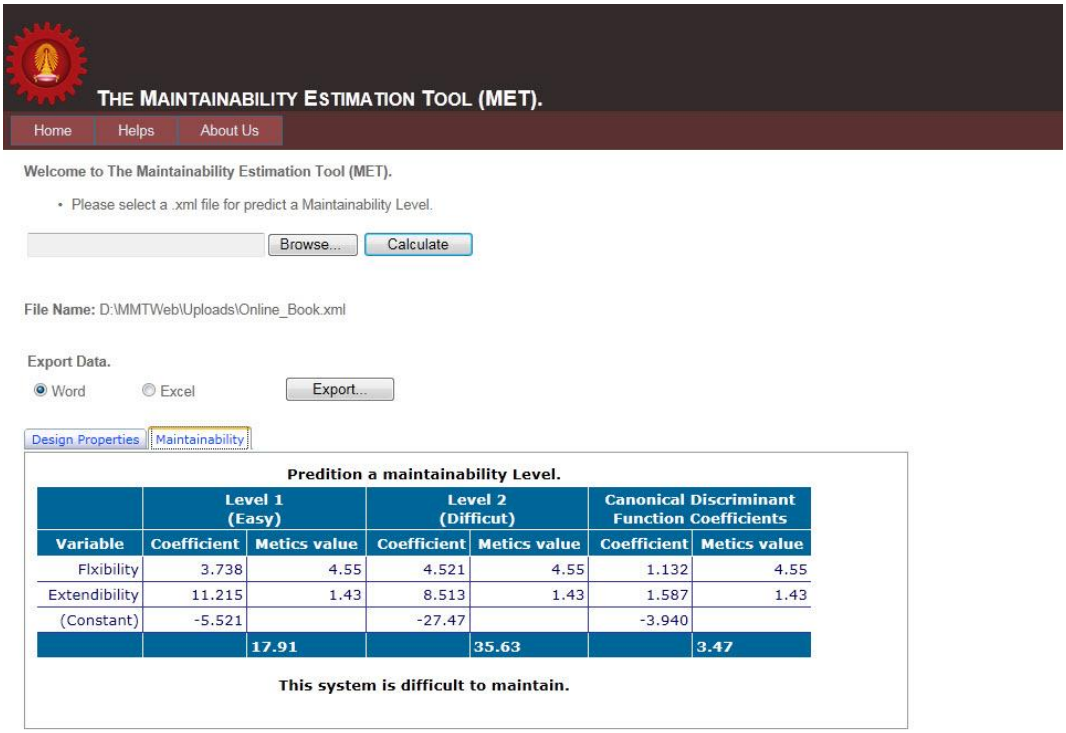
รูปที่ 21 ขั้นตอนการนำไฟล์เอกสารเอ็กซ์เอ็มแอล

มีขั้นตอนดังนี้

1) คลิก Browse เลือกโฟล์เดอร์ที่เก็บไฟล์เอกสารและเลือกไฟล์เอกสารเอ็กซ์เอ็มแอลของแผนภาพคลาส
2) คลิก Calculate ระบบจะแสดงผลจากการคำนวณ ดังรูปที่ 22 ส่วนแสดงการคำนวณด้วยมาตรวัดเชิงวัตถุและการคำนวณหาค่าคุณภาพของการออกแบบ



รูปที่ 22 ส่วนแสดงผลการคำนวณของเครื่องมือ

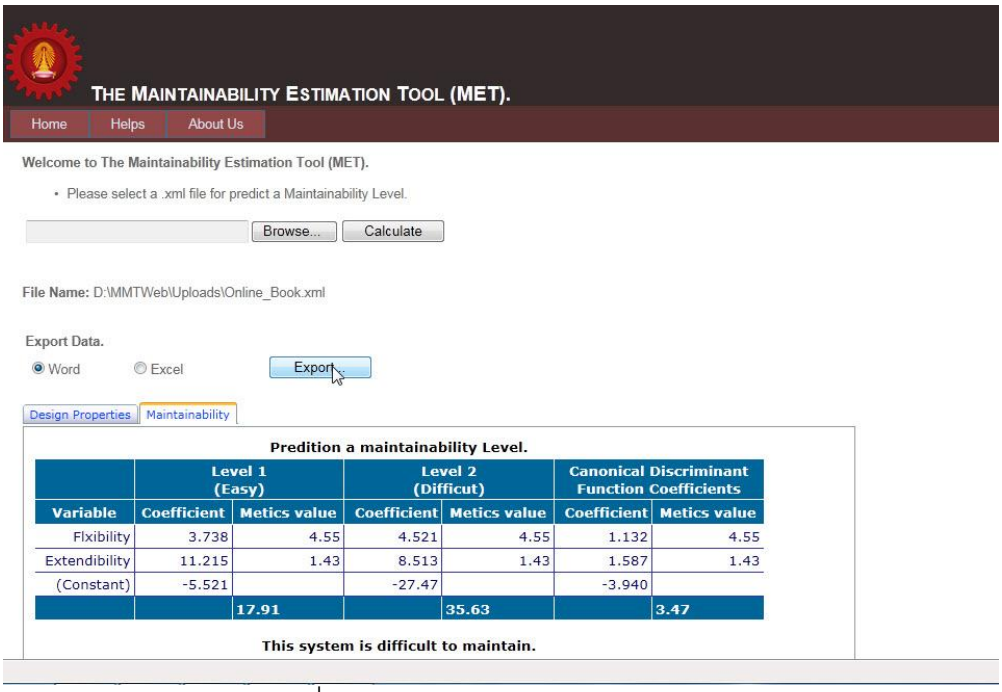


รูปที่ 23 ส่วนแสดงผลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์

3) หน้าจอการแสดงผลการคำนวณด้วยมาตรวัดเชิงวัตถุ การคำนวณคุณภาพของการออกแบบ และการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ตามรูปที่ 22 และรูปที่ 23 ส่วนแสดงผลการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์มีรายละเอียดดังนี้

- ส่วนการนำเอาข้อมูลออก (Export Data) สามารถนำเอาข้อมูลออกในรูปแบบเอกสารไฟล์ Word และ Excel โดยจะมีนามสกุลของไฟล์เป็น *.doc* และ *.xls* ตามลำดับ
- ส่วนการแสดงผลที่ได้จากการคำนวณ โดยแบ่งออกเป็น 2 ส่วน ได้แก่
    - *Design Properties* แสดงการคำนวณด้วยมาตรวัดทั้ง 11 มาตรวัด การคำนวณหาค่าคุณภาพของการออกแบบซอฟต์แวร์ และแสดงผลของมูลที่ได้จากการคำนวณด้วยมาตรวัดในรูปแบบของแผนภูมิแท่ง
    - *Maintainability* แสดงการคำนวณด้วยสมการจำแนกกลุ่มเพื่อเป็นการวัดความสามารถในการบำรุงรักษาซอฟต์แวร์ดังรูปที่ 23

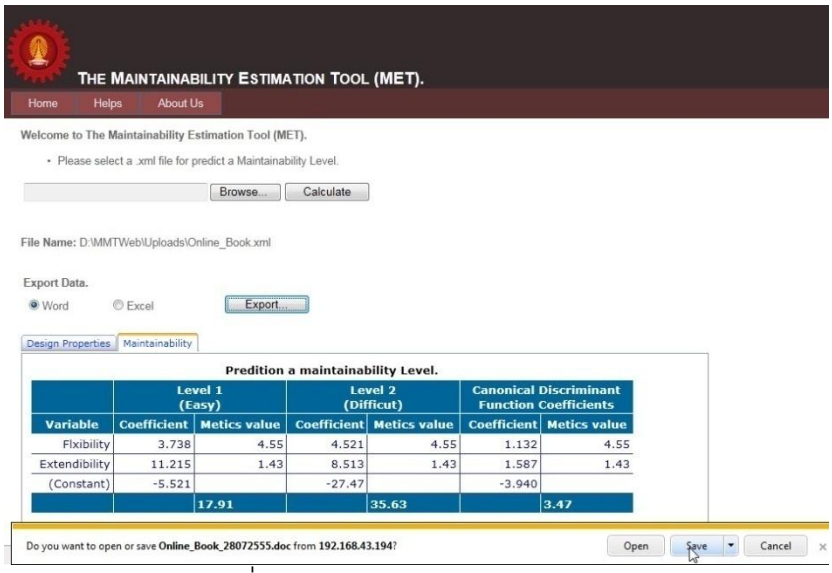
4) การนำเอาข้อมูลออก (Export Data) โดยเครื่องมือสามารถรองรับการ Export Data ได้ 2 รูปแบบเอกสาร ได้แก่ เอกสาร Word และเอกสาร Excel รูปที่ 24 แสดงหน้าจอการนำข้อมูลออก



รูปที่ 24 การนำข้อมูลออก (Export Data)

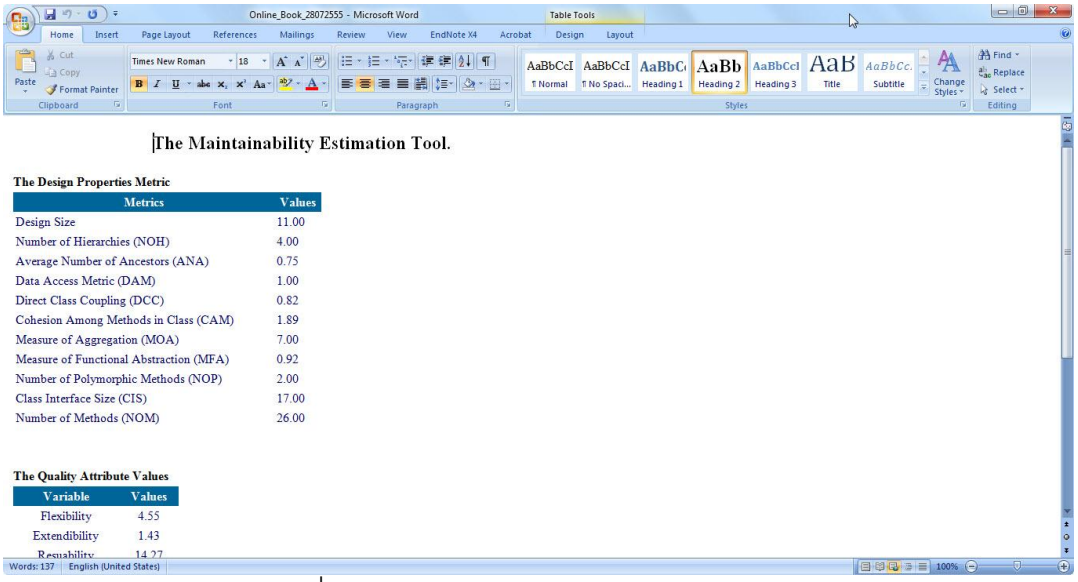
ขั้นตอนในการนำข้อมูลออก

1) เลือกรูปแบบไฟล์ที่ต้องการนำเอาข้อมูลออกเลือก Word และคลิก Export

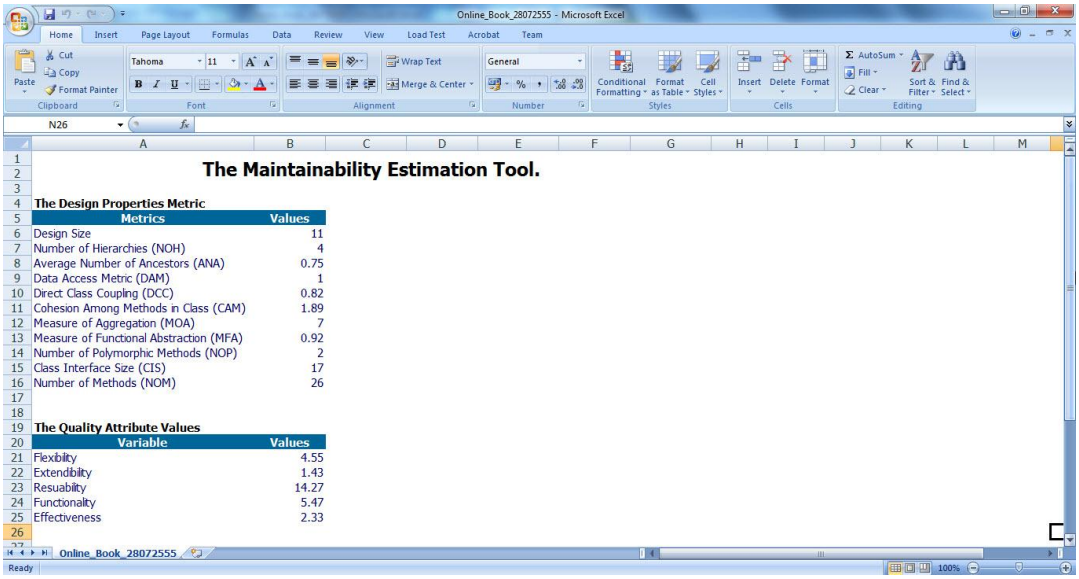


รูปที่ 25 แสดงหน้าจอการบันทึกข้อมูล

2) หน้าจอจะปรากฏให้บันทึกข้อมูลดังรูปที่ 25 เลือก Save และเปิดข้อมูลที่ทำการบันทึก จะแสดงผลข้อมูลในรูปแบบของเอกสาร Word ดังรูปที่ 26 และรูปที่ 27 เป็นการแสดงผลในรูปแบบเอกสาร Excel



รูปที่ 26 การแสดงผลของข้อมูลของเอกสาร Word



The Maintainability Estimation Tool.

The Design Properties Metric

Metrics	Values
Design Size	11
Number of Hierarchies (NOH)	4
Average Number of Ancestors (ANA)	0.75
Data Access Metric (DAM)	1
Direct Class Coupling (DCC)	0.82
Cohesion Among Methods in Class (CAM)	1.89
Measure of Aggregation (MOA)	7
Measure of Functional Abstraction (MFA)	0.92
Number of Polymorphic Methods (NOP)	2
Class Interface Size (CIS)	17
Number of Methods (NOM)	26

The Quality Attribute Values

Variable	Values
Flexibility	4.55
Extendibility	1.43
Resuability	14.27
Functionality	5.47
Effectiveness	2.33

รูปที่ 27 การแสดงผลของข้อมูลเอกสาร Excel

ประวัติผู้เขียนวิทยานิพนธ์

นางสาวอลิสรา หินชีระนันท์ เกิดวันที่ 3 มกราคม พ.ศ. 2527 ที่จังหวัดกาญจนบุรี สำเร็จการศึกษาหลักสูตรวิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศเพื่อการออกแบบ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยศิลปากร ในปีการศึกษา 2549 และเข้าศึกษาต่อในหลักสูตรวิทยาศาตรมหาบัณฑิต สาขาวิชาวิศวกรรมซอฟต์แวร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาตร์ จุฬาลงกรณ์มหาลัยวิทยาลัย ในปีการศึกษา 2552