การเปรียบเทียบวิธีการประมาณการค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น

นายวชิรา สุขสวัสดิ์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาการพัฒนาซอฟต์แวร์ด้านธุรกิจ ภาควิชาสถิติ
คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2556
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย
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 University Graduate School.

THE COMPARISON OF EFFORT ESTIMATION METHODS OF MOBILE APPLICATION DEVELOPMENT

Mr. Vajira Sukhasvasdi

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Business Software Development
Department of Statistics
Faculty of Commerce and Accountancy
Chulalongkorn University
Academic Year 2013

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การเปรียบเทียบวิธีการประมาณการค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น
โดย	นายวชิรา สุขสวัสดิ์
สาขาวิชา	การพัฒนาซอฟต์แวร์ด้านธุรกิจ
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร.ถาวร อานุภาพไตรรงค์

คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะพาณิชยศาสตร์และการบัญชี

(รองศาสตราจารย์ ดร.พสุ เดชะรินทร์)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.สมจารี ปรียานนท์)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(ผู้ช่วยศาสตราจารย์ ดร.ถาวร อานุภาพไตรรงค์)

.....กรรมการ

(รองศาสตราจารย์ ดร.อัษฎาพร ทรัพย์สมบูรณ์)

.....กรรมการภายนอกมหาวิทยาลัย

(ผู้ช่วยศาสตราจารย์ ดร.วรลักษณ์ วงศ์โดยหวัง ศิริเจริญ)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

วชิรา สุขสวัสดิ์ : การเปรียบเทียบวิธีการประมาณการค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น. (THE COMPARISON OF EFFORT ESTIMATION METHODS OF MOBILE APPLICATION DEVELOPMENT) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร.ถาวร อานุภาพไตรรงค์, 173 หน้า.

การประมาณค่าความพยายามในการพัฒนาซอฟต์แวร์เป็นหนึ่งกิจกรรมที่มีความสำคัญต่อความสำเร็จของโครงการ ดังนั้นการประมาณค่าความพยายามที่มีความแม่นยำจะช่วยให้ผู้จัดการโครงการสามารถรับมือกับปัจจัยทางด้านต้นทุนและแรงงาน ตลอดจนกำหนดการต่างๆ ของโครงการได้อย่างเหมาะสม

การพัฒนาโมบายแอพพลิเคชั่นเป็นการพัฒนาซอฟต์แวร์ประเภทหนึ่งที่กำลังได้รับความนิยมในปัจจุบัน ซึ่งโมบายแอพพลิเคชั่นคือ แอพพลิเคชั่นที่ทำงานอยู่บนระบบปฏิบัติการที่ติดตั้งลงในอุปกรณ์เคลื่อนที่ ซึ่งจะเน้นไปที่การตอบสนองการทำงานต่างๆ แก่ผู้ใช้ในชีวิตประจำวัน ได้แก่ การใช้งานอินเตอร์เน็ต อีเมล์ ถ่ายภาพ ซื้อสินค้าออนไลน์ หรือด้านความบันเทิงอื่นๆ ด้วยหน้าจอที่มีขนาดเล็กกว่าซอฟต์แวร์บนอุปกรณ์คอมพิวเตอร์ จากการทบทวนวรรณกรรมในอดีตพบว่ายังไม่มีการศึกษาถึงวิธีการประมาณการค่าความพยายามที่เหมาะสมสำหรับการพัฒนาโมบายแอพพลิเคชั่น การศึกษาครั้งนี้จึงทำการเปรียบเทียบวิธีการประมาณการค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีการสี่วิธี ได้แก่ วิธีฟังก์ชั่นพอยต์ วิธีอ็อบเจกต์พอยต์ วิธียูสเคสพอยต์ และวิธีของ Nikunj Sakhrelia นักพัฒนาโมบายแอพพลิเคชั่นอิสระ เป็นวิธีการที่พัฒนามาจากประสบการณ์ในการพัฒนาโมบายแอพพลิเคชั่นของเขาและได้นำเสนอไว้บนเว็บไซต์

ผลการวิเคราะห์ข้อมูลพบว่า การประมาณขนาดซอฟต์แวร์ด้วยวิธีของ Nikunj Sakhrelia มีความแม่นยำมากที่สุด โดยให้ผลความแม่นยำไม่แตกต่างกับวิธีฟังก์ชั่นพอยต์และวิธีอ็อบเจกต์พอยต์ อย่างมีนัยสำคัญ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาควิชา สถิติ ลายมือชื่อนิสิต .....

สาขาวิชา การพัฒนาซอฟต์แวร์ด้านธุรกิจ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก .....

ปีการศึกษา 2556

5481665426 : MAJOR BUSINESS SOFTWARE DEVELOPMENT
KEYWORDS: SOFTWARE DEVELOPMENT EFFORT / EFFORT ESTIMATION METHODS / MOBILE APPLICATION DEVELOPMENT / FUNCTION POINT ESTIMATION / OBJECT POINT ESTIMATION / USE CASE POINT ESTIMATION / PROJECT MANAGEMENT

VAJIRA SUKHASVASDI: THE COMPARISON OF EFFORT ESTIMATION METHODS OF MOBILE APPLICATION DEVELOPMENT. ADVISOR: ASST. PROF. THARWON ARNUPHAPTRAIRONG, Ph.D., 173 pp.

Effort estimation is an important activity to the success of the software development project. The effort estimation accuracy can help project managers deal with cost and resource limitation of the projects.

Mobile application development is a type of software development that is gaining popularity today. Mobile applications are application software designed to run on a mobile devices, such as a smartphone or tablet computer. Mobile applications frequently serve to provide users with similar services to those who accessed on PCs. Mobile applications are generally small and individual.

However, there is no research study on the mobile application development effort.This research compared effort estimation methods for mobile application based on four different methods, including function Point, Object point, Use case Point, and a method which presented by Nikunj Sakhrelia. -- an independent mobile application developer.

The findings show that Nikunj Sakhrelia's method is the most precise prediction method. In addition, the accuracy of Nikunj Sakhrelia's method show no statistical difference from Function Point and Object Point significantly.

Chulalongkorn University

Department:	Statistics	Student's Signature	.....
Field of Study:	Business Software Development	Advisor's Signature	.....
Academic Year:	2013		

กิตติกรรมประกาศ

ผู้วิจัยขอขอบพระคุณผู้ช่วยศาสตราจารย์ ดร. ถาวร อานุภาพไตรรงค์ อาจารย์ที่ปรึกษาวิทยานิพนธ์เป็นอย่างสูง ที่กรุณาสละเวลาในการให้คำปรึกษา ชี้แนะแนวทาง ตรวจสอบแก้ไขข้อบกพร่องต่างๆ รวมทั้งให้กำลังใจแก่ผู้วิจัยตลอดมา จนวิทยานิพนธ์ฉบับนี้เสร็จสมบูรณ์ และขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร. สมจารี ปรียานนท์ ประธานกรรมการวิทยานิพนธ์ รองศาสตราจารย์ ดร. อัษฎาพร ทรัพย์สมบูรณ์ กรรมการวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร.วรลักษณ์ วงศ์โดยหวัง ศิริเจริญ กรรมการภายนอกมหาวิทยาลัย ที่กรุณาสละเวลาอันมีค่า ให้คำแนะนำและชี้แนะสิ่งต่างๆ ตลอดจนอาจารย์ทุกท่านที่ได้มอบความรู้แก่ผู้วิจัย นอกจากนี้ยังขอขอบคุณไปยังอาจารย์ ดร.เดชานุชิต กตัญญูทวีทิพย์ ที่ให้คำปรึกษาและแนะนำแหล่งข้อมูลสำหรับการทำวิทยานิพนธ์ครั้งนี้ คุณสิทธิพล ลิ้มชัยชะดา และคุณโสภณา ถีติปริวัตร์ สำหรับข้อมูลและคำแนะนำเกี่ยวกับกระบวนการพัฒนาโมบายแอพพลิเคชั่น และแหล่งข้อมูลทุกท่านที่ได้เสียสละเวลาให้ข้อมูลแก่ผู้วิจัย

ที่สำคัญยิ่งขอขอบพระคุณพ่อ คุณแม่ และพี่ชาย ที่สนับสนุนทั้งกำลังใจและกำลังทรัพย์ในการศึกษาตลอดมา ขอบคุณเพื่อนกลุ่มมะนาวที่เป็นที่ปรึกษาที่ดี พี่ๆฝ่ายกำกับงานปฏิบัติการทุกคนที่คอยเป็นห่วงและให้กำลังใจเสมอ เพื่อนๆพี่ๆน้องๆทุกคนที่คอยช่วยเหลือและเป็นกำลังใจ จนวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงลงได้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญรูป ..... ญ

สารบัญตาราง ..... ฏ

บทที่ 1 บทนำ 1

1.1 ที่มาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ของการวิจัย 5

1.3 ขอบเขตของการวิจัย 5

1.4 ขั้นตอนและวิธีการดำเนินการวิจัย 6

1.5 ประโยชน์ที่คาดว่าจะได้รับ 6

บทที่ 2 ทฤษฎีและวรรณกรรมที่เกี่ยวข้อง 7

2.1 การประมาณขนาดซอฟต์แวร์ด้วยการวิเคราะห์ฟังก์ชั่นพอยต์ (Function Point Analysis : FPA) 7

2.1.1. คำนวณหาค่าฟังก์ชั่นพอยต์ที่ยังไม่ถูกปรับค่า (UFP) 8

2.1.2. คำนวณตัวแปรปรับค่า (Value Adjustment Factor: VAF) 14

2.1.3. คำนวณค่าฟังก์ชั่นพอยต์ที่ปรับค่าแล้ว 24

2.2 การประมาณขนาดซอฟต์แวร์ด้วยวิธีอ็อบเจกต์พอยต์ (Object Points) 24

2.2.1. การหาจำนวนอ็อบเจกต์ 25

2.2.2. การจัดกลุ่มให้กับอ็อบเจกต์ 25

2.2.3. การให้ค่าน้ำหนักแก่อ็อบเจกต์ที่แบ่งกลุ่มแล้ว 26

2.3 การประมาณขนาดซอฟต์แวร์ด้วยวิธียูสเคสพอยต์ (Use Case Point : UCP) 28

2.3.1. ค่าน้ำหนักยูสเคสที่ยังไม่ปรับค่า (UUCW) 28

2.3.2. ค่าน้ำหนักผู้กระทำการที่ยังไม่ปรับค่า (UAW) 29

2.3.3. ค่าปัจจัยความซับซ้อนเชิงเทคนิค (TCF) 30

2.3.4. ค่าปัจจัยความซับซ้อนทางสภาพแวดล้อม (ECF) 31

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

หน้า

2.3.5. ค่ายูสเคสพอยต์ (UCP) 35
2.4 วิธีการประมาณการค่าความพยายามที่นำเสนอโดย Nikunj Sakhrelia นักพัฒนาโมบายแอพพลิเคชั่นอิสระ 36
2.5 การถอด(Extract)ฟังก์ชั่นพอยต์และยูสเคสพอยต์จากข้อมูลความต้องการในรูปแบบการเขียนเป้าหมาย (Goal) และสถานการณ์ (Scenario) บนพื้นฐานของความต้องการแบบข้อความ (Text Requirement) 38
2.6 ลักษณะและความท้าทายในการพัฒนาโมบายแอพพลิเคชั่น 53
บทที่ 3 ระเบียบวิธีวิจัย 55
3.1 การดำเนินงานวิจัย 55
3.2 ตัวแปรที่ทำการศึกษา 98
3.3 สมมติฐานการวิจัย 99
3.4 ความถูกต้องและความน่าเชื่อถือของข้อมูล 100
3.5 กรอบการวิเคราะห์ข้อมูล 101
บทที่ 4 ผลการวิเคราะห์ข้อมูล 102
4.1 ผลการเก็บรวบรวมข้อมูล 102
4.2 วิธีการคำนวณขนาดและค่าความพยายาม 104
4.3 การทดสอบสมมติฐาน 107
4.3.1. การทดสอบการแจกแจงปกติของข้อมูล 107
4.3.2. เปรียบเทียบค่าความพยายามจากวิธีการที่แตกต่างกัน 110
4.4 ความแม่นยำของวิธีการประมาณค่าความพยายามทั้ง 4 วิธี 112
4.4.1. ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ 114
4.4.2. ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ 117
4.4.3. ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ 120
4.4.4. ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia 124
4.5 การเปรียบเทียบความแม่นยำของวิธีการประมาณการค่าความพยายามทั้ง 4 วิธี 127

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

4.6 สรุปผลการวิเคราะห์ข้อมูล 130
บทที่ 5 สรุปผลการวิจัย 131
5.1 สรุปผลการวิจัย 131
5.2 อภิปรายผล 131
5.3 การนำงานวิจัยไปใช้ 133
รายการอ้างอิง 134
ภาคผนวก ก 139
ภาคผนวก ข 153
ประวัติผู้เขียนวิทยานิพนธ์ 173

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญรูป

หน้า

รูปที่ 2.1 แสดงโมเดลของการวิเคราะห์ฟังก์ชั่นพอยต์ 8
รูปที่ 2.2 โครงสร้างของสถานการณ์(Scenario) 40
รูปที่ 2.3 แสดงผังการไหลของเป้าหมายในระดับต่างๆของบริการเอทีเอ็ม 45
รูปที่ 2.4 แสดงขอบเขตของระบบประมวลผลคำสั่งซื้อ 46
รูปที่ 2.5 แสดงสถานการณ์ตัวอย่างสำหรับพิจารณาฟังก์ชั่นเชิงข้อมูล 47
รูปที่ 2.6 แสดงความสัมพันธ์ระหว่างเป้าหมายและองค์ประกอบอื่น 50
รูปที่ 2.7 แสดงแผนภาพยูสเคสสำหรับการให้บริการเอทีเอ็ม โดยใช้การแปลงตามกฎทั้ง 4 ข้อ 52
รูปที่ 3.1 แสดงขั้นตอนการดำเนินงานวิจัย 57
รูปที่ 3.2 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นส่วนที่ 1 60
รูปที่ 3.3 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นส่วนที่ 2 62
รูปที่ 3.4 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นส่วนที่ 3 67
รูปที่ 3.5 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 1 ของโมบายแอพพลิเคชั่นตัวอย่าง 71
รูปที่ 3.6 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 2 ของโมบายแอพพลิเคชั่นตัวอย่าง 72
รูปที่ 3.7 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 2 ของโมบายแอพพลิเคชั่นตัวอย่าง (ต่อ) 73
รูปที่ 3.8 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 3 ของโมบายแอพพลิเคชั่นตัวอย่าง 74
รูปที่ 3.9 แสดงหน้าจอหลักและหน้าจอกล้อง ของโมบายแอพพลิเคชั่นตัวอย่าง 75
รูปที่ 3.10 แสดงหน้าจอผลการสแกนและหน้าจอ History ของโมบายแอพพลิเคชั่นตัวอย่าง 75
รูปที่ 3.11 แสดงการระบุฟังก์ชั่นเชิงข้อมูลของโมบายแอพพลิเคชั่นตัวอย่าง 76
รูปที่ 3.12 แสดงการระบุฟังก์ชั่นเชิงรายการของโมบายแอพพลิเคชั่นตัวอย่าง 77
รูปที่ 3.13 แสดงการนับ DET RET ของโมบายแอพพลิเคชั่นตัวอย่าง 78

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

รูปที่ 3.14 แสดงการกรอกข้อมูลลงในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่น ด้วยวิธีฟังก์ชั่นพอยต์ 81
รูปที่ 3.15 แสดงหน้าจอทั้งหมดของโมบายแอพพลิเคชั่นตัวอย่าง 82
รูปที่ 3.16 แสดงการนับส่วนของหน้าจอของโมบายแอพพลิเคชั่นตัวอย่าง 83
รูปที่ 3.17 แสดงการนับจำนวนแหล่งข้อมูลข้อมูลบนหน้าจอของโมบายแอพพลิเคชั่นตัวอย่าง 84
รูปที่ 3.18 แสดงการนับโมดูลของโมบายแอพพลิเคชั่นตัวอย่าง 84
รูปที่ 3.19 แสดงการกรอกข้อมูลในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่น ด้วยวิธีอ็อบเจกต์พอยต์ 86
รูปที่ 3.20 แสดงการแสดงการระบุยูสเคสและผู้กระทำการของโมบายแอพพลิเคชั่นตัวอย่าง 87
รูปที่ 3.21 แสดงแผนภาพยูสเคสของโมบายแอพพลิเคชั่นตัวอย่าง 88
รูปที่ 3.22 แสดงการนับจำนวนรายการของโมบายแอพพลิเคชั่นตัวอย่าง 88
รูปที่ 3.23 แสดงการกรอกข้อมูลในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วย วิธียูสเคสพอยต์ 90
รูปที่ 3.26 แสดงการนับองค์ประกอบหน้าจอของโมบายแอพพลิเคชั่นตัวอย่าง 93
รูปที่ 3.27 แสดงการพิจารณาขั้นตอนการทำงานบนหน้าจอ History ของ โมบายแอพพลิเคชั่นตัวอย่าง 93

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดงระดับความซับซ้อนของ ILFs และ EIFs 10
ตารางที่ 2.2 แสดงระดับความซับซ้อนของ EI 11
ตารางที่ 2.3 แสดงระดับความซับซ้อนของ EO และ EQ 13
ตารางที่ 2.4 แสดงการนับค่าฟังก์ชั่นพอยต 13
ตารางที่ 2.5 แสดงการกำหนดระดับปัจจัย Data Communication 15
ตารางที่ 2.6 แสดงการกำหนดระดับปัจจัย Distributed Data Processing 15
ตารางที่ 2.7 แสดงการกำหนดระดับปัจจัย Performance 16
ตารางที่ 2.8 แสดงการกำหนดระดับปัจจัย Heavily Used Configuration 17
ตารางที่ 2.9 แสดงการกำหนดระดับปัจจัย Transaction Rate 17
ตารางที่ 2.10 แสดงการกำหนดระดับปัจจัย On-Line Data Entry 18
ตารางที่ 2.11 แสดงการกำหนดระดับปัจจัย End-User Efficiency 18
ตารางที่ 2.12 แสดงการกำหนดระดับปัจจัย On-Line Update 19
ตารางที่ 2.13 แสดงการกำหนดระดับปัจจัย Complex Processing 20
ตารางที่ 2.14 แสดงการกำหนดระดับปัจจัย Reusability 20
ตารางที่ 2.15 แสดงการกำหนดระดับปัจจัย Installation Ease 21
ตารางที่ 2.16 แสดงการกำหนดระดับปัจจัย Operation Ease 21
ตารางที่ 2.17 แสดงการกำหนดระดับปัจจัย Multiple Sites 22
ตารางที่ 2.18 แสดงการกำหนดระดับปัจจัย Facilitate change 22
ตารางที่ 2.19 แสดงการคำนวณปัจจัยคุณลักษณะทั่วไปของระบบ 23
ตารางที่ 2.20 การแบ่งกลุ่มอ็อบเจกต์ 26
ตารางที่ 2.21 แสดงค่าความน้ำหนักของวิธีอ็อบเจกต์พอยต์ 26
ตารางที่ 2.22 แสดงตัวอย่างการคำนวณค่าอ็อบเจกต์พอยต์ 27
ตารางที่ 2.23 แสดงค่าถ่วงน้ำหนักของยูสเคส ซึ่งพิจารณาจากจำนวนรายการ 29
ตารางที่ 2.24 แสดงค่าถ่วงน้ำหนักของผู้กระทำการ 29

ตารางที่ 2.25 แสดงค่าถ่วงน้ำหนักของปัจจัยที่มีผลต่อความซับซ้อนทางด้านเทคนิค..... 30
ตารางที่ 2.26 แสดงการกำหนดระดับปัจจัย Familiar with development method used 31
ตารางที่ 2.27 แสดงการกำหนดระดับปัจจัย Application experience..... 32
ตารางที่ 2.28 แสดงการกำหนดระดับปัจจัย Object-Oriented experience..... 32
ตารางที่ 2.29 แสดงการกำหนดระดับปัจจัย Lead analyst capability..... 33
ตารางที่ 2.30 แสดงการกำหนดระดับปัจจัย Motivation 33
ตารางที่ 2.31 แสดงการกำหนดระดับปัจจัย Stable requirements 33
ตารางที่ 2.32 แสดงการกำหนดระดับปัจจัย Part-time workers 34
ตารางที่ 2.33 แสดงการกำหนดระดับปัจจัย Difficult programming language..... 34
ตารางที่ 2.34 แสดงค่าถ่วงน้ำหนักของผลกระทบทางสภาพแวดล้อมของการพัฒนา..... 35
ตารางที่ 2.35 แสดงการแบ่งฟังก์ชั่นการทำงานและองค์ประกอบบนหน้าจอตามวิธีของ Nikunj Sakhrelia 37
ตารางที่ 2.36 ตัวอย่างสถานการณ์ของบริการ เอทีเอ็ม ที่เขียนโดยใช้กฎข้อ S1-S6..... 44
ตารางที่ 3.1 อธิบายตัวเลือกในการกรอกข้อมูลช่อง Subject 63
ตารางที่ 3.2 อธิบายตัวเลือกในการกรอกข้อมูลช่อง Verb..... 64
ตารางที่ 3.3 อธิบายตัวเลือกในการกรอกข้อมูลช่อง Use Type..... 66
ตารางที่ 3.4 อธิบายตัวเลือกในการกรอกข้อมูลช่อง Module Type..... 66
ตารางที่ 3.5 แสดงฟิลด์ข้อมูลในแบบฟอร์มการเก็บข้อมูลที่ใช้ในการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์และวิธีอ็อบเจกต์พอยต์..... 69
ตารางที่ 3.6 แสดงฟิลด์ข้อมูลในแบบฟอร์มการเก็บข้อมูลที่ใช้ในการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์และวิธีของ Nikunj Sakhrelia..... 70
รูปที่ 3.10 แสดงหน้าจอผลการสแกนและหน้าจอ History ของโมบายแอพพลิเคชั่นตัวอย่าง..... 75
ตารางที่ 3.6 แสดงระดับความซับซ้อนของฟังก์ชั่นเชิงข้อมูล..... 79
ตารางที่ 3.7 ตารางแสดงระดับความซับซ้อนของฟังก์ชั่นเชิงรายการ EI 79
ตารางที่ 3.8 ตารางแสดงระดับความซับซ้อนของฟังก์ชั่นเชิงรายการ EO และ EQ..... 79
ตารางที่ 3.9 การแบ่งกลุ่มอ็อบเจกต์..... 85

ตารางที่ 3.10 แสดงค่าถ่วงน้ำหนักของยูสเคส 89
ตารางที่ 3.11 แสดงค่าถ่วงน้ำหนักของผู้กระทำการ 89
ตารางที่ 3.12 แสดงการแบ่งฟังก์ชั่นการทำงานและองค์ประกอบบนหน้าจอตามวิธีของ Nikunj Sakhrelia 94
ตารางที่ 3.13 แสดงค่าความสามารถในการพัฒนาซอฟต์แวร์ 97
ตาราง 4.1 แสดงผลการเก็บข้อมูลโมบายแอพพลิเคชั่น 103
ตาราง 4.2 แสดงผลการคำนวณขนาดโมบายแอพพลิเคชั่น 105
ตาราง 4.3 แสดงผลการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น 106
ตารางที่ 4.4 ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ 108
ตารางที่ 4.5 ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ 108
ตารางที่ 4.6 ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ 109
ตารางที่ 4.7 ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia 110
ตาราง 4.8 ผลการทดสอบ Kruskal –Wallis ของค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นทั้ง 4 วิธี จากข้อมูลทั้งหมด 111
ตาราง 4.9 ผลการเปรียบเทียบค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น สำหรับคู่ที่ข้อมูลมีการแจงแจกปกติ 112
ตาราง 4.10 ผลการเปรียบเทียบค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น สำหรับคู่ที่ข้อมูลไม่มีการแจงแจกปกติ 112
ตารางที่ 4.11 แสดงผลการคำนวณค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีฟังก์ชั่นพอยต์ 115
ตาราง 4.12 แสดงผลการทดสอบการแจกแจงข้อมูลของวิธีฟังก์ชั่นพอยต์ 116
ตาราง 4.13 ผลการทดสอบความแม่นยำในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ 117

CHULALONGKORN UNIVERSITY

ตารางที่ 4.14 แสดงผลการคำนวณค่า MRE ของวิธีอ็อบเจกต์พอยต์ 118
ตาราง 4.15 แสดงผลการทดสอบการแจกแจงข้อมูลของวิธีอ็อบเจกต์พอยต์ 119
ตาราง 4.16 ผลการทดสอบความแม่นยำในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ 120
ตารางที่ 4.17 แสดงผลการคำนวณค่า MRE ของวิธียูสเคสพอยต์ 122
ตาราง 4.18 แสดงผลการทดสอบการแจกแจงข้อมูลของวิธียูสเคสพอยต์ 123
ตาราง 4.19 ผลการทดสอบความแม่นยำในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ 124
ตารางที่ 4.20 แสดงผลการคำนวณค่า MRE ของวิธีของ Nikunj Sakhrelia 125
ตาราง 4.21 แสดงผลการทดสอบการแจกแจงข้อมูลของวิธีของ Nikunj Sakhrelia 126
ตาราง 4.22 ผลการทดสอบความแม่นยำในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia 127
ตาราง 4.23 แสดงผลการเปรียบเทียบความแม่นยำของวิธีการประมาณค่าความพยายามทั้ง 4 วิธี 128
ตาราง 4.24 แสดงผลการทดสอบความแม่นยำของวิธีการประมาณค่าความพยายามรายคู่ 129
ตาราง 4.25 สรุปผลการเปรียบเทียบวิธีการประมาณค่าความพยายามทั้ง 4 วิธี 130

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ในยุคปัจจุบันระบบคอมพิวเตอร์ได้ถูกผนวกรวมเข้าไปในอุปกรณ์ที่เรียกว่า สมาร์ตโฟน (Smart Phone) ซึ่งเป็นโทรศัพท์ที่สามารถติดตั้งระบบปฏิบัติการรวมถึงซอฟต์แวร์ เพื่อทำงานต่างๆ ได้เช่นเดียวกับคอมพิวเตอร์ เพื่อตอบสนองความต้องการใช้งานที่มากขึ้น โดยอุปกรณ์สมาร์ตโฟนนั้นเป็นหนึ่งใน อุปกรณ์จำพวกอุปกรณ์เคลื่อนที่ (Mobile Device) ที่ได้ถูกพัฒนาขึ้นมาเพื่อตอบสนองความต้องการใช้งานนอกสถานที่ เพราะมีจุดเด่น คือ มีน้ำหนักเบา ขนาดเล็ก จึงพกพาได้สะดวก (สุชาดา พลาชัยภิรมย์ศิล, 2554) ทั้งนี้สมาร์ตโฟนไม่เพียงแต่ใช้สำหรับสนทนาด้วยเสียงเท่านั้น ยังสามารถสนทนาผ่านวิดีโอ เชื่อมต่ออินเตอร์เน็ทเพื่อรับข้อมูลข่าวสาร เล่นเกม ฟังเพลง ตกแต่งรูปภาพ และการใช้งานอย่างอื่นอีกมากมาย การใช้งานที่หลากหลายนี้เกิดจากการติดตั้ง โมบายแอพพลิเคชั่น (Mobile Application) ที่หมายถึง ชุดของโปรแกรมหรือซอฟต์แวร์ที่สามารถติดตั้งและใช้งานได้ผ่านอุปกรณ์เคลื่อนที่ (Zheng et al. , 2010) โดยแบ่งประเภทของโมบายแอพพลิเคชั่นออกได้เป็น 21 ประเภท ดังนี้ 1) Book 2) Business 3) Education 4) Entertainment 5) Finance 6) Games 7) Health & Fitness 8) Lifestyle 9) Medical 10) Music 11) Navigation 12) News 13) Newsstand 14) Photo & Video 15) Productivity 16) Reference 17) Social Network 18) Sports 19) Travel 20) Utilities 21) Weather (Google Developer Groups , 2014)

ด้วยความหลากหลายของโมบายแอพพลิเคชั่นจึงทำให้ สมาร์ตโฟน กลายเป็นอุปกรณ์ที่ได้รับความนิยมอย่างมากในปัจจุบัน ตามแผนยุทธศาสตร์การส่งเสริมอุตสาหกรรมซอฟต์แวร์ พ.ศ.2555-2558 ฉบับปรับปรุงปี พ.ศ.2556 ได้นำเสนอข้อมูลที่น่าสนใจเกี่ยวกับมูลค่าการผลิตโมบายแอพพลิเคชั่นในไทยเปรียบเทียบระหว่างปี พ.ศ.2554 กับ พ.ศ.2555 พบว่ามีการเติบโตถึงร้อยละ 35.9 หรือมีมูลค่าการผลิตสูงขึ้นจาก 1,065 ล้านบาทเป็น 1,447 ล้านบาท (สำนักงานส่งเสริมอุตสาหกรรมซอฟต์แวร์แห่งชาติ , 2556) สอดคล้องกับข้อมูลการเติบโตของตลาดโมบายแอพพลิเคชั่นทั่วโลก ตามรายงานการวิจัยของพอร์ติโอ้ รีเสิร์ช พบว่าในปี ค.ศ. 2012 มูลค่าการดาวน์โหลดโมบายแอพพลิเคชั่นทั่วโลกมีมูลค่า 12,000 ล้านดอลลาร์สหรัฐ นับจำนวนการดาวน์โหลดเป็น 46,000 ล้านครั้ง สิ่งที่น่าสนใจและสะท้อนถึงการเติบโตอย่างรวดเร็วของโมบายแอพพลิเคชั่นคือ ยอดจำนวนการดาวน์โหลดสะสมจนถึงสิ้นปี ค.ศ. 2011 อยู่ที่ 37,000 ล้านครั้ง แต่เพิ่มเป็น 83,000 ล้านครั้งในสิ้นปี ค.ศ. 2012 ซึ่งคาดการณ์ว่าจะเติบโตเพิ่มมากขึ้นในอนาคต (Portio Research , 2013)

ร้านค้าออนไลน์ของโมบายแอพพลิเคชั่นจะขึ้นอยู่กับระบบปฏิบัติการที่ใช้งานในอุปกรณ์เคลื่อนที่ของบริษัทต่างๆ ไม่ว่าจะเป็น (1) Amazon Appstore ร้านค้าออนไลน์ของโมบาย

แอพพลิเคชั่นที่ใช้งานกับ Mobile Device ของบริษัท Amazon ได้แก่ เครื่อง Kindle fire (2) Apple App Store ร้านค้าออนไลน์ของโมบายแอพพลิเคชั่นที่ใช้งานกับ Mobile Device ของบริษัท Apple Inc. ที่รองรับระบบปฏิบัติการ iOS ได้แก่ iPhone iPod touch iPad (3) BlackBerry App World ร้านค้าออนไลน์ของโมบายแอพพลิเคชั่นที่ใช้งานกับ Mobile Device ของบริษัท RIM ที่รองรับระบบปฏิบัติการ Blackberry OS (4) Google Android Market ร้านค้าออนไลน์ของโมบายแอพพลิเคชั่นที่ใช้งานกับ Mobile Device ที่รองรับระบบปฏิบัติการ Android ซึ่งมีด้วยกันหลายรุ่นหลายยี่ห้อ ได้แก่ Samsung LG HTC (ปัจจุบันได้เปลี่ยนชื่อเป็น Google Play Store) (5) Nokia Ovi Store ร้านค้าออนไลน์ของโมบายแอพพลิเคชั่นที่ใช้งานกับ Mobile Device ของบริษัท Nokia ที่ใช้ระบบปฏิบัติการ Symbian OS (6) Windows Phone Appstore ร้านค้าออนไลน์ของโมบายแอพพลิเคชั่นล่าสุดของบริษัท Nokia สำหรับ Mobile Device ที่ใช้ระบบปฏิบัติการ Windows Phone ซึ่งถูกพัฒนาขึ้นโดย บริษัท ไมโครซอฟต์ เป็นต้น แต่จากทั้งหมดมีเพียงแค่ 2 รายเท่านั้นที่มีการเติบโตสูงขึ้นเรื่อยๆคือ Apple App Store และ Google Android Market (Holzinger, Treitler, and Slany,2012);(Tracy,2012)

การใช้งานที่มากขึ้นได้สอดคล้องกับสถิติการขายสมาร์ตโฟนทั่วโลก จาก Gartner และไอดีซี (IDC) บริษัทวิจัยทางเทคโนโลยีสารสนเทศ ที่ได้นำเสนอข้อมูลการเติบโตของตลาดสมาร์ตโฟนในไตรมาสแรกของปี ค.ศ. 2013 ที่มีการเปลี่ยนแปลงมากขึ้นถึงร้อยละ 41.6 จากไตรมาสเดียวกันของปี ค.ศ. 2012 ทำให้เห็นว่านอกจากแอพพลิเคชั่นแล้วพฤติกรรมการใช้งานของผู้ใช้หันมาบริการผ่านอุปกรณ์เคลื่อนที่มากขึ้น ดังนั้นตลาดโมบายแอพพลิเคชั่นจึงเป็นตลาดที่น่าสนใจมากสำหรับนักพัฒนาที่จะนำเสนอแอพพลิเคชั่นใหม่ๆออกมาเพื่อตอบสนองการใช้งานของผู้ใช้ (Gartner,2013);(Framingham,2013)

ความสำเร็จของอุปกรณ์เคลื่อนที่และโมบายแอพพลิเคชั่นนั้นอาจมาจากรูปแบบของการใช้งาน การควบคุม หรือที่เรียกว่าปฏิสัมพันธ์ของผู้ใช้งานกับอุปกรณ์ ซึ่งเทคโนโลยีจอสัมผัส (Touch Screen) ทำให้การใช้งานง่ายและสนุก ซึ่งอาจเป็นเหตุผลที่ทำให้อุปกรณ์เคลื่อนที่ในปัจจุบันประสบความสำเร็จและเป็นที่นิยมมากกว่า อุปกรณ์ยุคก่อนที่ยังมีการควบคุมการใช้งานด้วยปุ่ม (วงหทัย ตันชีวะวงศ์,2557)

เนื่องจากความนิยมที่เกิดขึ้นในปัจจุบันองค์กรภาคธุรกิจต่างๆ ให้ความสนใจ ที่จะมีแอพพลิเคชั่นเป็นของตัวเองเพื่อเสริมสร้างกลยุทธ์ทางธุรกิจ บ้างเพื่อให้ขายสินค้าได้ง่ายและสะดวกขึ้น บ้างเพื่อให้เกิดความรวดเร็วในการส่งข่าวสารถึงลูกค้า บางรายเพื่อต้องการส่งเสริมทางการตลาด เพราะการที่ลูกค้าติดตั้งแอพพลิเคชั่นขององค์กรก็คือการนำตราสินค้าขององค์กรติดตัวไปในทุกๆที่ ซึ่งไม่ว่าจะเพื่อการตอบสนองในด้านใดองค์กรต่างๆ ก็จะต้องเสียค่าใช้จ่ายสำหรับการพัฒนาไม่ว่าจะ

เป็นการพัฒนาขึ้นเองโดยฝ่ายงานขององค์กรเอง หรือเป็นการว่าจ้างบริษัทภายนอก หรือนักพัฒนาโมบายแอพพลิเคชั่นอิสระ จะต้องประมาณการต้นทุนค่าแอพพลิเคชั่นที่จะเกิดขึ้นว่าจะคุ้มค่ากับต้นทุนที่จะเสียไปในการพัฒนาหรือไม่ และในทางกลับกันก็เป็นประโยชน์ต่อด้านผู้รับพัฒนาเองเช่นกัน

การประมาณขนาดของซอฟต์แวร์ (Software Size Estimation) เป็นสิ่งสำคัญ ในการบริหารโครงการซอฟต์แวร์ เพื่อหาขนาดของซอฟต์แวร์ในการวางแผนโครงการซอฟต์แวร์ เพราะขนาดของซอฟต์แวร์ (Software Size) เป็นตัวแปรที่ใช้ในการประมาณค่าความพยายามที่ใช้ในการพัฒนาซอฟต์แวร์ (Development Effort) ซึ่งก็คือต้นทุนของซอฟต์แวร์ (Cost) การประมาณการต้นทุนซอฟต์แวร์เป็นสิ่งที่สำคัญในการดำเนินโครงการอย่างมากเนื่องจากพบว่า การประเมินระยะเวลา ต้นทุน และค่าความพยายาม เป็นปัจจัยที่ส่งผลกระทบกับโครงการพัฒนาซอฟต์แวร์อย่างมาก ในปี ค.ศ. 2014 Jorgensen ได้ศึกษาถึงปัจจัยความล้มเหลวของโครงการซอฟต์แวร์ขนาดเล็ก ทั้งนี้โครงการซอฟต์แวร์ขนาดเล็กในการศึกษาของ Jorgensen หมายถึง โครงการพัฒนาซอฟต์แวร์ที่มีราคาต่ำกว่า 1,000 ดอลล่าร์สหรัฐ ซึ่งพบว่าปัจจัยทางด้านการประมาณการต้นทุนและการวางแผนโครงการเป็นปัจจัยที่สำคัญ โดยกล่าวว่าในการแข่งขันของตลาดซอฟต์แวร์นั้นผู้ว่าจ้างมัก พิจารณาเลือกผู้ให้บริการในการพัฒนาซอฟต์แวร์ที่ให้ราคาต่ำสุด นั่นสะท้อนว่าหากผู้พัฒนาซอฟต์แวร์คำนึงถึงการนำเสนอราคาที่ต่ำโดยไม่สนใจค่าความพยายามและต้นทุนที่ควรจะเป็นอาจจะส่งผลให้มีโอกาสสูงที่โครงการพัฒนาซอฟต์แวร์นั้นจะล้มเหลว(Jorgensen,2014) ซึ่งสอดคล้องกับการศึกษาปัจจัยความล้มเหลวของโครงการซอฟต์แวร์ขนาดใหญ่ปีค.ศ. 1999 Linberg พบว่าร้อยละ 20-ของโครงการซอฟต์แวร์ต้องที่ล้มเหลว จากทั้งหมดมีร้อยละ 46-เกิดจากปัญหาต้นทุน และตารางเวลาที่เกินกำหนด (Linberg, 1999) ในปีค.ศ. 2007 รายงานของเดอะ สแตนดิส กรุ๊ป ยังคงพบปัญหาต้นทุนและระยะเวลาในการพัฒนาซอฟต์แวร์ที่เกินกำหนด โดยพบว่าร้อยละ 46 ของโครงการที่ยังคงเกิดปัญหาต้นทุนที่เพิ่มมากขึ้น และร้อยละ 19 ที่โครงการต้องถูกยกเลิก (Rubinstein,2007) การประมาณขนาดของซอฟต์แวร์จึงเป็นส่วนสำคัญที่จำเป็นจะต้องคำนึงถึงในการบริหารโครงการ โดยก่อนหน้านี้ขนาดของซอฟต์แวร์จะถูกวัดด้วยจำนวนบรรทัดของโปรแกรม (Source Line of Code: SLOC) แต่การวัดจากจำนวนบรรทัดไม่อำนวยต่อการพัฒนาโปรแกรมในบางภาษา เพราะจำนวนบรรทัดของโปรแกรมอาจแตกต่างกันตามผู้พัฒนาแต่ละคน และที่สำคัญการที่จะรู้จำนวนบรรทัดได้ก็ต่อเมื่อโปรแกรมถูกพัฒนาขึ้นแล้ว จึงไม่ใช่วิธีที่สามารถนำมาช่วยได้ในขั้นต้นของโครงการพัฒนา

ต่อมาได้มีการคิดค้นวิธีการประมาณการซอฟต์แวร์ที่จะช่วยในการวิเคราะห์ในช่วงเริ่มต้นโครงการ ในปีค.ศ.1979 วิธีการประมาณซอฟต์แวร์ด้วยการวิเคราะห์ฟังก์ชั่นพอยต์ (Function Point Analysis :FPA) ก็ได้ถูกนำเสนอโดย Albrecht ซึ่งเป็นวิธีการคำนวณหาขนาดของซอฟต์แวร์จากฟังก์ชั่นของระบบที่จะเกิดขึ้นผ่านมุมมองของผู้ใช้งาน ตั้งแต่ขั้นตอนการวิเคราะห์และออกแบบ จึงเป็นสาเหตุทำให้ในปีค.ศ. 1979 วิธีการประมาณซอฟต์แวร์ด้วยการวิเคราะห์ฟังก์ชั่นพอยต์

ประสบความสำเร็จ (Albrecht , 1979) และ วิธีการนี้ยังได้รับการยอมรับว่าเป็นวิธีการประมาณการซอฟต์แวร์ที่สามารถใช้ได้โดยไม่ขึ้นกับเทคโนโลยีหรือภาษาใดๆ (Fetcke , Abran , and Nguyen,1997) เพราะการนับจำนวนฟังก์ชั่นจะนับจากจำนวนฟังก์ชั่นที่มีอยู่ในเอกสารความต้องการซอฟต์แวร์ (Software Requirement Specification)

วิธีอ็อบเจกต์พอยต์ (Object Point) วิธีการประมาณขนาดที่เหมาะกับแอพพลิเคชั่นที่ใช้เครื่องมือระดับสูงในการพัฒนาที่มีตัวช่วยในการออกแบบส่วนติดต่อผู้ใช้และเครื่องมือช่วยออกแบบระบบ อ็อบเจกต์ในที่นี้จะไม่ใช่อ็อบเจกต์ในลักษณะของซอฟต์แวร์เชิงวัตถุ แต่เป็นการนำข้อมูลจากเอกสารการออกแบบมาทำการคำนวณขนาดโดยจะพิจารณาจากเอกสารการออกแบบหน้าจอรายงาน และแบบฐานข้อมูลมาพิจารณาซึ่งให้ผลการคำนวณที่ใกล้เคียงกับวิธีฟังก์ชั่นพอยต์ (Issa, Odeh, and Coward ,2007) เหมาะสมกับการใช้ประมาณการในขั้นเริ่มต้นของโครงการเพราะไม่ต้องใช้ข้อมูลรายละเอียดทางเทคนิคในการพัฒนามาใช้ประกอบการประมาณการ(Malathi and Sridhar, 2012)

วิธีการประมาณขนาดด้วยยูสเคสพอยต์ (Use Case Point) เป็นวิธีที่อ้างอิงจากฟังก์ชั่นพอยต์ เสนอโดย Karner ในปี ค.ศ. 1993 จะใช้แผนภาพยูสเคสในการคำนวณทำให้วิธีการนี้สามารถทำได้ในขั้นต้นของการพัฒนา อีกทั้งยังคำนึงถึงปัจจัยทางด้านเทคนิคและปัจจัยทางสภาพแวดล้อมในการพัฒนาอีกด้วย แต่แผนภาพยูสเคสที่จะนำมาใช้ในการคำนวณผู้วาดจะต้องมีความชำนาญและความเข้าใจในซอฟต์แวร์ที่กำลังพัฒนา (Ribu,2001)

หลังจากการประมาณขนาดซอฟต์แวร์แล้วจะนำค่าขนาดของซอฟต์แวร์ที่ได้มาทำการแปลงเป็นค่าความพยายาม (Effort) เพื่อนำค่าจากการประมาณการไปใช้ประโยชน์ในการจัดการทรัพยากร (Resource)ของโครงการตลอดจนการจัดตารางเวลาและกำหนดการต่างๆ โดยที่ค่าความพยายามที่ใช้ในการพัฒนาซอฟต์แวร์ (Development effort) หมายถึง จำนวนแรงงานที่จะต้องใช้เพื่อทำงานให้เสร็จสมบูรณ์ ค่าความพยายามในงานวิจัยนี้จะใช้หน่วย คน-ชั่วโมง (Man-hours) คน-ชั่วโมง (Man-hours) คือ หน่วยของค่าความพยายามของหนึ่งคนในการทำงานด้วยจำนวนชั่วโมงเพื่อทำให้งานเสร็จสิ้น

เนื่องจากการพัฒนาโมบายแอพพลิเคชั่นเป็นการพัฒนาโครงการที่มีขนาดไม่ใหญ่และความซับซ้อนต่ำกว่าเมื่อเทียบกับซอฟต์แวร์ที่ทำงานบนเครื่องคอมพิวเตอร์ ในวงจรการพัฒนา (Software Development Life Cycle) โดยทั่วไปจะมีการจัดทำเอกสารรายละเอียดของการพัฒนาไม่กี่ประเภทจึงทำให้เอกสารที่จะสามารถนำมาใช้ในการประมาณการได้จะเป็นเอกสารความต้องการซอฟต์แวร์ (Software Requirement Specification) และ แบบจำลองหน้าจอที่จะจัดทำขึ้นตั้งแต่ช่วงของการวิเคราะห์และออกแบบระบบ ทั้งนี้จากการค้นหาข้อมูลผู้วิจัยได้พบว่ามีนักพัฒนาโมบายแอพพลิเคชั่น

บางรายได้เสนอวิธีการประมาณการของตัวเองโดยใช้ข้อมูลดังกล่าว ซึ่งเป็นวิธีการอย่างง่าย โดยในปี ค.ศ. 2012 วิธีการหนึ่งที่น่าสนใจถูกเสนอโดย Nikunj Sakhrelia นักพัฒนาซอฟต์แวร์ชาวอินเดีย ซึ่งวิธีการนี้มีลักษณะการประมาณการคล้ายคลึงกับวิธีอ็อบเจกต์พอยต์ สอดคล้องกับหลักการของการประมาณการขนาดซอฟต์แวร์ คือ วิธีการประมาณการที่ทำให้เกิดความแม่นยำที่สุดจะต้องเป็นวิธีที่สามารถประมาณการขนาดได้ในช่วงการออกแบบซอฟต์แวร์

ดังนั้นในงานวิจัยครั้งนี้ผู้วิจัยจึงศึกษาโดยการเปรียบเทียบวิธีการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น ด้วยข้อมูลจากแอพพลิเคชั่นที่ถูกพัฒนาขึ้นจริง ซึ่งเมื่อพิจารณาจากข้อมูลที่ได้จากนักพัฒนาโมบายแอพพลิเคชั่น ผู้วิจัยได้เลือกเปรียบเทียบวิธีการประมาณขนาดโมบายแอพพลิเคชั่น

(1) วิธีฟังก์ชั่นพอยต์ (2) วิธีอ็อบเจกต์พอยต์ (3)วิธียูสเคสพอยต์ และ (4) วิธีของ Nikunj Sakhrelia นักพัฒนาโมบายแอพพลิเคชั่นอิสระ เพื่อทราบถึงวิธีการประมาณขนาดที่เหมาะสมสำหรับใช้ในการประมาณการค่าความพยายามของการพัฒนาโมบายแอพพลิเคชั่น

1.2 วัตถุประสงค์ของการวิจัย

เปรียบเทียบวิธีการประมาณการค่าความพยายามของการพัฒนาโมบายแอพพลิเคชั่น ด้วยวิธีการประมาณการซอฟต์แวร์ 4 วิธี คือ (1) วิธีฟังก์ชั่นพอยต์ (2) วิธีการอ็อบเจกต์พอยต์ (3)วิธียูสเคสพอยต์ และ (4) วิธีของ Nikunj Sakhrelia

1.3 ขอบเขตของการวิจัย

1. ศึกษาเฉพาะแอพพลิเคชั่นที่ทำงานบนสมาร์ตโฟน
2. การประมาณการค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นที่ใช้ในงานวิจัยนี้จะอยู่บนพื้นฐานของวิธีการประมาณขนาดซอฟต์แวร์ 4 วิธีการ คือ (1) วิธีฟังก์ชั่นพอยต์ (2) วิธีการอ็อบเจกต์พอยต์ (3)วิธียูสเคสพอยต์ และวิธีที่เสนอโดยนักพัฒนาโมบายแอพพลิเคชั่นอิสระ ได้แก่ (4) วิธีของ Nikunj Sakhrelia

1.4 ขั้นตอนและวิธีการดำเนินการวิจัย

1. ดำเนินการสอบความเป็นไปได้ในการได้ข้อมูล
2. สร้างเครื่องมือการเก็บข้อมูล
3. เก็บรวบรวมข้อมูลจากบริษัทและนักพัฒนาโมบายแอพพลิเคชั่นอิสระ
4. ประมาณการขนาดโมบายแอพพลิเคชั่นด้วยวิธีการประมาณขนาดซอฟต์แวร์ทั้ง 4 วิธี
5. ประมาณค่าความพยายาม (Effort) ของโมบายแอพพลิเคชั่นด้วยวิธีการทั้ง 4 วิธี
6. เปรียบเทียบผลการประมาณค่าความพยายามของทั้ง 4 วิธี
7. สรุปผลการวิจัย

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เป็นแนวทางแก่นักพัฒนาโมบาย แอพพลิเคชั่นในการประมาณค่าความพยายาม
2. เป็นข้อมูลสำหรับผู้จัดการโครงการโมบาย แอพพลิเคชั่นในการเลือกวิธีการประมาณการต้นทุนและค่าความพยายาม
3. ช่วยผู้จัดการโครงการในการจัดการต้นทุนการพัฒนา

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 2
ทฤษฎีและวรรณกรรมที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎีและวรรณกรรมที่เกี่ยวข้องกับการประมาณการขนาดซอฟต์แวร์เพื่อใช้ในการประมาณค่าความพยายามในการพัฒนาซอฟต์แวร์ ได้แก่ วิธีฟังก์ชั่นพอยต์ (IFPUG,2000) วิธีอ็อบเจกต์พอยต์ (Banker,1994) วิธียูสเคสพอยต์ (Karner , 1993) และวิธีการที่เสนอโดย Nikunj Sakhrelia (Sakhrelia, 2012 :Online) นักพัฒนาโมบายแอพพลิเคชั่นชาวอินเดีย ซึ่งมีรายละเอียดต่างๆ ดังนี้

2.1 การประมาณขนาดซอฟต์แวร์ด้วยการวิเคราะห์ฟังก์ชั่นพอยต์ (Function Point Analysis : FPA)

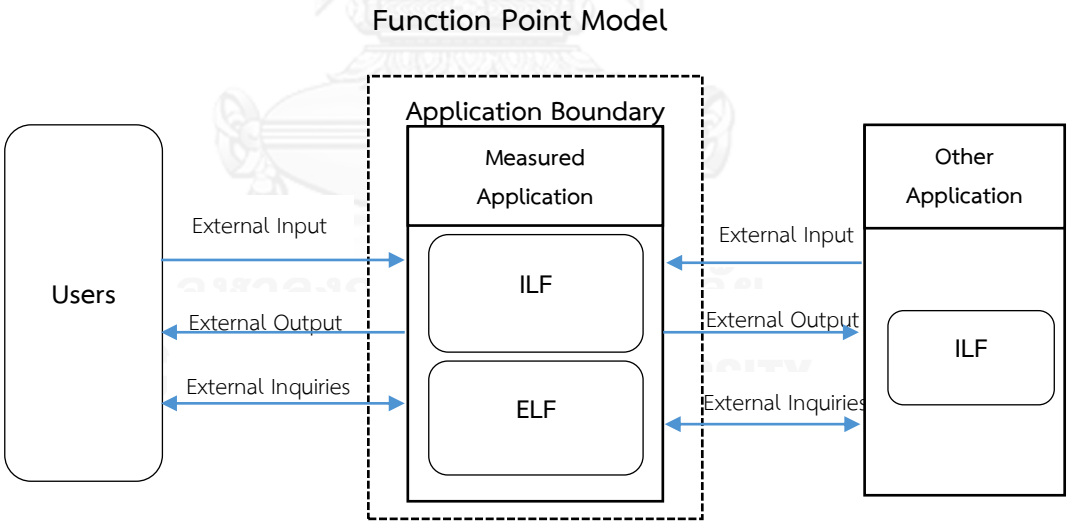
การประมาณขนาดซอฟต์แวร์ด้วยการวิเคราะห์ฟังก์ชั่นพอยต์ คือ วิธีการประมาณขนาดด้วยการนับจำนวนฟังก์ชั่นการทำงานจากข้อกำหนดความต้องการ (Requirement Specification) โดยวิธีนี้ถูกคิดค้นขึ้นในปี ค.ศ. 1979 โดย A.J. Albrecht (Saxena and Shrivastava,2009); (วรัฏฐา เจริญสถาพงษ์,2549) ซึ่งอาศัยหลักการคิดที่ว่าหากแบ่งงานหรือปัญหาออกเป็นส่วนย่อยๆอย่างอิสระต่อกันจะช่วยทำให้มนุษย์ทำความเข้าใจและแก้ปัญหานั้นๆ ได้ง่ายขึ้น ส่งผลให้ทำงานต่างๆได้สะดวกมากขึ้น ดังนั้นในการวิเคราะห์ฟังก์ชั่นพอยต์จึงได้มีการแบ่งฟังก์ชั่นการทำงานจากข้อกำหนดความต้องการออกเป็นส่วนย่อยๆ จากกัน และเนื่องจากการใช้ข้อมูลจากข้อกำหนดความต้องการมาใช้ในการวิเคราะห์ วิธีนี้จึงสามารถนำไปประยุกต์ใช้ได้ง่ายโดยไม่สนใจความแตกต่างของภาษาที่ใช้พัฒนารวมถึงเทคโนโลยีอื่นๆด้วย วิธีฟังก์ชั่นพอยต์นี้ได้รับการปรับปรุงให้เหมาะสมทันสมัยอยู่เสมอโดยกลุ่มผู้ใช้งานฟังก์ชั่นพอยต์สากล (International Function Point Users Group : IFPUG) และสมาคมผู้ใช้ตัววัดซอฟต์แวร์แห่งเนเธอร์แลนด์ (The Netherlands Software Metrics users Association :NESMA) โดยมาตรฐานของ IFPUG จะเรียกว่า IFPUG FSM (Functional Size Measurement) และมาตรฐานของ NESMA จะเรียกว่า NESMA FSM โดยมีสูตรในการคำนวณ ดังนี้

$$FP = UFP * VAF$$

จากสูตรคำนวณ ฟังก์ชั่นพอยต์(Function Point : FP) จะได้จาก ค่าฟังก์ชั่นพอยต์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point : UFP) คูณกับปัจจัยคุณลักษณะของระบบ (Value Adjustment Factor : VAF) การนำค่าคุณลักษณะของระบบมาคำนวณด้วยนั้นเพราะแต่ละระบบย่อมมีความแตกต่างกันความยากง่ายในการพัฒนาแตกต่างกัน ทั้งนี้ขึ้นอยู่กับความสามารถของนักพัฒนา หากไม่นำปัจจัยเหล่านี้มาคำนวณก็จะทำให้ค่า FP ที่ได้ ไม่สามารถแสดงถึงประสิทธิผลการทำงานที่แท้จริง ขั้นตอนการหาค่า FP จากแบ่งเป็น 3 ขั้นตอน คือ (1) คำนวณหาค่าฟังก์ชั่นพอยต์ที่ยังไม่ถูกปรับค่า (UFP) (2) คำนวณค่าปัจจัยคุณลักษณะของระบบ (VAF) และ (3) คำนวณค่าฟังก์ชั่นพอยต์ที่ปรับค่าแล้ว

2.1.1. คำนวณหาค่าฟังก์ชั่นพอยต์ที่ยังไม่ถูกปรับค่า (UFP)

ในการวิเคราะห์ฟังก์ชันพอยต์ ได้จำแนกความต้องการเชิงฟังก์ชันของผู้ใช้ (Functional User Requirement) ออกเป็นสองประเภท คือ ฟังก์ชันเชิงข้อมูล (Data Function) และฟังก์ชันเชิงรายการเปลี่ยนแปลง (Transaction Function) โดยแบ่งออกเป็น 5 หมวดหมู่ตามหลักของวิธี IFPUG FSM ดังแสดงในรูปที่ 2.1



รูปที่ 2.1 แสดงโมเดลของการวิเคราะห์ฟังก์ชั่นพอยต์ (Fecke, Abran and Nguyen, 1998)

1. ฟังก์ชันเชิงข้อมูล

1) แฟ้มข้อมูลเชิงตรรกะภายใน (Internal Logical Files: ILFs) ได้แก่ ไฟล์ข้อมูลหลักที่ระบบเป็นเจ้าของ หรือไฟล์ข้อมูลที่สามารถสร้าง เรียกใช้ หรือเปลี่ยนแปลงได้โดยระบบ โดยมักจะถูกปรับปรุงด้วยข้อมูลจากภายนอก(External Input) เรคคอร์ดที่ทำหน้าที่เทียบเท่ากับไฟล์จะถูกนับรวมเป็น ILFs ด้วย

2) แฟ้มข้อมูลที่ต้องการจากภายนอก (External Interface Files: EIFs) ได้แก่ ไฟล์ข้อมูลที่ใช้เพื่อการอ้างอิงเท่านั้น โดยใช้ร่วมกันกับระบบอื่น ระบบสามารถเรียกใช้ได้แต่จะถูกเก็บรักษาโดยระบบภายนอก

2. ฟังก์ชันเชิงรายการเปลี่ยนแปลง

1) ข้อมูลนำเข้าภายนอก (External Inputs: EIs) ข้อมูลที่รับเข้ามาในระบบเพื่อปรับปรุงข้อมูลใน ILFs

2) ข้อมูลส่งออกภายนอก (External Outputs: EOs) ข้อมูลที่เกิดจากการประมวลผลข้อมูลจากภายในระบบเพื่อนำส่งข้อมูลออกสู่ภายนอกโดยให้นับการแสดงข้อมูลที่มีรูปแบบแตกต่างกัน

3) การสอบถามภายนอก (External Inquiries: EQs) การประมวลผลและแสดงข้อมูลเมื่อมีการสอบถามจากภายนอก (Query) เมื่อแยกฟังก์ชั่นการทำงานออกเป็นประเภทต่างๆ แล้ว ก็จะทำการคิดค่าความซับซ้อนของฟังก์ชั่นแต่ละประเภท คือ

- แฟ้มข้อมูลเชิงตรรกะภายใน (Internal Logical Files: ILFs) และ แฟ้มข้อมูลที่ต้องการจากภายนอก (External Interface Files: EIFs) จะคิดค่าความซับซ้อนจาก การนับองค์ประกอบข้อมูล (Data Element Types : DET) คือ ฟิลด์ (Field) ข้อมูลที่สนใจในแต่ละตาราง และ เรคคอร์ดข้อมูล (Record Element Types : RET) ซึ่งหมายถึง เรคคอร์ดข้อมูลที่มีความสัมพันธ์กับฟังก์ชั่นที่สนใจ หรือส่วนที่เป็นเอนทิตีย่อย โดยมีรายละเอียดดังนี้

  วิธีนับ DET

  1. แต่ละฟิลด์ข้อมูลจะนับเป็น 1 DET
  2. นับ 1 DET สำหรับแต่ละกลุ่มข้อมูลจากผู้ใช้ ที่ไปมีผลกระทบต่อ ILF หรือ EIF
  3. หากมีการเรียกใช้งาน ILF หรือ EIF จากหลายส่วนของระบบจะนับ DET เฉพาะฟิลด์ข้อมูลที่ส่วนนั้นๆเรียกใช้งาน

วิธีนับ RET

1. แต่ละเรคคอร์ดข้อมูลที่มีความสัมพันธ์กับ ตารางหรือเอนทิตีย่อยอื่นให้นับ 1 RET
2. หากไม่มีความสัมพันธ์กับ ตารางหรือเอนทิตีย่อยอื่น ให้นับ 1 DET
3. นำค่า DET ,RET ที่นับได้มาคำนวณค่าความซับซ้อนจากตารางที่ 2.1

ตารางที่ 2.1 แสดงระดับความซับซ้อนของ ILFs และ EIFs (IFPUG, 2000)

	DET		
RET	**1 - 19**	**20 – 50**	**มากกว่า 50**
1	ต่ำ	ต่ำ	ปานกลาง
2-5	ต่ำ	ปานกลาง	สูง
ตั้งแต่ 6 ขึ้นไป	ปานกลาง	สูง	สูง

ค่าฟังก์ชั่นพอยต์ (Function Point) ของแต่ละระดับความซับซ้อนของ ILFs คือ

- ระดับความซับซ้อนต่ำ มีค่าฟังก์ชั่นพอยต์ เท่ากับ 7
- ระดับความซับซ้อนปานกลาง มีค่าฟังก์ชั่นพอยต์ เท่ากับ 10
- ระดับความซับซ้อนสูง มีค่าฟังก์ชั่นพอยต์ เท่ากับ 15

ค่าฟังก์ชั่นพอยต์ ของแต่ละระดับความซับซ้อนของ EIFs คือ

- ระดับความซับซ้อนต่ำ มีค่าฟังก์ชั่นพอยต์ เท่ากับ 5
- ระดับความซับซ้อนปานกลาง มีค่าฟังก์ชั่นพอยต์ เท่ากับ 7
- ระดับความซับซ้อนสูง มีค่าฟังก์ชั่นพอยต์ เท่ากับ 10

- ข้อมูลนำเข้าภายนอก (External Inputs: EIs) จะคิดค่าความซับซ้อนจากการนับ จำนวนประเภทของไฟล์ข้อมูลอ้างอิง (File Type Referenced : FTR) และ การนับองค์ประกอบข้อมูล (DET) โดยมีรายละเอียดดังนี้

วิธีนับ FTR

1. นับ 1 FTR สำหรับแต่ละ ILF ที่ถูกเพิ่ม แก้ไข ปรับปรุงภายใต้กระบวนการของ EI

2. นับ 1 FTR สำหรับแต่ละ ILF หรือ ELF ที่ถูกอ้างอิงถึงในกระบวนการ EI
3. นับ 1 FTR สำหรับแต่ละ ILF ที่ถูกเรียกเก็บรักษาภายใต้กระบวนการของ EI

วิธีนับ DET

1. นับ 1 DET สำหรับแต่ฟิลด์ข้อมูลหรือแอนทริบิวต์ที่ได้รับการกำหนดจากข้อกำหนดความต้องการผู้ใช้ (ไม่นับที่ซ้ำ) รวมทั้งนับ คีย์นอก (Foreign Key) ที่เกิดขึ้นจากการทำงาน เพื่อให้ระบบย่อยได้ทำงานอย่างสมบูรณ์
2. ไม่นับฟิลด์ที่ไม่ได้กรอกโดยผู้ใช้ เช่นการคำนวณวันที่ปัจจุบันโดยระบบ
3. นับ 1 DET สำหรับการส่งข้อความแจ้งเตือน (Notification) ไปยังภายนอกเพื่อแจ้งข้อผิดพลาดหรือแจ้งสถานะของการทำงาน
4. นับ 1 DET สำหรับการทำงาน 1 งานจากกระบวนการ EI
5. นำค่า DET ,FTR ที่นับได้มาคำนวณค่าความซับซ้อนจากตารางที่ 2.2

ตารางที่ 2.2 แสดงระดับความซับซ้อนของ EI (IFPUG,2000)

	DET		
FTR	**2. - 4**	**5 – 15**	**มากกว่า 15**
<2	ต่ำ	ต่ำ	ปานกลาง
2	ต่ำ	ปานกลาง	สูง
>2	ปานกลาง	สูง	สูง

ค่าฟังก์ชั่นพอยต์ของแต่ละระดับความซับซ้อนคือ

- ระดับความซับซ้อนต่ำ มีค่าฟังก์ชั่นพอยต์เท่ากับ 3
- ระดับความซับซ้อนปานกลาง มีค่าฟังก์ชั่นพอยต์เท่ากับ 4
- ระดับความซับซ้อนสูง มีค่าฟังก์ชั่นพอยต์เท่ากับ 6

- ข้อมูลส่งออกภายนอก (External Outputs: EOs) และ การสอบถามภายนอก (External Inquiries: EQs) จะคิดค่าความซับซ้อนจากการนับ จำนวนประเภทของไฟล์ข้อมูลอ้างอิง (File Type Referenced : FTR) และ การนับองค์ประกอบข้อมูล (DET) โดยมีรายละเอียดดังนี้

  วิธีนับ FTR

  1. นับ 1 FTR สำหรับ 1 ILF ที่ถูกเก็บรักษาภายใต้กระบวนการของ EO หรือ EQ
  2. นับ 1 FTR สำหรับ 1 ILF หรือ ELF ที่ถูกอ้างอิงถึงด้วยกระบวนการของ EO หรือ EQ
  3. นับเพียง 1 FTR หาก ILF หรือ ELF นั้นๆ ได้รับการอ่าน อ้างอิง หรือเก็บรักษาโดยกระบวนการของ EO หรือ EQ

  วิธีนับ DET

  1. นับ 1 DET สำหรับแต่ฟิลด์ข้อมูลหรือแอนทริบิวต์ที่ได้รับการกำหนดจากข้อกำหนดความต้องการผู้ใช้ (ไม่นับที่ซ้ำ) รวมทั้งนับ คีย์นอก (Foreign Key) ที่เกิดขึ้นจากการทำงาน เพื่อให้ระบบย่อยได้ทำงานอย่างสมบูรณ์
  2. นับ 1 DET สำหรับ แต่ละฟิลด์หรือแอททริบิวต์ที่อยู่นอกระบบงาน
  3. หาก DET นั้นๆ เกิดขึ้นทั้งภายในและภายนอกระบบให้นับเพียง 1 DET
  4. นับ 1 DET สำหรับการส่งข้อความแจ้งเตือน (Notification) ไปยังภายนอกเพื่อแจ้งข้อผิดพลาดหรือแจ้งสถานะการทำงาน
  5. นับเป็น 1 DET สำหรับแต่ละความสามารถในการกำหนดการทำงานที่จัดการโดย EO และ EQ เช่น นับ 1 DET สำหรับการกดปุ่มตกลง, การคลิกเมาส์ที่ให้ผลเท่ากับการตอบตกลง
  6. นับ 1 DET สำหรับฟิลด์ข้อมูลหนึ่งฟิลด์ที่มีการจัดเก็บข้อมูลเชิงตรรกะไว้หลายที่ เช่น ข้อมูลที่อยู่ ที่เก็บข้อมูลแยกฟิลด์ ถนน อำเภอ จังหวัด เป็นต้น
  7. นับ 1 DET สำหรับข้อมูลที่ประกอบด้วยคำ ประโยค หรือย่อหน้า
  8. นำค่า DET ,FTR ที่นับได้มาคำนวณค่าความซับซ้อนจากตารางที่ 2.3

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 2.3 แสดงระดับความซับซ้อนของ EO และ EQ (IFPUG,2000)

FTR	DET		
	1- 5	**6 – 19**	**มากกว่า 19**
<2	ต่ำ	ต่ำ	ปานกลาง
2 - 3	ต่ำ	ปานกลาง	สูง
>3	ปานกลาง	สูง	สูง

ค่าฟังก์ชั่นพอยต์ของแต่ละระดับความซับซ้อน EO คือ

- ระดับความซับซ้อนต่ำ มีค่าฟังก์ชั่นพอยต์เท่ากับ 4
- ระดับความซับซ้อนปานกลาง มีค่าฟังก์ชั่นพอยต์เท่ากับ 5
- ระดับความซับซ้อนสูง มีค่าฟังก์ชั่นพอยต์เท่ากับ 7

ค่าฟังก์ชั่นพอยต์ของแต่ละระดับความซับซ้อน EQ คือ

- ระดับความซับซ้อนต่ำ มีค่าฟังก์ชั่นพอยต์เท่ากับ 3
- ระดับความซับซ้อนปานกลาง มีค่าฟังก์ชั่นพอยต์เท่ากับ 4
- ระดับความซับซ้อนสูง มีค่าฟังก์ชั่นพอยต์เท่ากับ 6

หลังจากนั้นนำจำนวนฟังก์ชั่นพอยต์ทั้งหมดที่คำนวณได้มากรอกใส่ในตารางที่ 2.4 เพื่อจะคำนวณค่าฟังก์ชั่นพอยต์ที่ยังไม่ได้ปรับค่า

ตารางที่ 2.4 แสดงการนับค่าฟังก์ชั่นพอยต์ (IFPUG,2000)

ชนิด	**ระดับความซับซ้อน**			**ผลรวม**
	ต่ำ	**ปานกลาง**	**สูง**	
ILF	_____x 7	_____x 10	_____x 15	_____
ELF	_____x 5	_____x 7	_____x 10	_____
EI	_____x 3	_____x 4	_____x 6	_____
EO	_____x 4	_____x 5	_____x 7	_____
EQ	_____x 3	_____x 4	_____x 6	_____
UFP				_____

2.1.2. คำนวณตัวแปรปรับค่า (Value Adjustment Factor: VAF)

หลังจากที่ได้ค่าฟังก์ชั่นพอยต์จากขั้นตอนแรกแล้ว การประมาณขนาดซอฟต์แวร์ด้วยวิธีฟังก์ชั่นพอยต์ จะต้องทำการปรับค่าด้วยตัวแปรปรับค่า โดยคำนวณจากค่าปัจจัยทั้ง 14 ข้อ ได้แก่ ปัจจัย Data Communication (ตารางที่ 2.5) ปัจจัย Distributed Data Processing (ตารางที่ 2.6) ปัจจัย Performance (ตารางที่ 2.7) ปัจจัย Heavily Used Configuration (ตารางที่ 2.8) ปัจจัย Transaction Rate (ตารางที่ 2.9) ปัจจัย On-Line Data Entry (ตารางที่ 2.10) ปัจจัย End-User Efficiency (ตารางที่ 2.11) ปัจจัย Online Update (ตารางที่ 2.12) ปัจจัย Complex Processing (ตารางที่ 2.13) ปัจจัย Reusability (ตารางที่ 2.14) ปัจจัย Installation Ease (ตารางที่ 2.15) ปัจจัย Operational Ease (ตารางที่ 2.16) ปัจจัย Multiple Sites (ตารางที่ 2.17) ปัจจัย Facilitate change (ตารางที่ 2.18) โดยให้กำหนดอิทธิพลของปัจจัยในแต่ละด้านว่ามีความเกี่ยวข้องกับระบบมากน้อยเพียงใด โดยมีระดับของอิทธิพลตั้งแต่ 0 – 5 ตามคำอธิบายในตาราง 2.5-2.18

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 2.5 แสดงการกำหนดระดับปัจจัย Data Communication (Longstreet 2004)

1.Data Communication จะมองเรื่องของจำนวนของสิ่งที่ช่วยอำนวยความสะดวกในการถ่ายโอนข้อมูล หรือแลกเปลี่ยนข้อมูลระหว่างซอฟต์แวร์หรือระบบ	
0	ซอฟต์แวร์ทั้งหมดรันแบบ Batch หรือ standalone PC
1	ซอฟต์แวร์เป็นแบบ Batch แต่มีการใช้ลักษณะของ Remote Data Entry หรือ Remote Printing
2	ซอฟต์แวร์เป็นแบบ Batch แต่มีการใช้ลักษณะของ Remote Data Entry และ Remote Printing
3	ซอฟต์แวร์เป็นแบบ Online Data Collection หรือ *Teleprocessing จากส่วน Front-end สู่ Batch
4	ซอฟต์แวร์มีลักษณะของ Teleprocessing ซึ่งมากกว่าในส่วนของ Front-end แต่สนับสนุนการติดต่อ Teleprocessing Protocol เพียงชนิดเดียว
5	ซอฟต์แวร์มีลักษณะของ Teleprocessing ซึ่งมากกว่าในส่วนของ Front-end แต่สนับสนุนการติดต่อ Teleprocessing Protocol ที่หลากหลายชนิดมากขึ้น

ตารางที่ 2.6 แสดงการกำหนดระดับปัจจัย Distributed Data Processing (Longstreet 2004)

2. Distributed Data Processing จะมองในเรื่องการจัดการข้อมูลในเชิง Distributed และขั้นตอนหรือหน้าที่ในการจัดการ	
0	ซอฟต์แวร์สามารถช่วยในการถ่ายข้อมูลหรือช่วยในการดำเนินการระหว่างกันได้
1	ซอฟต์แวร์สามารถเตรียมข้อมูลสำหรับ End User เพื่อดำเนินขั้นตอนต่อไปในส่วนอื่นๆ เช่น PC Spreadsheets และ PC DBMS
2	ซอฟต์แวร์สามารถเตรียมข้อมูลเพื่อถ่ายโอนและถ่ายโอนไปยังขั้นตอนอื่นๆ ในระบบเพื่อประมวลผลได้ (ไม่เพียงแต่ถ่ายโอนสำหรับ End User)
3	ซอฟต์แวร์มีลักษณะของ Distributed processing และการถ่ายโอนข้อมูลเป็นแบบออนไลน์แต่เป็นในทิศทางเดียว

2. Distributed Data Processing จะมองในเรื่องการจัดการข้อมูลในเชิง Distributed และขั้นตอนหรือหน้าที่ในการจัดการ	
4	ซอฟต์แวร์มีลักษณะของ Distributed processing และการถ่ายโอนข้อมูลเป็นแบบออนไลน์ แต่เป็นใน 2 ทิศทาง
5	ซอฟต์แวร์มีการจัดการการถ่ายโอนข้อมูลแบบไดนามิกขึ้นอยู่กับแต่ละส่วนของระบบ

ตารางที่ 2.7 แสดงการกำหนดระดับปัจจัย Performance (Longstreet 2004)

3. Performance จะมองในเรื่องของ Response Time ประสิทธิภาพของซอฟต์แวร์นั้นจะได้รับการประเมินโดย User ซึ่งเป็นผลสืบเนื่องจากการออกแบบพัฒนาการติดตั้งและรองรับซอฟต์แวร์นั้นๆ	
0	ไม่มีสภาพของประสิทธิภาพที่ User ต้องการเป็นพิเศษ
1	ประสิทธิภาพและการออกแบบตามความต้องการของ User ได้รับการทบทวนแต่ไม่มีความต้องการกระทำอะไรเป็นพิเศษ
2	Response Time และผลลัพธ์ของการทำงานเป็นเรื่องสำคัญและจำเป็นระหว่างช่วงเวลาที่มีการใช้งานสูงๆ การประมวลผลต้องให้เรียบร้อยก่อนวัดถัดไป
3	Response Time และผลลัพธ์ของการทำงานเป็นเรื่องที่สำคัญและจำเป็นระหว่างช่วงเวลาที่มีการใช้งานในธุรกิจ ไม่มีการออกแบบสำหรับ CPU ที่ต้องการใช้งานเฉพาะที่มีการประมวลผลที่มีความต้องการเฉพาะ รวมทั้งการออกแบบอินเตอร์เฟสมีการระบุเฉพาะ
4	เพิ่มในส่วนของความต้องการของ User ในเรื่องของประสิทธิภาพตั้งแต่ขั้นตอนของการวิเคราะห์ระบบงาน
5	เพิ่มในส่วนของการนำเครื่องมือในการวิเคราะห์ประสิทธิภาพเข้ามาใช้ในขั้นตอนการออกแบบ พัฒนา และนำไปใช้เพื่อที่จะให้ตรงกับประสิทธิภาพที่ User มี

ตารางที่ 2.8 แสดงการกำหนดระดับปัจจัย Heavily Used Configuration (Longstreet 2004)

4. Heavily Used Configuration จะมองในเรื่องของการใช้ที่เกี่ยวกับฮาร์ดแวร์ และ Platform ซึ่งเป็นคุณลักษณะหนึ่งของซอฟต์แวร์ เช่น User ต้องการรันซอฟต์แวร์บนอุปกรณ์หรือเครื่องที่มีการใช้งานอย่างหนัก	
0	ไม่มีความชัดเจนในเรื่องของความเข้มงวดในความสามารถในการจัดการให้สามารถใช้งานได้
1	มีความเข้มงวดในการจัดการให้ใช้งานได้แต่ความเข้มงวดก็ยังน้อยในซอฟต์แวร์ซึ่งเป็นตัวอย่าง
2	มีระบบรักษาความปลอดภัยหรือการคำนึงถึงข้อจำกัดด้านเวลารวมอยู่ด้วย
3	มีการเจาะจงถึง Processor (เช่น Centrino P4) สำหรับซอฟต์แวร์
4	มีการระบุถึงความเข้มงวดอย่างยิ่งในการประมวลผลซอฟต์แวร์ต่อหน่วยประมวลผลกลาง (Central Processor) ซึ่งมีลักษณะที่เฉพาะเจาะจง
5	มีข้อจำกัดพิเศษบนซอฟต์แวร์ในส่วนประกอบแบบ Distributed ของระบบ

ตารางที่ 2.9 แสดงการกำหนดระดับปัจจัย Transaction Rate (Longstreet 2004)

5. Transaction Rate จะมองในเรื่องความถี่ในการประมวลผล Transaction ว่าเป็นแบบรายวัน รายสัปดาห์ รายเดือน	
0	ไม่มีการคาดถึงช่วงที่มีระดับสูงสุดของ Transaction
1	ช่วงที่ระดับของจำนวน Transaction เพิ่มสูงสุดนั้นได้ถูกคาดคะเน (รายเดือน รายปี)
2	มีการคาดคะเนว่าจำนวน Transaction สูงสุดเป็นรายสัปดาห์
3	มีการคาดคะเนว่าจำนวน Transaction สูงสุดเป็นรายวัน
4	มีอัตราของ Transaction สูงมาก Requirement ต้องการมากเพียงพอเพื่อที่จะมีการวิเคราะห์ประสิทธิภาพของงานในขั้นตอนการออกแบบ
5	มีอัตราของ Transaction สูงมาก Requirement ต้องมากเพียงพอเพื่อที่จะมีการวิเคราะห์ประสิทธิภาพของงานในขั้นตอนการออกแบบ รวมไปถึงขั้นตอนของการพัฒนาและการติดตั้ง

ตารางที่ 2.10 แสดงการกำหนดระดับปัจจัย On-Line Data Entry (Longstreet 2004)

6. On-Line Data Entry จะมองในเรื่องของเปอร์เซ็นต์ของข้อมูลที่ผ่านมาทางออนไลน์	
0	ทุกๆ Transaction จะเป็นรูปแบบของ Batch
1	1% ถึง 7% ของ Transaction เป็น Interactive Data Entry
2	8% ถึง 15% ของ Transaction เป็น Interactive Data Entry
3	16% ถึง 23% ของ Transaction เป็น Interactive Data Entry
4	24% ถึง 30% ของ Transaction เป็น Interactive Data Entry
5	> 30% ของ Transaction เป็น Interactive Data Entry

ตารางที่ 2.11 แสดงการกำหนดระดับปัจจัย End-User Efficiency (Longstreet 2004)

7. End-User Efficiency จะมองในเรื่องของผู้ใช้งานระบบว่าสามารถใช้งานระบบได้อย่างมีประสิทธิภาพหรือไม่ ซึ่งลักษณะของปัจจัยที่ส่งผลต่อ User Friendly การให้ความสำคัญกับประสิทธิภาพประกอบด้วย

- Navigation Aids
- Menu
- Online Help and Document
- Automated Cursor Movement
- Scrolling
- Remote Printing (Via Online Transaction)
- Reassigned Function Keys
- Batched Job Submitted From Online Transactions
- Cursor Selection Of Screen Data
- Heavy Use Of Reverse Video Highlighting Color Underling And Other Indicators
- Hard Copy User Documentation Of Online Transactions
- Mouse Interface
- Pop Up Windows
- As Few Screen As Possible To Accomplish A Business Function
- Bilingual Support

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

- Multilingual Support	
0	ไม่มีการให้ความสำคัญกับประสิทธิภาพตามข้อกำหนดข้างต้น
1	มีการให้ความสำคัญกับประสิทธิภาพ 1-3 ข้อจากข้อกำหนดข้างต้น
2	มีการให้ความสำคัญกับประสิทธิภาพ 4-5 ข้อจากข้อกำหนดข้างต้น
3	มากกว่า 6 ข้อจากข้อกำหนดข้างต้นโดย User ไม่มีความต้องการที่เฉพาะในเรื่อง Efficiency
4	มากกว่า 6 ข้อจากข้อกำหนดข้างต้นโดย User มีการใช้ความสนใจกับเรื่องของ Efficiency โดยต้องมีการออกแบบซอฟต์แวร์อย่างเฉพาะเพื่อสนับสนุน Efficiency
5	มากกว่า 6 ข้อจากข้อกำหนดข้างต้นโดย User มีการใช้ความสนใจกับเรื่องของ Efficiency โดยต้องมีการใช้งานอุปกรณ์และขั้นตอนที่พิเศษซึ่งจะทำให้การทำงานนั้นบรรลุตามวัตถุประสงค์

ตารางที่ 2.12 แสดงการกำหนดระดับปัจจัย On-Line Update (Longstreet 2004)

8. Online Update จะมองในเรื่องของ Internal Logical File ว่ามีจำนวนมากน้อยเพียงใด ที่ได้รับการอัพเดทผ่าน Online Transaction	
0	ไม่มีการอัพเดทออนไลน์
1	การอัพเดทเป็นการอัพเดท 1-3 ไฟล์โดยการอัพเดทนั้นเป็นไปได้ช้า
2	การอัพเดทเป็นการอัพเดท 4 ไฟล์ขึ้นไป โดยการอัพเดทนั้นเป็นไปได้ช้า
3	สามารถออนไลน์อัพเดทได้ในส่วนหลักๆของ Internal Logical Files
4	เพิ่มเติมการป้องกันข้อมูลสูญหายในขณะทำ Online Update ซึ่งได้มีการออกแบบและสร้างอย่างพิเศษในซอฟต์แวร์
5	เพิ่มเติมการป้องกันข้อมูลสูญหายของข้อมูลรวมทั้งมีการจัดการระบบกู้ข้อมูลอัตโนมัติ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 2.13 แสดงการกำหนดระดับปัจจัย Complex Processing (Longstreet 2004)

9. Complex Processing เป็นส่วนของคุณลักษณะของซอฟต์แวร์ โดยส่วนประกอบมีดังนี้ - มีส่วนขยายของ Logical Processing - มีส่วนขยายของ Mathematical Processing - มีการประมวลผลที่ซับซ้อนเพื่อจัดการกับ Input และ Output เช่นการมี Multimedia - ข้อผิดพลาดที่เกิดจากการประมวลผลก่อให้เกิด Transaction ที่ไม่สมบูรณ์นั้นต้องทำการประมวลผลอีกครั้ง - Sensitive Control เช่นความปลอดภัยในการประมวลผลข้อมูล	
0	ไม่มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน
1	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน 1 ข้อ
2	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน 2 ข้อ
3	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน 3 ข้อ
4	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน 4 ข้อ
5	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบนครบ 5 ข้อ

ตารางที่ 2.14 แสดงการกำหนดระดับปัจจัย Reusability (Longstreet 2004)

10. Reusability จะมองในเรื่องของซอฟต์แวร์ที่ได้ทำการพัฒนานั้นสามารถนำไปปรับใช้หรือสามารถนำไปพัฒนาต่อไปในโครงการอื่นๆ ได้หรือไม่	
0	ไม่มีการ Reuse โค้ด
1	การ Reuse มีเพียงภายในซอฟต์แวร์
2	มีการนำโค้ดไป Reuse น้อยกว่า 10% ในหลายๆ งาน
3	มีการนำโค้ดไป Reuse มากกว่า 10% ในหลายๆ งาน
4	ทั้งซอฟต์แวร์และเอกสารประกอบการพัฒนาค่อนข้างเอื้อต่อการนำไป Reuse
5	ทั้งแอพพลิเคชั่นและเอกสารประกอบการพัฒนาเอื้อต่อการนำไป Reuse

ตารางที่ 2.15 แสดงการกำหนดระดับปัจจัย Installation Ease (Longstreet 2004)

11. Installation Ease จะมองในเรื่องของการติดตั้งซอฟต์แวร์	
0	ไม่มีลักษณะพิเศษในการติดตั้ง
1	มีลักษณะพิเศษในการติดตั้ง แต่ไม่มีความต้องการเพิ่มเติมจาก User
2	การติดตั้งมีการกำหนดจาก User มีคู่มือในการติดตั้งและปรับเปลี่ยนโดยคู่มือที่จัดทำนั้นต้องผ่านการตรวจสอบและจะไม่คำนึงถึงผลกระทบจากการติดตั้ง
3	การติดตั้งมีการกำหนดจาก User มีคู่มือในการติดตั้งและปรับเปลี่ยนโดยคู่มือที่จัดทำนั้นต้องผ่านการตรวจสอบและคำนึงถึงผลกระทบจากการติดตั้ง
4	เพิ่มจากระดับของปัจจัยที่ 2 โดยมีระบบติดตั้งและเปลี่ยนแปลงอัตโนมัติ
5	เพิ่มจากระดับของปัจจัยที่ 3 โดยมีระบบติดตั้งและเปลี่ยนแปลงอัตโนมัติ

ตารางที่ 2.16 แสดงการกำหนดระดับปัจจัย Operation Ease (Longstreet 2004)

12. Operation Ease จะมองในเรื่องของประสิทธิภาพหรือระบบอัตโนมัติในการ Start Up Back Up และขั้นตอนการ Recovery และมีการทดสอบอย่างมีประสิทธิภาพ โดยต้องมีคู่มือประกอบซอฟต์แวร์ที่ได้จัดทำขึ้น (Manual)	
0	ไม่มีลักษณะพิเศษในการจัดทำ Start Up Back Up และ Recovery
1	มีลักษณะพิเศษที่มีประสิทธิภาพในการจัดทำ Start Up Back Up และ Recovery
2	
3	
4	
5	ไม่ต้องมีผู้จัดการซอฟต์แวร์ โดยซอฟต์แวร์นั้นสามารถจัดการเองได้ รวมทั้งสามารถย้อนเพื่อแก้ไขข้อผิดพลาด

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ 2.17 แสดงการกำหนดระดับปัจจัย Multiple Sites (Longstreet 2004)

13. Multiple Sites จะมองในเรื่องของซอฟต์แวร์ว่าสามารถพัฒนาและสนับสนุนการติดตั้ง ใช้งานได้หลายพื้นที่ หรือหลายองค์กร	
0	User ไม่มีความต้องการพิเศษในการทำ Multiple Site
1	ความจำเป็นที่จะต้องมี Multiple Site หรือไม่นั้นต้องมีการพิจารณา ซึ่งถ้าเป็นแบบ Multiple Site แล้วนั้น ซอฟต์แวร์จะถูกพัฒนาภายใต้ลักษณะของ ฮาร์ดแวร์และซอฟต์แวร์เดียวกัน
2	ความจำเป็นที่จะต้องมี Multiple Site หรือไม่นั้นต้องมีการพิจารณา ซึ่งถ้าเป็นแบบ Multiple Site แล้วนั้น ซอฟต์แวร์จะถูกพัฒนาภายใต้ลักษณะของ ฮาร์ดแวร์และซอฟต์แวร์ที่เหมือนกัน
3	ความจำเป็นที่จะต้องมี Multiple Site หรือไม่นั้นต้องมีการพิจารณา ซึ่งถ้าเป็นแบบ Multiple Site แล้วนั้น ซอฟต์แวร์จะถูกพัฒนาภายใต้ลักษณะของ ฮาร์ดแวร์และซอฟต์แวร์ที่แตกต่างกัน
4	เอกสารประกอบได้มีการจัดการทดสอบและสนับสนุนซอฟต์แวร์แบบ Multiple Sites ภายใต้ระดับของปัจจัยที่ 1 หรือ 2
5	เอกสารประกอบได้มีการจัดการทดสอบและสนับสนุนซอฟต์แวร์แบบ Multiple Sites ภายใต้ระดับของปัจจัยที่ 3

ตารางที่ 2.18 แสดงการกำหนดระดับปัจจัย Facilitate change (Longstreet 2004)

14. Facilitate change จะมองในเรื่องของซอฟต์แวร์ว่าสามารถออกแบบและพัฒนาเพื่อสามารถเปลี่ยนซอฟต์แวร์ได้ตามความสะดวกหรือไม่

- การ Query ข้อมูลที่ยืดหยุ่นและสิ่งที่ช่วยออกแบบรายงานให้ง่ายขึ้น โดยสามารถดึงข้อมูลอย่างง่ายๆ ที่ต้องการได้ เช่น ตรรกะแบบ And/Or ซึ่งใช้งานกับ 1 Internal Logical File
- การ Query ข้อมูลที่ยืดหยุ่นและสิ่งที่ช่วยออกแบบรายงานให้ง่ายขึ้น โดยสามารถดึงข้อมูลที่ซับซ้อนที่ต้องการได้ เช่น ตรรกะแบบ And/Or ซึ่งใช้งานมากกว่า 1 Internal Logical File
- การ Query ข้อมูลที่ยืดหยุ่นและสิ่งที่ช่วยออกแบบรายงานให้ง่ายขึ้น โดยสามารถดึงข้อมูลที่ซับซ้อนได้ เช่น ตรรกะแบบ And/Or ซึ่งใช้งานได้มากกว่า 1 Internal Logical File
- ข้อมูลที่ใช้ในการควบคุมทางธุรกิจนั้นจะถูกเก็บไว้ในตารางโดย User จะทำการ Maintain โดยการ Online Interactive Processes การเปลี่ยนแปลงใดๆ ที่เกิดขึ้นนั้นจะกระทบกับวันที่มาทำธุรกิจในวันถัดไปได้

	- ข้อมูลที่ใช้ในการควบคุมทางธุรกิจนั้นจะถูกเก็บไว้ในตารางโดย User จะทำการ Maintain โดยการ Online Interactive Processes การเปลี่ยนแปลงใดๆ ที่เกิดขึ้นจะส่งผลทันทีทันใด
0	ไม่มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน
1	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 1 ข้อ
2	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 2 ข้อ
3	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 3 ข้อ
4	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 4 ข้อ
5	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 5 ข้อ

เมื่อทำการพิจารณาเกณฑ์ในการกำหนดอิทธิพลของปัจจัยทั้ง 14 ด้านจากตาราง 2.5-2.19 แล้วจึงทำการกรอกค่าลงในตาราง 2.19 เพื่อทำการคำนวณค่าตัวแปรปรับค่า (VAF) จากสูตร

VAF = 0.65+ [0.01* ผลรวมปัจจัย 14 ด้าน]

ตารางที่ 2.19 แสดงการคำนวณปัจจัยคุณลักษณะทั่วไปของระบบ

	คุณลักษณะ	**ระดับอิทธิพล**		**คุณลักษณะ**	**ระดับอิทธิพล**
1	Data Communication		8	Online Update	
2	Distribution Data Processing		9	Complex Processing	
3	Performance		10	Reusability	
4	Heavily Used Configuration		11	Installation Ease	
5	Transaction Rate		12	Operational Ease	
6	On-line Data Entry		13	Multiple Sites	
7	End-User Efficiency		14	Facilitate change	
				รวมค่า ปัจจัย 1-14	

2.1.3. คำนวณค่าฟังก์ชั่นพอยต์ที่ปรับค่าแล้ว

เมื่อนำค่า UFP และ VAF ที่หาได้มาคูณกัน จะทำให้ได้ค่าฟังก์ชั่นพอยต์ที่ได้รับการปรับค่าตามสูตร

FP = UFP * VAF

การประมาณขนาดซอฟต์แวร์ด้วยวิธีฟังก์ชั่นพอยต์เป็นวิธีที่ได้รับการยอมรับอย่างแพร่หลาย โดยประมาณการจากการนับฟังก์ชั่นการทำงานในมุมมองของผู้ใช้งานซึ่งทำให้สามารถทำการประมาณการได้ในช่วงต้นของการพัฒนา (Choi, Park,and Sugumaran, 2006) โดยในเวลาต่อมาวิธีฟังก์ชั่นพอยต์ ได้ถูกนำไปปรับปรุง พัฒนาต่อยอดเป็นวิธีการประมาณการอื่นๆหลายวิธี ทั้งนี้เหตุผลที่ทำให้วิธีการนี้ประสบความสำเร็จนอกจากการที่สามารถทำการประมาณการได้ในช่วงต้นของการพัฒนาแล้ว วิธีการประมาณยังไม่ขึ้นกับเทคนิคทางการโปรแกรมหรือภาษาที่ใช้พัฒนาแต่อย่างใด (ชมพูนุช เผ่าประพัธน์,2556)

2.2 การประมาณขนาดซอฟต์แวร์ด้วยวิธีอ็อบเจกต์พอยต์ (Object Points)

เป็นทางเลือกหนึ่งสำหรับวิธีการประมาณการซอฟต์แวร์ ที่สามารถนำมาใช้ได้ตลอดวงจรพัฒนาซึ่งเป็นวิธีที่ง่ายต่อการคำนวณ โดยพิจารณาจากส่วนที่ซอฟต์แวร์ติดต่อกับผู้ใช้ (User Interface) อ็อบเจกต์พอยต์ในที่นี้ไม่ได้ หมายถึง อ็อบเจกต์ ในเชิงโปรแกรม (Class) (Issa et al, 2007) แต่จำนวนของ อ็อบเจกต์ นั้นคำนวณได้จาก

- จำนวนของหน้าจอที่แสดงผลแยกออกจากกัน
- จำนวนของรายงานที่สร้างโดยระบบ
- จำนวนโมดูลที่พัฒนาโดยภาษายุคที่ 3 (3rd Generation Language) หรือ 3 จีแอลขึ้นไป ที่ต้องพัฒนาเพื่อสนับสนุนการทำงานของฐานข้อมูล โดย 3 จีแอล เป็นภาษาระดับสูงที่มีลักษณะการเขียนเป็นประโยค และใกล้เคียงกับภาษามนุษย์มากขึ้น ตัวอย่างภาษาในยุคนี้ เช่น ภาษาเบสิค(BASIC) ภาษาปาสคาล (PASCAL) ภาษาโคบอล(COBOL) ภาษาซี(C) เป็นต้น

แล้วจึงนำข้อมูลเหล่านี้ไปคำนวณหาค่าอ็อบเจกต์พอยต์ (Object Points) จากสมการ

$$\mathbf{NOP} = \mathbf{(Object\ Points)} \times \frac{\mathbf{(100 - \%reuse)}}{\mathbf{100}}$$

โดยที่

NOP คือ ค่าอ็อบเจกต์พอยต์ใหม่

Object Points คือ ผลรวมค่าอ็อบเจกต์พอยต์ของแต่ละอ็อบเจกต์

% reuse คือ เปอร์เซ็นต์ของอ็อบเจกต์ที่มีการนำมาใช้

ขั้นตอนในการประมาณการขนาดด้วย อ็อบเจกต์พอยต์ มีดังนี้

2.2.1. การหาจำนวนอ็อบเจกต์

ขั้นตอนนี้จะทำการนับจำนวนองค์ประกอบของแอพพลิชั่น ตามหลักการของอ็อบเจกต์พอยต์ คือ จำนวนหน้าจอ จำนวนรายงาน จำนวนโมดูล 3 จีแอล

2.2.2. การจัดกลุ่มให้กับอ็อบเจกต์

เมื่อได้จำนวนของอ็อบเจกต์จากขั้นที่ 1 แล้ว ก็จะนำ อ็อบเจกต์ ที่เป็นหน้าจอ และรายงาน มาแบ่งออกเป็นกลุ่มๆ คือ ง่าย (Simple) ปานกลาง (Medium) และยาก (Difficult) โดยพิจารณาจากเงื่อนไขตามตารางที่ 2.20 ส่วนอ็อบเจกต์ ที่เป็น โมดูล 3 จีแอล จะพิจารณาให้เป็น กลุ่มยาก ทั้งหมด

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 2.20 การแบ่งกลุ่มอ็อบเจกต์ (Issa et al., 2007)

หน้าจอ (Screen)				**รายงาน (Report)**			
จำนวนหน้าจอที่ปรากฏ (Number of Views Contained)	**จำนวนของแหล่งข้อมูลตาราง (Source of Data Tables)**			**จำนวนส่วนของรายงาน (Number of Sections Contained)**	**จำนวนของแหล่งข้อมูลตาราง (Source of Data Tables)**		
	<4 (<2 Srvr <3 Clnt)	**≤8 (2/3 Srvr 3-5 Clnt)**	**>8 (>3 Srvr >5 Clnt)**		**<4 (<2 Srvr <3 Clnt)**	**≤8 (2/3 Srvr 3-5 Clnt)**	**>8 (>3 Srvr >5 Clnt)**
<3	ง่าย	ง่าย	ปานกลาง	0 หรือ 1	ง่าย	ง่าย	ปานกลาง
3-7	ง่าย	ปานกลาง	ยาก	2 หรือ 3	ง่าย	ปานกลาง	ยาก
≥8	ปานกลาง	ยาก	ยาก	>3	ปานกลาง	ยาก	ยาก

หมายเหตุ srvr หมายถึง เครื่องผู้ให้บริการ (Server) clnt หมายถึง เครื่องผู้รับบริการ (Client)

2.2.3. การให้ค่าน้ำหนักแก่อ็อบเจกต์ที่แบ่งกลุ่มแล้ว

โดยนำอ็อบเจกต์พอยต์ที่ได้รับการแบ่งกลุ่มแล้วมาคำนวณค่าน้ำหนัก โดยค่าน้ำหนักจะขึ้นอยู่กับกลุ่มของอ็อบเจกต์พอยต์ คือ ง่าย ปานกลาง ยาก ตามตารางที่ 2.21

ตารางที่ 2.21 แสดงค่าความน้ำหนักของวิธีอ็อบเจกต์พอยต์

ประเภทของอ็อบเจกต์	**ง่าย**	**ปานกลาง**	**ยาก**
หน้าจอ (Screen)	1	2	3
รายงาน (Report)	2	5	8
โมดูล 3 จีแอล (3GL)	-	-	10

ตัวอย่างการพิจารณาค่าอ็อบเจกต์พอยต์

แอพพลิเคชั่น เอ (ชื่อสมมติ) เป็นแอพพลิเคชั่นสำหรับทำรายการจองตั๋วเครื่องบิน มีจำนวน 3 หน้าจอ 1 หน้าจอรายงาน และ 3 ตารางข้อมูล ซึ่งมีรายละเอียดดังนี้

- หน้าจอที่ 1 หน้าจอสำหรับจองเที่ยวบิน
  มีข้อมูลที่เกี่ยวข้อง 3 ตาราง คือ ตาราง Customer info ,Customer History, Available Seat และสามารถทำงานรายการเสร็จสิ้นใน 1 หน้าจอ
- หน้าจอที่ 2 หน้าจอแสดงราคาเที่ยวบินที่เลือก
  มีข้อมูลที่เกี่ยวข้อง 2 ตาราง คือ Available Seat, Flight Plan และสามารถทำงานรายการเสร็จสิ้นใน 1 หน้าจอ
- หน้าจอที่ 3 หน้าจอแสดงการค้นหาเที่ยวบิน
  มีข้อมูลที่เกี่ยวข้อง 2 ตาราง คือ Available Seat, Flight Plan และสามารถทำงานรายการเสร็จสิ้นใน 1 หน้าจอ
- หน้ารายงานสรุปการขายประจำเดือน สำหรับแสดงรายการขายประจำเดือน หรือประจำปี มีข้อมูลที่เกี่ยวข้อง 3 ตาราง และมี 4 ส่วนของรายงาน

จากข้อมูลข้างต้น สามารถแสดงวิธีการคำนวณค่าอ็อบเจกต์พอยต์ของโมบายแอพพลิเคชั่นนี้ได้ ดังตารางที่ 2.22

ตารางที่ 2.22 แสดงตัวอย่างการคำนวณค่าอ็อบเจกต์พอยต์

ชื่ออ็อบเจกต์	ประเภท	กลุ่มอ็อบเจกต์	ค่าน้ำหนักอ็อบเจกต์
จองเที่ยวบิน	หน้าจอ	ง่าย	1
แสดงราคา	หน้าจอ	ง่าย	1
ค้นหาเที่ยวบิน	หน้าจอ	ง่าย	1
รายงานการขาย	รายงาน	ปานกลาง	5
		รวม	9

วิธีการนี้เหมาะกับการพัฒนาที่ใช้เครื่องมือระดับสูงในการพัฒนา และการพิจารณาประมาณค่าสามารถทำได้จากเอกสารความต้องการซอฟต์แวร์ ซึ่งไม่ต้องใช้รายละเอียดในการพัฒนามาคิดคำนวณ (Malathi And Sridhar, 2012) ซึ่งวิธีนี้ให้ผลลัพธ์ใกล้เคียงกันกับวิธีฟังก์ชั่นพอยต์ข้อดีคือใช้เวลาน้อยกว่า

อ็อบเจกต์พอยต์จึงเป็นอีกทางเลือกนอกเหนือจากฟังก์ชั่นพอยต์มีความเหมาะสมกับแอพพลิเคชั่นที่ใช้เครื่องมือระดับสูงในการพัฒนา และเนื่องจากไม่จำเป็นต้องใช้รายละเอียดในการพัฒนามาเป็นข้อมูลประกอบการประมาณการทำให้วิธีการนี้เหมาะสมในการใช้ประมาณการในช่วงต้นของการพัฒนา(Malathi and Sridhar,2012)

2.3 การประมาณขนาดซอฟต์แวร์ด้วยวิธียูสเคสพอยต์ (Use Case Point : UCP)

ยูสเคสพอยต์เป็นวิธีการประมาณการซอฟต์แวร์ที่จะถูกนำมาใช้ในโครงการพัฒนาซอฟต์แวร์ที่มีการสร้างเอกสารแบบจำลองยูเอ็มแอล(UML) ซึ่งเป็นวิธีการที่ใช้ในกระบวนการออกแบบและพัฒนา

โดยแนวคิดของยูสเคสพอยต์จะขึ้นอยู่กับ ความต้องการของระบบ ที่ถูกเขียนออกมาในรูปแบบแผนภาพยูสเคส (Use Case) โดยการคำนวณยูสเคสพอยต์ของโครงการพัฒนาซอฟต์แวร์จะคำนวณจากส่วนต่างๆได้แก่

1. ค่าน้ำหนักยูสเคสที่ยังไม่ปรับค่า (Unadjusted Use Case Weight: UUCW)
2. ค่าน้ำหนักผู้กระทำการที่ยังไม่ปรับค่า (Unadjusted Actor Weight: UAW)
3. ปัจจัยความซับซ้อนเชิงเทคนิค (Technical Complexity Factor: TCF)
4. ปัจจัยความซับซ้อนทางสภาพแวดล้อม (Environmental Complexity Factor: ECF)

ขั้นตอนการประมาณการซอฟต์แวร์โดยใช้ยูสเคสพอยต์

2.3.1. ค่าน้ำหนักยูสเคสที่ยังไม่ปรับค่า (UUCW)

คำนวณจากการกำหนดค่าน้ำหนักความซับซ้อนให้แต่ละยูสเคส โดยให้ขึ้นกับจำนวนรายการ (transaction) ในยูสเคสนั้น ซึ่งรายการก็คือ ขั้นตอนการทำงานต่างๆของแต่ละยูสเคส โดยให้นับรวมส่วนต่อขยาย(Extension) ด้วย ซึ่งจะต้องระมัดระวังในการนับส่วนต่อขยายเพราะไม่สามารถนับแค่จำนวนเส้นได้ เนื่องจากส่วนต่อขยายบางตัวไม่ใช่รายการแต่เป็นผลการทำงานของสถานการณ์(Scenario)หลัก หรือบางกรณีที่ส่วนต่อขยายทำงานเหมือนกันแต่ใช้ชื่อต่างกัน โดยแต่ละยูสเคสจะสามารถกำหนดระดับความซับซ้อนโดยแบ่งออกเป็น 3 ระดับคือ ง่าย(Simple) ปานกลาง(Average) และ ซับซ้อน(Complex) ซึ่งค่าน้ำหนักของแต่ละระดับจะถูกกำหนดไว้แล้ว

ตารางที่ 2.23 แสดงค่าถ่วงน้ำหนักของยูสเคส ซึ่งพิจารณาจากจำนวนรายการ(Ribu,2001)

ระดับความซับซ้อนของยูสเคส	จำนวนรายการ	ค่าน้ำหนัก
ง่าย	≤ 3	5
ปานกลาง	4 - 7	10
ซับซ้อน	≥ 8	15

โดยค่าน้ำหนักยูสเคสที่ยังไม่ปรับค่าจะคำนวณได้จากสมการ

UUCW = (Total No. of Simple Use Cases x 5) + (Total No. Average Use Case x 10) + (Total No. Complex Use Cases x 15)

2.3.2. ค่าน้ำหนักผู้กระทำการที่ยังไม่ปรับค่า (UAW)

ค่าน้ำหนักผู้กระทำการที่ยังไม่ปรับค่าก็เป็นอีกปัจจัยที่มีผลต่อขนาดของซอฟต์แวร์ โดยการคำนวณจำนวนและความซับซ้อนของผู้ร้องขอบริการจากระบบหรือผู้กระทำการ (Actor) จะมีวิธีการคล้ายคลึงกับการคำนวณจำนวนและความซับซ้อนของยูสเคส โดยจะจำแนกผู้กระทำการของระบบออกเป็น 3 ระดับ คือ ง่าย (Simple) ปานกลาง (Average) และ ซับซ้อน (Complex) เช่นเดียวกัน แต่หลักการจำแนกและค่าน้ำหนักของแต่ละระดับจะมีความแตกต่างกัน

ตารางที่ 2.24 แสดงค่าถ่วงน้ำหนักของผู้กระทำการ(Ribu, 2001)

ระดับผู้กระทำการ	รายละเอียด	ค่าน้ำหนัก
ง่าย	ระบบภายนอกที่จะต้องติดต่อกับระบบผ่านทาง API	1
ธรรมดา	ระบบภายนอกที่จะต้องติดต่อกับระบบผ่านทางโปรโตคอล (e.g. TCP/IP, FTP, HTTP, database) ผู้ใช้งานที่ใช้งานระบบผ่าน text-based user interface	2
ซับซ้อน	ผู้ใช้งานที่ใช้งานระบบผ่านทาง GUI	3

โดยค่าน้ำหนักผู้กระทำการที่ยังไม่ปรับค่าจะคำนวณได้จากสมการ

UAW = (Total No. of Simple actors x 1) + (Total No. Average actors x 2) + (Total No. Complex actors x 3)

2.3.3. ค่าปัจจัยความซับซ้อนเชิงเทคนิค (TCF)

ค่าปัจจัยความซับซ้อนเชิงเทคนิคที่มีอิทธิพลต่อการพัฒนาซอฟต์แวร์ ซึ่งจะใกล้เคียงกับปัจจัยความซับซ้อนของระบบในวิธีฟังก์ชั่นพอยต์แต่จะมีการให้ค่าน้ำหนักของแต่ละปัจจัยที่แตกต่างกันที่แตกต่างกัน จะทำการคำนวณจากการกำหนดคะแนนให้แก่ ปัจจัยทางเทคนิคทั้ง 13 ปัจจัย โดยระดับอิทธิพลจะอยู่ในช่วง 0 (ไม่มีอิทธิพล) ถึง 5 (มีอิทธิพลสูงที่สุด) คะแนนนี้จะถูกคูณด้วยค่าถ่วงน้ำหนักของแต่ละปัจจัย เพื่อให้ได้ค่าผลรวมของปัจจัย (Total Factor) จากนั้นจะนำค่าผลรวมของปัจจัยที่ได้มาคำนวณหาค่าปัจจัยความซับซ้อนเชิงเทคนิค

ตารางที่ 2.25 แสดงค่าถ่วงน้ำหนักของปัจจัยที่มีผลต่อความซับซ้อนทางด้านเทคนิค(Ribu, 2001)

Technical Factor	Factor Description	Weight Factor	Score	Weight x Score
T1	Distributed System	2		
T2	Response adjectives	1		
T3	End-user efficiency	1		
T4	Complex processing	1		
T5	Reusable code	1		
T6	Easy to install	0.5		
T7	Easy to use	0.5		
T8	Portable	2		
T9	Easy to change	1		
T10	Concurrent	1		
T11	Security features	1		
T12	Access for third-parties	1		
T13	Special training required	1		
			Total	

โดยค่าปัจจัยความซับซ้อนเชิงเทคนิคจะคำนวณได้จากสมการ

$$TCF = 0.6 + (TF/100)$$

2.3.4. ค่าปัจจัยความซับซ้อนทางสภาพแวดล้อม (ECF)

ค่าปัจจัยความซับซ้อนทางสภาพแวดล้อมของการพัฒนาที่มีผลต่อการพัฒนาซอฟต์แวร์ มีวิธีการคำนวณก็จะคล้ายคลึงกับการหาค่าปัจจัยความซับซ้อนเชิงเทคนิค โดยจะต้องทำการให้คะแนนกับปัจจัยความซับซ้อนทางสภาพแวดล้อมแต่ละตัว โดยมีระดับของอิทธิพลตั้งแต่ 0 – 5 ตามคำอธิบายในตาราง 2.26-2.33

ตารางที่ 2.26 แสดงการกำหนดระดับปัจจัย Familiar with development method used (Ribu, 2001)

1.Familiar with development method used เป็นการวัดประสบการณ์ของทีมงานเกี่ยวกับกระบวนการพัฒนาที่ใช้ในโครงการ	
0	ทีมงานไม่รู้จักกระบวนการพัฒนาที่ใช้ในโครงการนี้
1	ทีมงานมีความรู้ในเชิงทฤษฎีเกี่ยวกับกระบวนการพัฒนาที่ใช้ในโครงการนี้
2	สมาชิกในทีมอย่างน้อยหนึ่งคนเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้เพียงครั้งเดียวหรือไม่กี่ครั้ง
3	สมาชิกของทีมอย่างน้อยครึ่งหนึ่งเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้ในหลายโครงการ
4	สมาชิกของทีมอย่างน้อยหนึ่งคนเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้ในหลายโครงการ
5	ทีมงานทั้งหมดมีเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้ในหลายโครงการ

ตารางที่ 2.27 แสดงการกำหนดระดับปัจจัย Application experience (Ribu, 2001)

2. Application experience เป็นการวัดประสบการณ์ของทีมงานเกี่ยวกับประเภทของโมบายแอพพลิเคชั่นที่กำลังจะพัฒนา หรือประสบการณ์ในการพัฒนาแอพพลิเคชั่นที่หลากหลายประเภท	
0	ทุกคนในทีมเป็นมือใหม่
1 หรือ 2	บางคนในทีมมีประสบการณ์เล็กน้อย 1 ปีถึง 1 ปี ครึ่ง สมาชิกที่เหลือเป็นมือใหม่
3	ทุกคนในทีมมีประสบการณ์มากกว่า 1 ปีครึ่ง
4	สมาชิกในทีมส่วนใหญ่มีประสบการณ์ 2 ปี
5	สมาชิกทุกคนมีประสบการณ์ 2 ปี

ตารางที่ 2.28 แสดงการกำหนดระดับปัจจัย Object-Oriented experience (Ribu, 2001)

3.Object-Oriented experience เป็นการวัดประสบการณ์ของทีมงานเกี่ยวกับการวิเคราะห์และออกแบบเชิงวัตถุ เช่นการสร้างแบบจำลองยูสเคส(Use Case modeling)ในขั้นตอนการวิเคราะห์ และการสร้างแบบจำลองคอมโพเน้นท์(Component modeling)ในขั้นตอนการออกแบบ	
0	ทีมงานไม่มีความคุ้นเคยกับการวิเคราะห์และออกแบบเชิงวัตถุเลย
1	ทุกคนในทีมมีประสบการณ์น้อยกว่า 1 ปี
2 หรือ 3	ทุกคนในทีมมีประสบการณ์ 1 ปี ถึง 1 ปี ครึ่ง
4	สมาชิกส่วนใหญ่ในทีมมีประสบการณ์มากกว่า 2 ปี
5	สมาชิกของทีมอย่างน้อยหนึ่งคนเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้ในหลายโครงการ

CHULALONGKORN UNIVERSITY

ตารางที่ 2.29 แสดงการกำหนดระดับปัจจัย Lead analyst capability (Ribu, 2001)

4. Lead analyst capability เป็นการวัดประสบการณ์ในการวิเคราะห์ความต้องการและสร้างแบบจำลอง	
0	นักวิเคราะห์เป็นมือใหม่
1 หรือ 2	นักวิเคราะห์มีประสบการณ์ไม่กี่โครงการ
3 หรือ 4	นักวิเคราะห์มีประสบการณ์อย่างน้อย 2 ปี จากหลายโครงการ
5	นักวิเคราะห์มีประสบการณ์อย่างน้อย 3 ปี กับรูปแบบโครงการที่หลากหลาย

ตารางที่ 2.30 แสดงการกำหนดระดับปัจจัย Motivation (Ribu, 2001)

5. Motivation ปัจจัยนี้เป็นการอธิบายถึงแรงจูงใจของทีมงานโดยรวม	
0	ไม่มีแรงจูงใจ
1 หรือ 2	มีแรงจูงใจเล็กน้อย
3 หรือ 4	ทีมงานมีแรงจูงใจในการดำเนินงานที่ดี
5	ทีมงานมีแรงจูงใจและแรงกระตุ้นอย่างมาก

ตารางที่ 2.31 แสดงการกำหนดระดับปัจจัย Stable requirements (Ribu, 2001)

6. Stable requirements ปัจจัยนี้เป็นการวัดระดับของการเปลี่ยนแปลงความต้องการและความไม่ชัดเจนของความต้องการ	
0	ความต้องการไม่แน่นอนและมีการเปลี่ยนแปลงความต้องการอย่างต่อเนื่อง
1 หรือ 2	ความต้องการไม่แน่นอน ลูกค้ามีการเปลี่ยนแปลงความต้องการในหลายช่วงของโครงการ
3 หรือ 4	มีการเปลี่ยนแปลงความต้องการเล็กน้อยเท่าที่จำเป็น
5	ความต้องการมีเสถียรภาพตลอดช่วงเวลาของโครงการ

CHULALONGKORN UNIVERSITY

ตารางที่ 2.32 แสดงการกำหนดระดับปัจจัย Part-time workers (Ribu, 2001)

7. Part-time workers มีการสอนงานสมาชิกใหม่ของทีมหรือหมายถึงสมาชิกของทีมไม่แน่นอน ซึ่งส่งผลต่อความสามารถในการพัฒนาซอฟต์แวร์(Productivity)ในทางลบ	
0	ไม่มีสมาชิกในทีมที่ทำงานแบบพาร์ทไทม์(Part-Time)
1 หรือ 2	บางส่วนของทีม (ประมาณ 20 เปอร์เซ็น) เป็นพนักงานแบบพาร์ทไทม์(Part-time)
3 หรือ 4	ครึ่งหนึ่งของทีมงานเป็นพนักงานแบบพาร์ทไทม์(Part-time)
5	ทุกคนในทีมเป็นพนักงานแบบพาร์ทไทม์(Part-time)

ตารางที่ 2.33 แสดงการกำหนดระดับปัจจัย Difficult programming language (Ribu, 2001)

8. Difficult programming language ปัจจัยนี้บ่งบอกถึงความเชี่ยวชาญของทีมงานในการใช้เครื่องมือและภาษาของการพัฒนาที่ใช้ในโครงการนี้	
0	ทุกคนในทีมมีประสบการณ์ในการเขียนโปรแกรมมากกว่า 2 ปี
1	สมาชิกทีมส่วนใหญ่มีประสบการณ์ในการเขียนโปรแกรมมากกว่า 2 ปี
2	ทุกคนในทีมมีประสบการณ์ในการเขียนโปรแกรมมากกว่า 1 ปีครึ่ง
3	สมาชิกในทีมส่วนใหญ่มีประสบการณ์ในการเขียนโปรแกรมมากกว่า 1 ปี
4	สมาชิกในทีมบางคนมีประสบการณ์ในการเขียนโปรแกรมประมาณ 1 ปี ที่เหลือเป็นมือใหม่
5	ทุกคนในทีมเป็นมือใหม่

หลังจากนั้นจึงกรอกระดับอิทธิพลของแต่ละปัจจัยลงในตารางตามตัวอย่างในตาราง 2.34 และทำการคำนวณค่าผลรวมของปัจจัยความซับซ้อนทางสภาพแวดล้อม

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 2.34 แสดงค่าถ่วงน้ำหนักของผลกระทบทางสภาพแวดล้อมของการพัฒนา

Experience Factor	Factor Description	Weight Factor	Score	Weight x Score
E1	Familiar with development method used	1.5		
E2	Application experience	0.5		
E3	Object-Oriented experience	1		
E4	Lead analyst capability	0.5		
E5	Motivation	1		
E6	Stable Requirements	2		
E7	Part-time workers	-1		
E8	Difficulty of programming language	-1		
Total EF				

โดยค่าปัจจัยความซับซ้อนทางสภาพแวดล้อมจะคำนวณได้จากสมการ

$$ECF = 1.4 + (-0.03 \times EF)$$

2.3.5. ค่ายูสเคสพอยต์ (UCP)

หลังจากทำการหาค่า UUCP ,UAW ,TCF,ECF แล้วจะทำการคำนวณหา UCP ได้จากสมการ

$$UCP = (UUCW + UAW) \times TCF \times ECF$$

เมื่อได้จำนวนยูสเคสพอยต์ของระบบแล้วจะสามารถประมาณการค่าความพยายามสำหรับระบบได้โดยนำค่ายูสเคสพอยต์มาคูณกับเวลาที่ใช้ต่อ 1 ยูสเคสพอยต์ ตามสมการนี้

$$Effort\ (Man - Hour) = UCP \times Hours/UCP$$

ข้อดีของวิธีการนี้คือในปัจจุบันมีเครื่องมือที่สามารถคำนวณค่ายูสเคสพอยต์ ให้อัตโนมัติทำให้ร่นระยะเวลาการคำนวณไปได้มาก และผลจากการประมาณขนาดของโปรแกรมจะเป็นอิสระจากทักษะและประสบการณ์ของทีมงานที่ทำการพัฒนา

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ส่วนข้อด้อยของวิธีการนี้คือจะสามารถประมาณการได้อย่างดีก็ต่อเมื่อเขียนยูสเคส ในระดับยูสเซอร์โกล(Use Goal) หรือระดับปฏิสัมพันธ์ (Interaction level) ขึ้นไป และไม่สามารถประมาณการได้จนกว่ายูสเคส ทั้งหมดจะถูกเขียน ซึ่งการเขียนยูสเคส ในระดับยูสเซอร์โกลเพื่ออธิบายถึงปฏิสัมพันธ์ที่เกิดขึ้นระหว่างผู้กระทำการและระบบ จะทำให้โครงการพัฒนาซอฟต์แวร์เกิดค่าพยายามมากขึ้น 10-20% เนื่องจากการเขียนยูสเคสใช้เวลาค่อนข้างมากทำให้การประมาณการจะล่าช้า และหากเขียนยูสเคส ขึ้นมาก่อนโดยไม่ได้วิเคราะห์การทำงานอย่างละเอียดจะทำให้ยูสเคส ที่นำมาประมาณไม่ใช่ยูสเคส ที่ใช้จริงในการพัฒนา

2.4 วิธีการประมาณการค่าความพยายามที่นำเสนอโดย Nikunj Sakhrelia นักพัฒนาโมบายแอพพลิเคชั่นอิสระ

จากการศึกษาวิธีการประมาณการขนาดของโมบายแอพพลิเคชั่นยังไม่พบวิธีการที่เฉพาะเจาะจงเพื่อนำมาเปรียบเทียบกับวิธีการในการประมาณอื่น โดยบริษัทที่รับพัฒนาโมบายแอพพลิเคชั่นจะมีวิธีการประมาณการเป็นของตัวเองซึ่งเป็นความลับทำให้ไม่สามารถเข้าถึงวิธีการเหล่านั้น แต่ทั้งนี้จากการค้นคว้าทางอินเตอร์เน็ตพบว่า มีเหล่านักพัฒนาโมบายแอพพลิเคชั่นอิสระที่ได้มีการนำเสนอวิธีในการประมาณการอย่างง่าย

ในปี ค.ศ. 2012 หนึ่งในวิธีที่น่าสนใจถูกเสนอโดย Nikunj Sakhrelia นักพัฒนาโมบายแอพพลิเคชั่นจากประเทศอินเดียผ่านทางเว็บไซต์แอพเซเลเรเตอร์ (Sakhrelia, 2012 : Online) ซึ่งวิธีการที่นำเสนอจะทำการสร้างแบบจำลองหน้าจอจากข้อมูลฟังก์ชั่นการทำงานในเอกสารความต้องการ จากนั้นจะแบ่งฟังก์ชั่นการทำงานและองค์ประกอบบนหน้าจอของโมบายแอพพลิเคชั่นออกเป็น 3 ประเภท คือ ง่าย ปานกลาง ซับซ้อน ตามคำอธิบายของ Nikunj Sakhrelia ในตาราง 2.35

Chulalongkorn University

ตารางที่ 2.35 แสดงการแบ่งฟังก์ชั่นการทำงานและองค์ประกอบบนหน้าจอตามวิธีของ Nikunj Sakhrelia

ระดับความซับซ้อน	รายละเอียดฟังก์ชั่นการทำงานและ องค์ประกอบบนหน้าจอของโมบายแอพพลิเคชั่น	ค่าน้ำหนัก (คน - ชั่วโมง)
ง่าย	การแสดงข้อมูลจากแหล่งข้อมูลภายใน หน้าจอแสดงผลพร้อมปุ่มหรือเมนูอย่างง่าย การใช้งานเอพีไอที่ไม่ต้องปรับแต่ง เข้าสู่ระบบ ลืมรหัสผ่าน การแสดงข้อมูลจากเว็บไซต์โดยใช้ Rssfeed การแสดงข้อมูลทวิตเตอร์(Twitter)ผ่าน Twitter feed การแสดงข้อมูลวิดีโอจากเว็บไซต์ยูทูป(Youtube) ผ่าน Youtube video feed แท็บเมนูที่มีการปรับแต่ง	4
ปานกลาง	มีการเรียกใช้เซอร์เวอร์เอพีไอ(Server API)	8
ซับซ้อน	ตารางแสดงข้อมูลแบบปรับแต่ง การใช้งานกล้อง การใช้งานเซนเซอร์ต่างๆของอุปกรณ์ บันทึกเสียง หรือวิดีโอ การใช้งานเอพีไอที่ต้องปรับแต่ง	16

ทั้งนี้บริษัทโฟร์ดีไซน์ บริษัทรับจ้างพัฒนาเว็บแอพพลิเคชั่น ระบบธุรกิจออนไลน์ และโมบายแอพพลิเคชั่น ที่มีสำนักงานอยู่ในสหรัฐอเมริกาและยุโรป ได้มีการใช้ฟังก์ชั่นการทำงานในการประมาณการราคาผ่านหน้าเว็บไซต์ (http://www.d4designstudios.com/mobile-app-price-estimate/) ซึ่งให้ผู้ว่าจ้างที่ต้องการประเมินราคาเบื้องต้นสามารถประมาณการราคาได้ โดยราคาประเมินก็ขึ้นอยู่กับฟังก์ชั่นการทำงานที่ต้องการ ดังนั้นจะเห็นได้ว่าส่วนที่สำคัญในการประมาณการขนาดของโมบายแอพพลิเคชั่นนั้นไม่ได้มีความแตกต่างกับหลักการของ วิธีการประมาณการซอฟต์แวร์ทั่วไป ไม่ว่าจะเป็นวิธีฟังก์ชั่นพอยต์ หรืออ็อบเจกต์พอยต์ ล้วนแต่ใช้ข้อมูลฟังก์ชั่นการทำงานจากเอกสารความต้องการเป็นข้อมูลในการประมาณราคา หากแต่ด้วยความแตกต่างในลักษณะทั่วไปของโมบายแอพพลิเคชั่น(ขนาดของโครงการ จำนวนหน้าจอ จุดประสงค์เพื่อทำงานอย่างใดอย่างหนึ่ง

ทำงานเสร็จภายในหน้าจอเดียว) ทำให้ผู้พัฒนาโมบายแอพพลิเคชั่นเลือกใช้วิธีการอย่างง่ายในการพัฒนา ทำให้เป็นที่น่าสนใจว่าวิธีการของนักพัฒนาโมบายแอพพลิเคชั่นอิสระทั่วไปจะสามารถให้ความแม่นยำในการประมาณการเทียบเท่ากับวิธีการในทางทฤษฎีและได้รับการยอมรับอย่างกว้างขวางหรือไม่

2.5 การถอด(Extract)ฟังก์ชั่นพอยต์และยูสเคสพอยต์จากข้อมูลความต้องการในรูปแบบการเขียนเป้าหมาย (Goal) และสถานการณ์ (Scenario) บนพื้นฐานของความต้องการแบบข้อความ (Text Requirement)

ในการรวบรวมข้อมูลความต้องการนั้นโดยทั่วไปมักจะใช้การเขียนในรูปแบบภาษาธรรมชาติ(Natural Language) ซึ่งก็คือการเขียนความต้องการออกมาในลักษณะข้อความ (Text Requirement) แต่ทั้งนี้ในการที่จะนำความต้องการที่เขียนในแบบข้อความ มาวิเคราะห์ฟังก์ชั่นพอยต์นั้น จะต้องมีกระบวนการในการจัดการเพื่อให้สามารถตรวจสอบย้อนกลับระหว่างความต้องการและฟังก์ชั่นพอยต์ซึ่งเป็นสิ่งที่สำคัญ เนื่องจากกระบวนการของการประมาณการโดยการประมาณการขนาดด้วยวิธีฟังก์ชั่นพอยต์นั้นไม่ได้ครอบคลุมถึงกระบวนในการสอบถามความต้องการ ในปี ค.ศ. 2006 Choi และคณะจึงได้เสนอวิธีในการถอดฟังก์ชั่นพอยต์จากข้อมูลความต้องการในรูปแบบการเขียนเป้าหมาย (Goal) และสถานการณ์ (Scenario) บนพื้นฐานของความต้องการแบบข้อความ (Text Requirement) (Choi et al., 2006)

การเขียนความต้องการในรูปแบบการเขียนเป้าหมาย (Goal) และสถานการณ์ (Scenario) นั้นเป็นวิธีที่ถูกนำไปใช้งานอย่างแพร่หลายและได้ให้นิยามเกี่ยวกับเป้าหมายและสถานการณ์ไว้ดังนี้

เป้าหมาย (Goal) หมายถึง บางสิ่งที่ผู้มีส่วนได้ส่วนเสีย (stakeholder) บางรายหวังที่จะบรรลุในอนาคต

สถานการณ์ (Scenario) หมายถึง พฤติกรรมหรือการกระทำที่เป็นไปได้ภายใต้ชุดของการปฏิสัมพันธ์ของหมู่ตัวแทน(agent)เพื่อบรรลุเป้าหมาย

สถานการณ์จะจับภาพความต้องการที่แท้จริงเพราะอธิบายถึงสถานการณ์จริงหรือพฤติกรรมที่เป็นรูปธรรม และเป้าหมายสามารถบรรลุได้ด้วยการดำเนินการ(Execution) ของสถานการณ์ ดังนั้นสถานการณ์จะอยู่ภายใต้เป้าหมายและมีความเป็นเอกลักษณ์ของแต่ละสถานการณ์ หรือในอีกด้านหนึ่งคือเป้าหมายช่วยให้ค้นพบสถานการณ์ สถานการณ์ยังสามารถช่วยในการค้นพบเป้าหมาย โดยเป้าหมาย สามารถแบ่งเป็นระดับได้ ดังนี้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ระดับธุรกิจ (Business Level)

จุดมุ่งหมายของระดับธุรกิจ คือ การระบุวัตถุประสงค์สูงสุดของระบบ เป้าหมายระดับนี้จะถูกระบุโดยองค์กรหรือผู้ใช้งานระบบ ยกตัวอย่างเช่น “การปรับปรุงการให้บริการแก่ลูกค้าธนาคารของเรา” นั่นคือเป้าหมายโดยรวมที่ถูกกำหนดขึ้นโดยธนาคาร

ระดับบริการ (Service Level)

จุดมุ่งหมายของระดับบริการ คือการระบุบริการที่ระบบควรมีให้แก่องค์กรและผู้ใช้งานในระดับนี้จะเสนอทางเลือกของสถาปัตยกรรมบริการ(Service Architecture) หลายๆแบบ ซึ่งทั้งหมดจะมีความสอดคล้องกับเป้าหมายระดับธุรกิจ

เป้าหมายและสถานการณ์ของระดับนี้จะถูกเขียนในรูปแบบคู่กัน คือ <G,Sc> เมื่อ G แทน เป้าหมาย Sc แทน สถานการณ์ เป้าหมายที่ถูกเขียนขึ้นแสดงออกถึงความเป็นไปได้อย่างใดอย่างหนึ่งที่จะตอบสนองเป้าหมายระดับธุรกิจ ตัวอย่างเช่น “ลูกค้าถอนเงินสดจากเครื่องเอทีเอ็ม ” เพื่อบรรลุเป้าหมายคือ “ถอดเงินสด”

ระดับการปฏิสัมพันธ์ (Interaction Level)

ที่ระดับนี้จะสนใจการปฏิสัมพันธ์ระหว่างระบบและตัวแทน(Agent) ซึ่งเป้าหมายและสถานการณ์ของระดับนี้จะถูกเขียนในรูปแบบคู่กัน คือ <G,Sc> เมื่อ G แทน เป้าหมาย Sc แทน สถานการณ์ การปฏิสัมพันธ์นี้จำเป็นสำหรับการบรรลุการบริการที่ถูกกำหนดในระดับบริการ

ตัวอย่างเช่น เป้าหมายระดับบริการ “ถอนเงินสดจากเอทีเอ็ม” แสดงถึงลักษณะของการให้บริการ สถานการณ์ระดับปฏิสัมพันธ์จะอธิบายถึงการไหลของปฏิสัมพันธ์ระหว่างระบบและตัวแทน ตัวอย่างเช่น “เอทีเอ็ม รับค่า จำนวนเงินจากผู้ใช้งาน” เป็นสถานการณ์เพื่อนำไปสู่การบรรลุเป้าหมายระดับบริการ

ระดับภายใน (Internal level)

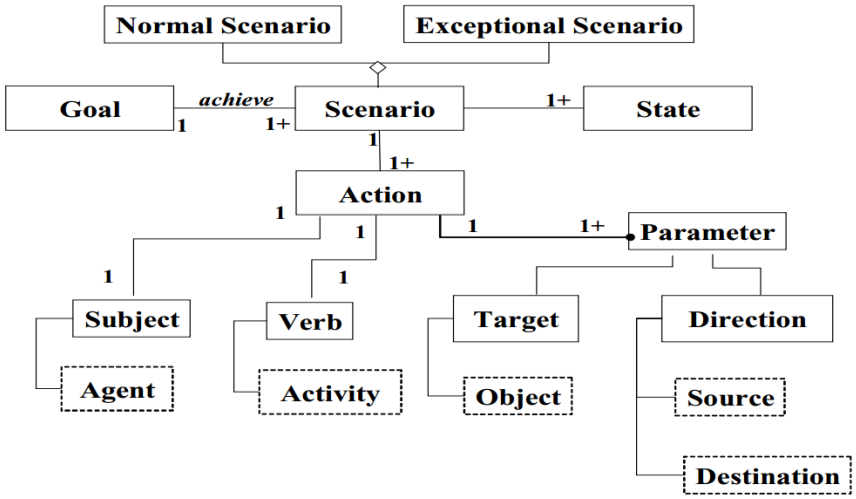
ระดับภายในจะให้ความสนใจกับอะไรที่ระบบต้องการเพื่อดำเนินการปฏิสัมพันธ์ที่เลือกจากระดับปฏิสัมพันธ์ “อะไร” หมายถึง การกระทำของระบบภายในที่เกี่ยวโยงกับอ็อบเจกต์ของระบบหรืออ็อบเจกต์ภายนอก เช่น ระบบภายนอก

เป้าหมายและสถานการณ์ของระดับนี้จะถูกเขียนในรูปแบบคู่กัน คือ <G,Sc> เมื่อ G แทน เป้าหมาย และ Sc แทน สถานการณ์ ตัวอย่างเช่น “ส่งมอบเงินสดให้กับผู้ใช้” คือเป้าหมายการ

ปฏิสัมพันธ์ สถานการณ์ระดับภายในที่เกี่ยวข้อง จะอธิบายการไหลของการปฏิสัมพันธ์ระหว่างอ็อบเจกต์เพื่อตอบสนองเป้าหมายระดับปฏิสัมพันธ์

โครงสร้างปละรูปแบบการเขียนเป้าหมายและสถานการณ์

โครงสร้างของสถานการณ์ประกอบไปด้วย ประธาน(Subject) กิริยา(Verb) และพารามิเตอร์(Parameter) พารามิเตอร์แต่ละตัวจะอยู่ในบทบาทที่แตกต่างกันขึ้นอยู่กับกิริยาซึ่งมีอยู่ 2 ประเภท ได้แก่ จุดมุ่งหมาย(Target) และทิศทาง (Direction) ตามรูปที่ 2.2



รูปที่ 2.2 โครงสร้างของสถานการณ์(Scenario) (Choi et al.,2006)

โครงสร้างของเป้าหมายนั้นจะคล้ายคลึงกับสถานการณ์แต่จะมีองค์ประกอบที่น้อยกว่า ได้แก่ กิริยา(Verb) และ พารามิเตอร์(Parameter)

สำหรับองค์ประกอบอื่นๆในโครงสร้างของสถานการณ์ที่จะถูกกล่าวถึงต่อไปได้แก่

ผู้กระทำการ (Actor) หมายถึง บุคคล ตำแหน่งหน้าที่ หรือองค์กรที่เป็นผู้ร้องขอบริการจากระบบที่สนใจอยู่ ซึ่ง ผู้กระทำการอาจจะเป็นระบบภายนอกซอฟต์แวร์ภายนอกหรืออุปกรณ์ที่มีการร้องขอใช้บริการจากระบบที่กำลังสนใจได้เช่นกัน (อัษฎาพร ทรัพย์สมบูรณ์, 2554)

ตัวแทน (Agent) หมายถึง ผู้ที่มีภาระในการกำหนดเป้าหมายหรือดำเนินการกิจกรรมในฐานะผู้ใช้งาน ซึ่งทั้งผู้กระทำการ(Actor) และระบบ สามารถเป็นตัวแทนได้

การกระทำ(Activity) หมายถึง กิริยาหลักที่จะแสดงถึงหนึ่งขั้นตอนในประโยคของสถานการณ์

อ็อบเจกต์(Object) หมายถึง กรอบความคิด(conceptual) หรือ เอนทิตี้กายภาพ (Physical entity) ที่เปลี่ยนแปลงไปโดยการกระทำ ยกตัวอย่างเช่น “ลูกค้าฝากเงินสดไปยังเครื่อง เอทีเอ็ม” สถานการณ์นี้อ็อบเจกต์คือเงินสดซึ่งมีลักษณะเป็นเอนทิตี้กายภาพ อ็อบเจกต์ เป็นได้หลายลักษณะ เช่น กรรมสิทธิ์ สถานที่ สถานะ จำนวน หรืออื่นๆ ที่สามารถเปลี่ยนแปลงไปตามการกระทำ

ทิศทาง (Direction) หมายถึง เอนทิตี้ที่ปฏิสัมพันธ์กับตัวแทนโดยจะเป็นได้ 2 ลักษณะคือ แหล่งข้อมูล(source) และปลายทาง (Destination) ซึ่งทิศทางจะระบุตำแหน่งเริ่มต้นและสิ้นสุดของอ็อบเจกต์ที่สื่อสารภายใต้สถานการณ์นั้น

ยกตัวอย่าง เช่น พิจารณา 2 สถานการณ์ดังต่อไปนี้

- ลูกค้าของธนาคารถอนเงินสดจากเอทีเอ็ม
- เอทีเอ็มรายงานธุรกรรมเงินสดไปยังธนาคาร

สถานการณ์แรก แหล่งข้อมูลของเงินสด คือ เอทีเอ็มส่วนสถานการณ์ที่ 2 ธนาคารคือปลายทาง ของธุรกรรมเงินสด

จากโครงสร้างของรูปแบบการเขียนเป้าหมายและสถานการณ์ Choi และคณะ(Choi et al., 2006) ได้เสนอว่า ในการเขียนเป้าหมายนั้นให้ใช้องค์ประกอบดังที่กล่าวมาแล้วก่อนหน้านี้เขียนเป้าหมายออกมาด้วยประโยคง่ายๆ โดยไม่มีขั้นตอนที่ซับซ้อนใดๆ แต่ในการเขียนสถานการณ์นั้นจำเป็นต้องมีขั้นตอนที่ซับซ้อนมากกว่าดังนั้นจึงได้เสนอกฎการเขียนสถานการณ์ทั้งหมด 6 ข้อและยกตัวอย่างด้วยสถานการณ์การให้บริการของเอทีเอ็ม โดยมีรายละเอียดดังนี้

กฎการเขียนสถานการณ์ 1 (Scenario authoring rule 1 :S1)

คำอธิบาย: สถานการณ์ทั้งหมดจะเขียนโดยใช้รูปแบบต่อไปนี้

ประธาน (ตัวแทน) + กิริยา + จุดมุ่งหมาย (อ็อบเจกต์) + ทิศทาง (แหล่งข้อมูล หรือ ปลายทาง)

สถานการณ์ที่คาดหวังเป็นการอธิบายถึงการกระทำอย่างใดอย่างหนึ่ง ซึ่งการกระทำนั้นจะต้องเป็นการกระทำที่ตอบสนองการบรรลุเป้าหมาย ควรอธิบายการกระทำที่คาดหวัง ไม่ใช่การกระทำที่ไม่คาดหวังหรือไม่เกี่ยวข้องกับขอบเขตของปัญหา

ตัวอย่าง: ลูกค้า (ตัวแทน) ฝาก (กิริยา) เงินสด (อ็อบเจกต์) ไปยัง เอทีเอ็ม (ปลายทาง)

กฎการเขียนสถานการณ์ 2 (Scenario authoring rule 2 :S2)

คำอธิบาย: ประธาน ควรจะถูกแทนด้วน “ตัวแทน”

ตัวอย่าง: เอทีเอ็ม (ตัวแทน) ส่งการแจ้งรหัสไปยังผู้ใช้

กฎการเขียนสถานการณ์ 3 (Scenario authoring rule 3 :S3)

คำอธิบาย: กิริยาควรจะสื่อถึงทั้งบริการในระดับบริการหรือการปฏิสัมพันธ์ในระดับปฏิสัมพันธ์ โดยที่ระดับปฏิสัมพันธ์กิริยาจะต้องเป็นกิริยาที่ต้องการกรรม

ตัวอย่าง: ผู้ใช้ ถอน(กิริยา) เงินสดจาก เอทีเอ็ม (ระดับบริการ)

เอทีเอ็ม แสดงผล (กิริยา) จำนวนเงินแก่ผู้ใช้ (ระดับปฏิสัมพันธ์)

กฎการเขียนสถานการณ์ 4 (Scenario authoring rule 4 :S4)

คำอธิบาย: จุดมุ่งหมาย ควรเป็น อ็อบเจกต์

ตัวอย่าง: เอทีเอ็ม ส่งเงินสด (อ็อบเจกต์) ให้กับผู้ใช้งาน

กฎการเขียนสถานการณ์ 5 (Scenario authoring rule 5 :S5)

คำอธิบาย: ทิศทาง ควรเป็นแหล่งข้อมูลหรือเป้าหมาย

ทิศทางทั้ง 2 ประเภท คือ แหล่งข้อมูลและปลายทาง ระบุถึงที่มาและปลายทางสำหรับของอ็อบเจกต์ที่สื่อสารในสถานการณ์นั้น

แหล่งข้อมูล คือจุดเริ่มต้นของการสื่อสาร และ ปลายทางเป็นจุดสิ้นสุดของการสื่อสาร บางครั้งการเขียนแหล่งข้อมูล มีจะมีคำเชื่อม เช่น จาก ใน ออกมาจาก และอื่นๆ ส่วนเป้าหมายก็จะมีคำเชื่อม เช่น ไปยัง สู่ เข้าไป และอื่นๆ

ตัวอย่าง: ลูกค้าของธนาคารถอนเงินสด จาก เอทีเอ็ม(แหล่งข้อมูล)

กฎการเขียนสถานการณ์ 6 (Scenario authoring rule 6 :S6)

คำอธิบาย: ระบบและตัวแทนอื่นๆ จะถูกใช้เป็นประธานและปลายทางเท่านั้น

ถ้าระบบถูกเขียนในช่องประธานตัวแทนอื่นๆ เช่น คน เครื่องจักร หรือระบบภายนอก ควรจะนำมาเติมในช่องปลายทางดังนั้น ตัวแทนอื่นๆที่ปฏิสัมพันธ์กับระบบจะถือว่าเป็นหนึ่งในผู้กระทำการ

ตัวอย่าง: ลูกค้าของธนาคาร(ตัวแทน)ถอนเงินสด จาก เอทีเอ็ม(แหล่งข้อมูล)

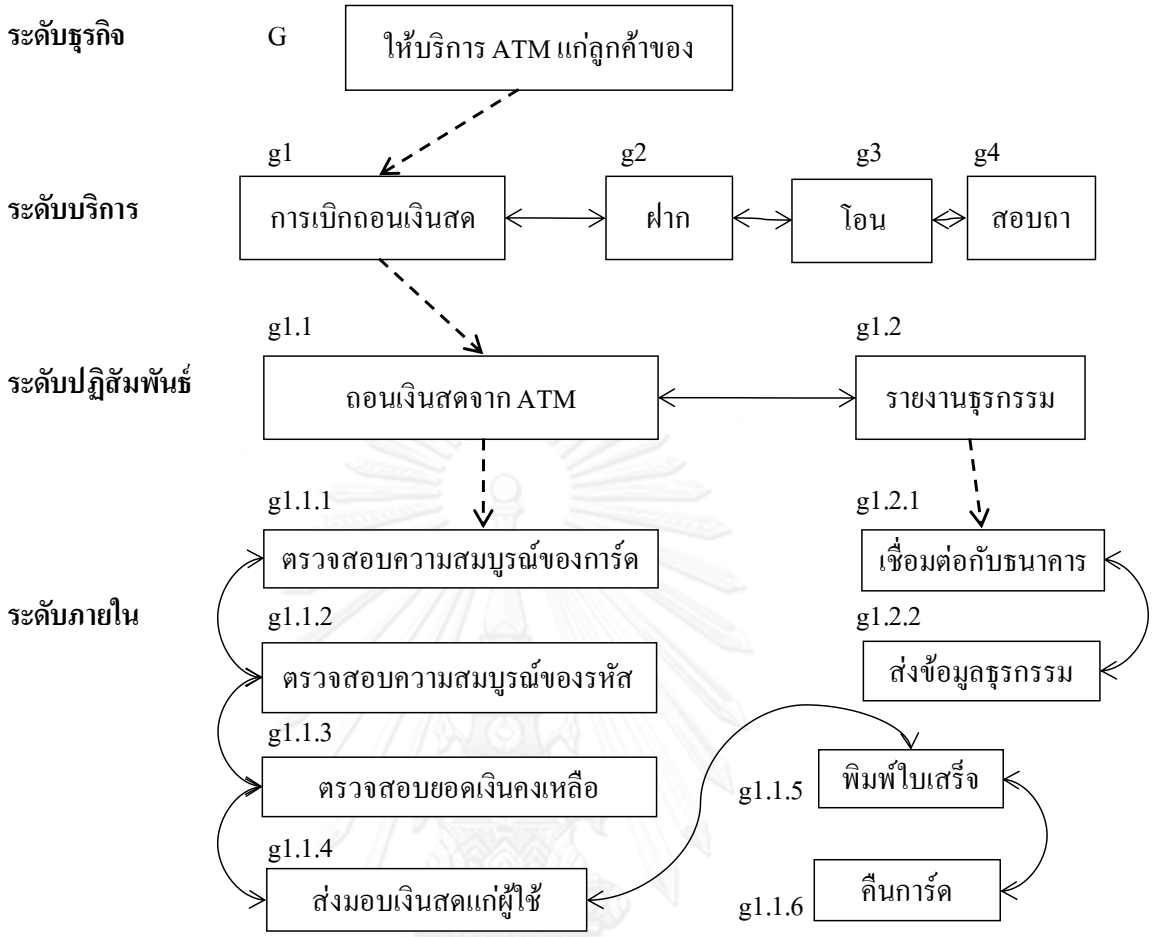
ตารางที่ 2.36 และรูปที่ 2.4 แสดงตัวอย่างการเขียนสถานการณ์และแสดงการไหลของเป้าหมาย สถานการณ์ และการการกระทำ ซึ่งแสดงให้เห็นพฤติกรรมของระบบ ของตัวอย่างบริการเอทีเอ็มโดยที่ในรูปที่ 2.3 สัญลักษณ์กรอบสี่เหลี่ยมแสดงถึงเป้าหมาย ลูกศรเส้นประทิศทางเดียวแสดงถึงการเชื่อมโยงระหว่างเป้าหมายกับเป้าหมายในระดับล่าง ลูกศรเส้นทึบสองทิศทางแสดงถึงการเชื่อมโยงของเป้าหมายที่มีเป้าหมายระดับบนอันเดียวกัน

จะเห็นได้ว่าสถานะ(State) นั้นแสดงเงื่อนไขที่จำเป็นของการกระทำและสถานะจะกลายเป็นเป้าหมายในระดับถัดลงไป เช่น “สถานะถ้าการ์ดสมบูรณ์” ที่ระดับปฏิสัมพันธ์ จะกลายเป็นเป้าหมายชื่อ “ตรวจสอบความสมบูรณ์ของการ์ด” ที่ระดับภายใน

ตารางที่ 2.36 ตัวอย่างสถานการณ์ของบริการ เอทีเอ็ม ที่เขียนโดยใช้กฎข้อ S1-S6 (Choi et al.,2006)

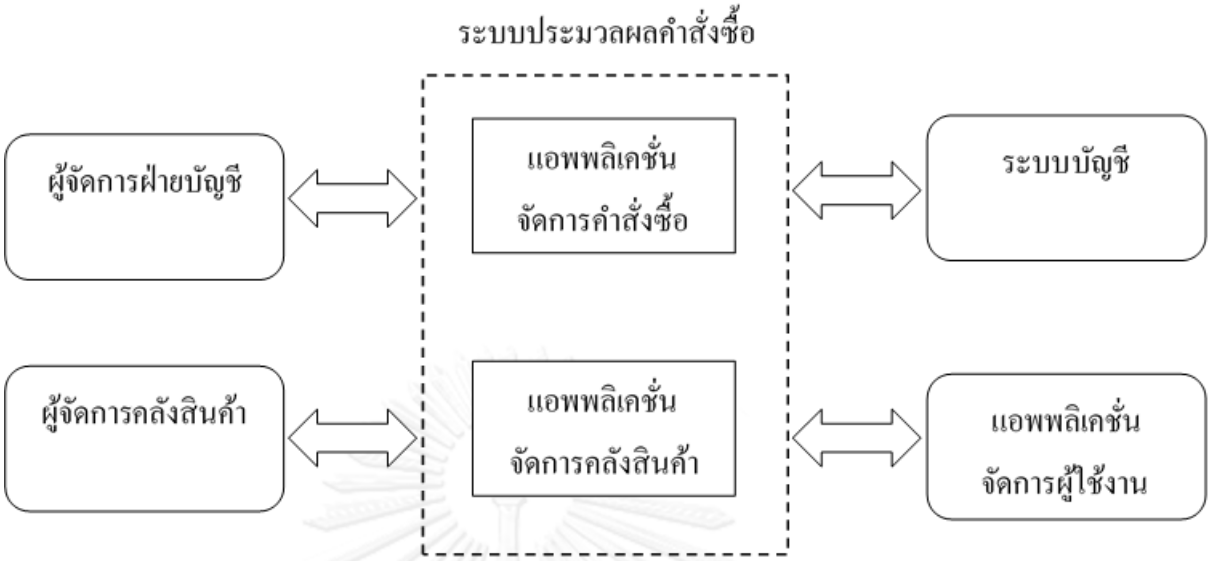
เป้าหมาย	สถานการณ์
การเบิกถอนเงินสด (g1)	1.ลูกค้าถอนเงินสดจาก ATM 2.ATM รายงานธุรกรรมเงินสดไปยังธนาคาร
ถอนเงินสดจาก ATM (g1.1)	1.ATM ได้รับการ์ดจากผู้ใช้ *ถ้าการ์ดสมบูรณ์(สถานะ)จึงทำงานต่อ* 2.ATM ส่งพรอมต์สำหรับกรอกรหัสแก่ผู้ใช้ 3.ATM ได้รับรหัสจากผู้ใช้ *ถ้ารหัสสมบูรณ์(สถานะ)จึงทำงานต่อ* 4.ATM แสดงพรอมต์สำหรับกรอกจำนวนเงิน 5.ATM ได้รับจำนวนเงินจากผู้ใช้ *ถ้าจำนวนเงินเพียงพอ(สถานะ)จึงทำงานต่อ* 6.ATM คืนการ์ดให้กับผู้ใช้ *ถ้าผู้ใช้ขอให้ ATM ออกใบเสร็จ(สถานะ)จึงทำงานต่อ* 7.ATM คายใบเสร็จที่พิมพ์ให้กับผู้ใช้ 8.ATM ส่งมอบเงินสดให้แก่ผู้ใช้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



รูปที่ 2.3 แสดงผังการไหลของเป้าหมายในระดับต่างๆของบริการเอทีเอ็ม(Choi et al.,2006)

หลังจากเขียนความต้องการในรูปแบบเป้าหมายและสถานการณ์แล้ว ขั้นตอนต่อไปก็จะนำความต้องการไปถอดเอาฟังก์ชั่นพอยต์ออกมาโดยอธิบายด้วยตัวอย่างระบบการประมวลผลคำสั่งซื้อ ซึ่งมีขอบเขตของระบบตามรูป 2.4



รูปที่ 2.4 แสดงขอบเขตของระบบประมวลผลคำสั่งซื้อ(Choi et al.,2006)

ขั้นที่ 1 การถอด ฟังก์ชั่นเชิงข้อมูล (ILFs และ ELFs) จากเอกสารความต้องการซอฟต์แวร์

พิจารณาจากจุดมุ่งหมายแสดงให้เห็นถึงข้อมูลหรือวิธีการส่งข้อมูลระหว่างผู้ใช้งาน และแอพพลิเคชั่นเป้าหมาย(Target Application) หรือแอพพลิเคชั่นภายนอก(External Application) ว่าข้อมูลถูกเก็บรักษาไว้ที่ใด ซึ่งถ้าข้อมูลนี้ถูกเก็บรักษาโดยแอพพลิเคชั่นเป้าหมาย จะทำให้สามารถระบุได้ว่าเป็น ILFs หากถูกเก็บโดย แอพพลิเคชั่นภายนอกจะถูกกำหนดให้เป็น ELFs

การนับ DET และ RET

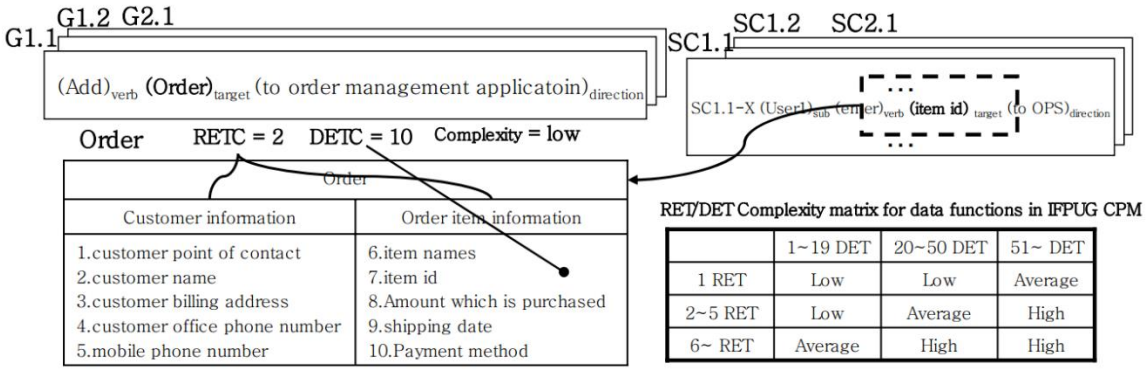
แต่ละข้อมูล(Data) ที่ไม่ซ้ำในช่องจุดมุ่งหมายให้นับเป็น 1 DET และถ้าข้อมูลเหล่านั้นสามารถ จัดกลุ่มเป็นกลุ่มย่อย ให้นับ 1 RET ต่อจำนวนกลุ่มย่อยจากนั้นสามารถนำไปเทียบค่าความซับซ้อนจากตาราง DET RET ได้

ตัวอย่าง : กำหนดให้ 1 คือเป้าหมาย 2 และ 3 คือ สถานการณ์ภายใต้เป้าหมาย

1. เพิ่ม(กิริยา)คำสั่งซื้อ(จุดมุ่งหมาย)ไปยังแอพพลิเคชั่นจัดการคำสั่งซื้อ(ทิศทาง)
2. แอพพลิเคชั่นจัดการคำสั่งซื้อ(ประธาน)รับ(กิริยา)invoice id(จุดมุ่งหมาย)จากระบบบัญชี(ทิศทาง)
3. แอพพลิเคชั่นจัดการคำสั่งซื้อ(ประธาน)รับ(กิริยา) invoice contents (จุดมุ่งหมาย)จากระบบบัญชี (ทิศทาง)

กลุ่มข้อมูล “Order” ถูกเก็บโดยแอพพลิเคชั่นจัดการคำสั่งซื้อ ส่วน “invoice id” และ “invoice contents” ไม่ได้อยู่ในกลุ่มข้อมูล “order” แต่อยู่ในกลุ่มข้อมูล “invoice information” ซึ่งถูกเก็บโดยระบบบัญชีดังนั้นในกรณีที่ระบบที่เราสนใจคือแอพพลิเคชั่นจัดการคำสั่งซื้อ “order” จะถูกระบุให้เป็น ILF และ “invoice information” จะเป็น EIF.

ตัวอย่าง : “Order” เป็นฟังก์ชั่นข้อมูลและเป็นข้อมูลบนเป้าหมาย G1.1 G1.2 และ G2.1 ตามรูป 2.6 มันประกอบด้วยข้อมูล 10 ตัวที่แตกต่างกัน ได้แก่ customer point of contract customer name customer billing address และอื่นๆ ซึ่งสามารถแบ่งออกได้เป็น 2 กลุ่ม คือ Customer Information และ Order item Information ซึ่งในกรณีนี้ “Order” จะประกอบด้วย 10 DET และ 2 RET



รูปที่ 2.5 แสดงสถานการณ์ตัวอย่างสำหรับพิจารณาฟังก์ชั่นเชิงข้อมูล(Choi et al.,2006)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ขั้นที่ 2 การถอด ฟังก์ชั่นเชิงรายการ (EI EO และ EQ) จากเอกสารความต้องการซอฟต์แวร์

พิจารณาจากกิริยาของสถานการณ์หากมีการกระทำในลักษณะของการกรอกข้อมูลเข้าสู่ระบบสถานการณ์นั้นจะถูกกำหนดให้เป็น EI หากมีการประมวลผลเพื่อแสดงข้อมูลผ่านการคำนวณทางคณิตศาสตร์หรือเปลี่ยนแปลงพฤติกรรมของระบบสถานการณ์นั้นจะถูกจัดเป็น EO ถ้าหากเป็นการเรียกดูข้อมูลจากฐานข้อมูลสถานการณ์นั้นจะถูกจัดเป็น EQ ตัวอย่างของกิริยาในการพิจารณาฟังก์ชั่นเชิงรายการแสดงตามตาราง 2.37

ตาราง 2.37 แสดงตัวอย่างของกิริยาในการพิจารณาฟังก์ชั่นเชิงรายการ

Data send type verbs	Data receiving type verbs
Send,display,add,dispatch,forward,direct, convey,remit,input,request,modify,fill, output,enter,list	Receive,get,accept,be given, pick up,collect,obtain,acquire,take, validate,check

การนับ DET และ FTR

DET ของแต่ละรายการ คือผลรวมของ

(จำนวนของข้อมูลที่ไม่ซ้ำในจุดมุ่งหมายของสถานการณ์ที่ทำให้รายการสำเร็จ + 1 (ถ้ามี error confirm หรือ complete message) + 1 (ถ้ามีขั้นตอนเริมต้นของการทำรายการ)+ 1 (ต่อ 1 ฟิลด์ข้อมูลที่ต้องกรอกสำหรับประมวลผล EO หรือ EQ)

ทั้งนี้ข้อมูลที่ไม่ซ้ำในจุดมุ่งหมายของสถานการณ์ควรเป็นการส่งข้อมูลเข้าหรือออกระหว่างแอพพลิเคชั่นเป้าหมาย และ แอพพลิเคชั่นภายนอก ซึ่งจะต้องอยู่ใน ILFs ELFs

FTR ของรายการ คือ จำนวนของ ILFs และ ELFs ที่ถูกอ้างอิงโดยรายการ

หลังจากจำแนกฟังก์ชั่นการทำงานทั้งหมดออกมาแล้ว ขั้นตอนต่อไปก็จะนำรายการฟังก์ชั่นที่ถอดจากเอกสารความต้องการไปใช้ประมาณการขนาดตามวิธีฟังก์ชั่นพอยต์เพื่อทำการประมาณขนาดและประมาณค่าความพยายามต่อไป

ตัวอย่างการวิเคราะห์ฟังก์ชั่นเชิงรายการ

- กำหนดสถานการณ์ดังนี้ ประธาน คือ ผู้ใช้งาน และ ปลายทางคือ คือ แอพพลิเคชั่นจัดการคำสั่งซื้อ ซึ่งเป็นแอพพลิเคชั่นเป้าหมาย ซึ่งเขียนในรูปแบบสถานการณ์ดังนี้

  “ผู้จัดการฝ่ายบัญชี(ประธาน)เพิ่ม(กิริยา)คำสั่งซื้อ(จุดมุ่งหมาย)ไปยังแอพพลิเคชั่นจัดการคำสั่งซื้อ(ปลายทาง)”

ดังนั้น เนื่องจากกิริยาในสถานการณ์นี้เป็นประเภท Data send type verb และเป็นการส่งข้อมูลไปยังแอพพลิเคชั่นเป้าหมาย ดังนั้น รายการนี้จะถูกระบุเป็นฟังก์ชั่นรายการประเภท EI สำหรับแอพพลิเคชั่นนี้

- การนับ DET สำหรับฟังก์ชั่น “เพิ่มคำสั่งซื้อ” โดยที่ฟังก์ชั่นนี้เกี่ยวข้องกับข้อมูลจำนวน 14 ตัว โดยแบ่งเป็น 10 ตัวจากชุดข้อมูล order อีก 2 ข้อมูลจากชุดข้อมูล invoice และ 2 ข้อมูลสำหรับชุดข้อมูล inventory ซึ่งในการทำงานจะมี Complete Message เพื่อยืนยันการทำรายการและถือว่า รายการนี้เริ่มต้นด้วยการกดปุ่ม OK ดังนั้นการนับจำนวน DET ของฟังก์ชั่นนี้จะได้เท่ากับ 16 (10+2+2+1+1)

- การนับ FTR ของฟังก์ชั่น “เพิ่มคำสั่งซื้อ” ซึ่งมีองค์ประกอบของข้อมูลที่ถูกนับเป็น DET ประกอบไปด้วยกลุ่มข้อมูลที่ถูกเก็บโดยแอพพลิเคชั่นเป้าหมาย 1 ไฟล์ชื่อว่า “order” และกลุ่มข้อมูลที่ถูกเก็บโดยแอพพลิเคชั่นภายนอก 2 ไฟล์ได้แก่ “invoice” และ “inventory” จึงนับเป็น 1 ILF และ 2 EIF ที่ถูกอ้างอิงโดยรายการ “เพิ่มคำสั่งซื้อ” ดังนั้น FTR ของฟังก์ชั่นนี้คือ 3

- กำหนดสถานการณ์ดังนี้ ประธาน คือ ผู้ใช้งาน และปลายทางคือ แอพพลิเคชั่นจัดการคลังสินค้าซึ่งเป็นแอพพลิเคชั่นเป้าหมาย ซึ่งเขียนในรูปแบบสถานการณ์ดังนี้

  “ผู้จัดการคลังสินค้า(ประธาน)ตรวจสอบ(กิริยา)สินค้าคงคลัง(จุดมุ่งหมาย)จาก
  แอพพลิเคชั่นจัดการคลังสินค้า(ปลายทาง)”

  ดังนั้นเนื่องจาก สถานการณ์นี้มีกิริยาเป็นประเภท Data receiving type verb ซึ่งมีวัตถุประสงค์หลัก คือการได้รับข้อมูลสินค้าคงคลังและการตรวจสอบ ดังนั้น รายการนี้จะถูกระบุเป็นฟังก์ชั่นรายการประเภท EO หรือ EQ

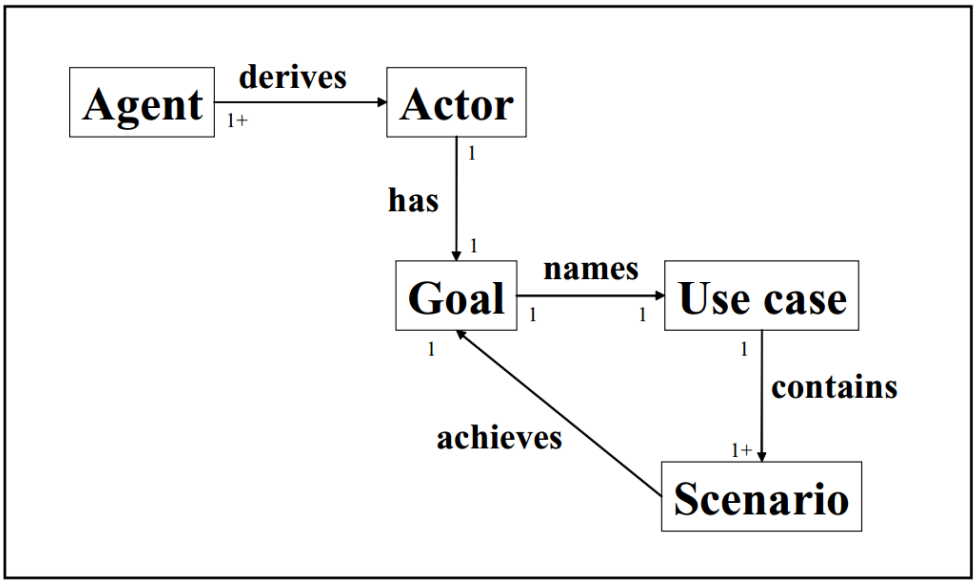
- หลังจากที่มีการระบุเป็น EO หรือ EQ แล้วนั้นจะต้องทำการวิเคราะห์ต่อว่า รายนั้นมีการคำนวณทางคณิตศาสตร์หรือไม่ ซึ่งในสถานการณ์

  “ผู้จัดการคลังสินค้า(ประธาน)ตรวจสอบ(กิริยา)สินค้าคงคลัง(จุดมุ่งหมาย)จาก
  แอพพลิเคชั่นจัดการคลังสินค้า(ปลายทาง)”

เป็นการสอบถามข้อมูลสินค้าคงคลังจากจากระบบไม่ได้มีการคำนวณหรือเปลี่ยนแปลงพฤติกรรมของระบบ ดังนั้นสถานการณ์นี้จะถูกระบุเป็นฟังก์ชั่นรายการประเภท EQ

ทั้งนี้ในงานของ Kim และคณะ ในปี 2004 ยังได้พูดถึงการนำความต้องการที่เขียนในรูปแบบเป้าหมายและสถานการณ์ ไปทำการสร้างแผนภาพยูสเคสด้วย โดยได้เสนอกฎในการจับคู่ ซึ่งมีแนวคิดหลักนั่นคือ การหยิบเอาเป้าหมายและสถานการณ์ในระดับปฏิสัมพันธ์มาช่วยในการสร้างยูสเคส ซึ่งวิธีการที่เสนอนี้จะให้ความสำคัญกับการปฏิสัมพันธ์ระหว่างระบบและตัวแทน

โดยวัตถุประสงค์ของการกำหนดยูสเคส คือเพื่ออธิบายว่าตัวแทนมีการปฏิสัมพันธ์กับระบบอย่างไรเพื่อเข้าถึงบริการและบรรลุเป้าหมายของเขา ด้วยเหตุนี้จึงเสนอ แผนภาพความสัมพันธ์ตามรูป 2.7 ที่เป็นการจับภาพของความสัมพันธ์ระหว่าง ตัวแทน เป้าหมาย สถานการณ์ และ ยูสเคส (Kim, Park,and Sugumaran, 2004)



รูปที่ 2.6 แสดงความสัมพันธ์ระหว่างเป้าหมายและองค์ประกอบอื่น(Kim et al., 2004)

หลังจากการวิเคราะห์แต่ละยูสเคส ก็จะเป็นการแต่งเติมด้วย องค์ประกอบของกิจกรรมภายในของซอฟต์แวร์ ซึ่งหมายถึงการนำเป้าหมายและสถานกานการณ์ที่ระดับภายในที่มีความสัมพันธ์กับเป้าหมายในระดับปฏิสัมพันธ์มาพิจารณาด้วยเพื่อให้ยูสเคสมีความสมบูรณ์ ซึ่งได้เสนอแนวทางในการแปลงความต้องการเป็นยูสเคสไว้ 4 ข้อ ซึ่งมีรายละเอียดดังนี้

แนวทางในการแปลงข้อที่ 1 (Conversion guiding rule 1 :C1)

คำอธิบาย: เป้าหมายที่ระดับปฏิสัมพันธ์จะถูกกำหนดให้เป็น ยูสเคส ตามรูป ที่ 2.7

ด้วยวิธีการด้านบนเป้าหมายที่ระดับปฏิสัมพันธ์จะถูกจับคู่กับยูสเคส และยูสเคสจะถูกตั้งชื่อให้สอดคล้องกับเป้าหมาย

ตัวอย่าง: สำหรับเป้าหมาย g1 เป้าหมายระดับปฏิสัมพันธ์สำหรับตัวอย่างบริการเอทีเอ็ม

ถอนเงินสดจากเอทีเอ็ม

รายงานธุรกรรม

เป้าหมายเหล่านี้จะถูกกำหนดเป็นยูสเคส

แนวทางในการแปลงข้อที่ 2 (Conversion guiding rule 2 :C2)

คำอธิบาย: ตัวแทนที่อยู่ในสถานการณ์ภายใต้เป้าหมายและเป็นผู้ดำเนินการเพื่อบรรลุเป้าหมายจะกลายเป็น ผู้กระทำการหลัก

เป้าหมายจะบรรลุได้โดยสถานการณ์และตัวแทน ที่พบในสถานการณ์ที่อภิปรายในกฎการเขียนสถานการณ์ ตัวแทนจะถูกนำไปใช้เป็นทั้ง ประธาน หรือปลายทาง เพราะฉะนั้นตัวแทนที่อยู่ในช่องของประธานหรือปลายทาง จะถูกระบุเป็นผู้กระทำการทั้งหมด ยกเว้น ระบบที่กำลังสนใจ

ตัวอย่าง: จากรูป ตัวอย่างสถานการณ์บริการเอทีเอ็ม 2.29 แสดง ผู้กระทำการที่พบในสถานการณ์ภายใต้ เป้าหมาย g1.1 (withdraw cash from เอทีเอ็ม) ของตัวอย่างการบริการเอทีเอ็ม สถานการณ์ Sc 1.1 มี 8 การกระทำ ตัวแทนที่เกี่ยวข้องกับปลายทางในการกระทำทั้งหมดถูกระบุว่า คือ ผู้ใช้งาน ดังนั้น ในกรณีของ Sc1.1, ผู้ใช้งาน คือ ผู้กระทำการ

แนวทางในการแปลงข้อที่ 3 (Conversion guiding rule 3 :C3)

คำอธิบาย: สถานะ(Sate) ที่มีอยู่ในสถานการณ์ที่ระดับปฏิสัมพันธ์จะจับคู่กับเป้าหมายระดับภายใน

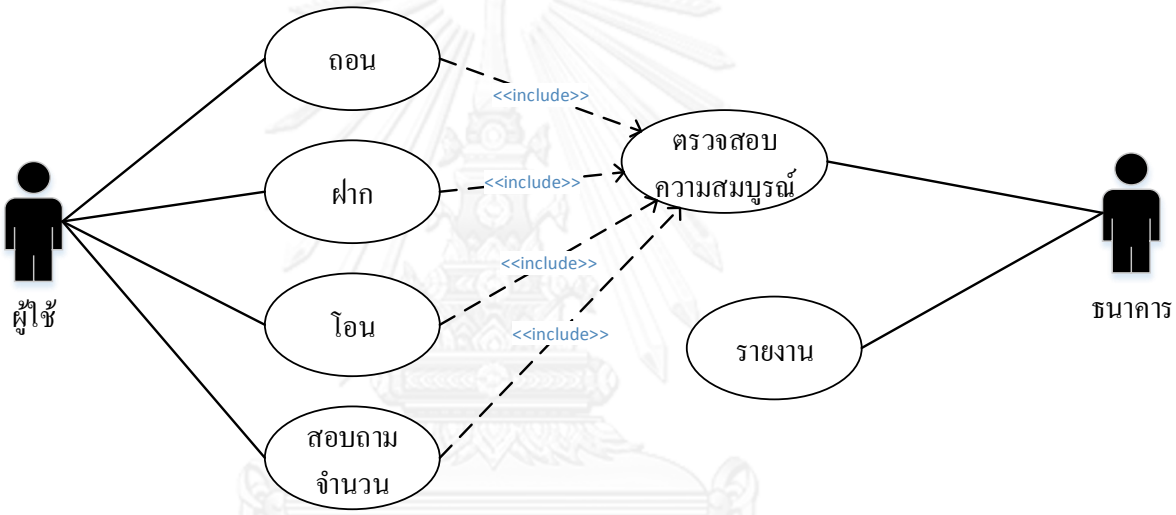
สถานะ(State) นำเสนอเงื่อนไขที่จำเป็นของการกระทำนั้น ซึ่งจะมีความเชื่อมโยงกับเป้าหมายระดับภายใน ดังนั้นเมื่อทำการกำหนดยูสเคส เป้าหมายและสถานการณ์ที่ระดับภายในจะช่วยให้เข้าใจรายละเอียดของเกี่ยวกับสถานะได้มากขึ้น

ตัวอย่าง: สถานะ “ถ้าการ์ดมีความสมบูรณ์” ของ Sc1.1 จะถูกอธิบายในเป้าหมายย่อย “ตรวจสอบความสมบูรณ์ของการ์ด” ที่ระดับภายใน

แนวทางในการแปลงข้อที่ 4 (Conversion guiding rule 4 :C4)

คำอธิบาย: ถ้าเป้าหมายที่ระดับภายในมีการเชื่อมโยงกับเป้าหมายระดับปฏิสัมพันธ์มากกว่า 1 เป้าหมาย จะต้องกำหนดให้เป้าหมายระดับภายในนั้นเป็นอีกหนึ่งยูสเคสด้วยความสัมพันธ์แบบ <include>

ตัวอย่าง:ในรูปที่ 2.7 เป้าหมาย “ตรวจสอบความสมบูรณ์ของการ์ด” เชื่อมโยงกับ 2 เป้าหมายในระดับปฏิสัมพันธ์ คือ “ถอนเงินสดจาก เอทีเอ็ม” และ “ฝากเงินสดเข้าไปยัง เอทีเอ็ม” เพราะฉะนั้นเป้าหมาย “ตรวจสอบความสมบูรณ์ของการ์ด”จะถูกกำหนดเป็นยูสเคสด้วยความสัมพันธ์ <include>



รูปที่ 2.7 แสดงแผนภาพยูสเคสสำหรับการให้บริการเอทีเอ็ม
โดยใช้การแปลงตามกฎทั้ง 4 ข้อ(Kim et al., 2004)

แผนภาพยูสเคสที่ได้จากการแปลงตามกฎทั้ง 4 ข้อ ซึ่งประกอบด้วย 2 ผู้กระทำการ คือ ผู้ใช้งาน และ ธนาคาร และ 6 ยูสเคส ยูสเคสที่เกี่ยวข้องกับผู้ใช้งานได้แก่ ถอน ฝาก โอน สอบถามจำนวน และยูสเคสเหล่านี้ยังประกอบด้วยยูสเคส ตรวจสอบความสมบูรณ์ ซึ่งเกี่ยวข้องกับธนาคาร และธนาคารเกี่ยวข้องกับอีกหนึ่งยูสเคสคือ รายงาน

วิธีการของ Choi และคณะ(Choi et al., 2006) นี้สามารถจัดการกับปัญหาในกระบวนการสอบถามความต้องการจากความต้องการในแบบข้อความและทำให้ได้ข้อมูลที่จำเป็นเพียงพอสำหรับการนำมาสร้างแผนภาพยูสเคสตลอดจนการวิเคราะห์ฟังก์ชั่นพอยต์เพื่อประมาณการขนาดซอฟต์แวร์ต่อไป

2.6 ลักษณะและความท้าทายในการพัฒนาโมบายแอพพลิเคชั่น

การพัฒนาโมบายแอพพลิเคชั่นนั้นถือว่าเติบโตและได้รับความนิยมอย่างรวดเร็ว นักพัฒนาจำนวนมากหันมาพัฒนาโมบายแอพพลิเคชั่น โดยให้ความสำคัญกับการใช้งานง่ายขั้นตอนไม่ซับซ้อนและฟังก์ชั่นการทำงานที่มีลักษณะเป็นงานเล็กๆ(Tracy,2012) ซึ่งก็จะมีความท้าทายที่แตกต่างไปจากการพัฒนาซอฟต์แวร์บนคอมพิวเตอร์ในปี ค.ศ.2013 Joorabchi และคณะจึงได้ทำการสำรวจความคิดเห็นจากนักพัฒนาโมบายแอพพลิเคชั่นผู้มีประสบการณ์ พบว่าความท้าทายของการพัฒนาโมบายแอพพลิเคชั่นนั้นมีในหลายประเด็น (Joorabchi, Mesbah, Kruchten,2013) ได้แก่

- การพัฒนาโมบายแอพพลิเคชั่นจำเป็นต้องพัฒนาโดยใช้โปรแกรมสภาพแวดล้อมจำลอง(Simulator) เพื่อสร้างสภาพแวดล้อมจำลองของอุปกรณ์เคลื่อนที่ทำให้มีผลต่อเวลาที่ใช้พัฒนา เพราะประสิทธิภาพของโปรแกรมสภาพแวดล้อมจำลองแตกต่างกันกับอุปกรณ์จริงและไม่สามารถใช้ฟังก์ชั่นบางอย่างได้ หรือการเรียกใช้งานเซนเซอร์บางอย่าง เช่น จีพีเอส(GPS) กล้องถ่ายรูป บลูทูธ(Bluetooth)
- รูปแบบของแพลตฟอร์ม แอนดรอยด์(Android)เป็นซอฟต์แวร์โอเพนซอร์ส(Open Source) ในขณะที่ไอโอเอส(iOs) และ วินโดวส์โฟน(Windows Phone) เป็นซอฟต์แวร์ปิด(Closed Source) ซึ่งทำให้เกิดความลำบากแก่นักพัฒนาในการจัดการกับคุณลักษณะบางอย่าง เช่น การหาวิธีเชื่อมต่อกับบลูทูธ(Bluetooth)ของไอโอเอส แต่ในอีกแง่มุมพบว่าซอฟต์แวร์โอเพนซอร์สก็สร้างปัญหาเช่นกันคือ การขาดมาตรฐานการใช้งาน
- ข้อมูลปริมาณมากไม่สามารถจัดเก็บภายในแอพพลิเคชั่นได้ ทำให้จำเป็นจะต้องมีการเชื่อมต่อข้อมูลกับแหล่งข้อมูลอื่นนั่นทำให้โมบายแอพพลิเคชั่นจำเป็นจะต้องพึ่งพาเครือข่ายข้อมูลที่มีประสิทธิภาพ เพราะบางแอพพลิเคชั่นจะทำงานไม่ได้หากไม่มีการเชื่อมต่อกับเครือข่าย
- ระบบร้านค้าออนไลน์มีมาตรการควบคุมแอพพลิเคชั่นที่จะขึ้นวางขายทำให้จะต้องคำนึงถึงมาตรการของร้านค้าออนไลน์ด้วยนอกจากความต้องการของซอฟต์แวร์แล้ว ซึ่งข้อจำกัดนี้ส่งผลให้เอพีไอ(API)บางตัวไม่สามารถใช้งานได้ทำให้อาจจะต้องเสียเวลามากขึ้นในการค้นหาเอพีไออื่นมาทดแทน
- ส่วนติดต่อผู้ใช้และประสบการณ์ของผู้ใช้งานที่ต่างกันไปในแต่ละแพลตฟอร์ม รวมทั้งการออกแบบตัวอุปกรณ์เคลื่อนที่ที่แตกต่างกันทำให้ นักพัฒนาต้องออกแบบการใช้งานของโมบายแอพพลิเคชั่นให้เหมาะสมกับพฤติกรรมการใช้งานหลายรูปแบบ หรือแต่แม้แต่ในแพลตฟอร์มเดียวกันก็ยังมีความแตกต่างกัน ทั้งในเรื่องของหน่วย

ประมวลผล หน่วยความจำ ความละเอียดของหน้าจอ การประมวลผลกราฟิก ในขณะที่โมบายแอพพลิเคชั่นพยายามพัฒนาให้มีการแสดงผลทางด้านกราฟิกที่สวยงามมากขึ้น แต่ก็ต้องคำนึงถึงอุปกรณ์รุ่นก่อนหน้าที่อาจจะมีประสิทธิภาพที่ด้อยกว่า

ในกระบวนการพัฒนาโมบายแอพพลิเคชั่นนอกจากจะให้ความสำคัญกับฟังก์ชั่นการใช้งานแล้วยังให้ความสำคัญกับประสบการณ์ของผู้ใช้งานด้วยดังนั้นแบบจำลองหน้าจอจึงมีความสำคัญเพราะแอพพลิเคชั่นที่ดีจะต้องออกแบบโดยคำนึงถึงพฤติกรรมการใช้งานของผู้ใช้ด้วยซึ่งนักพัฒนาจะให้ความสำคัญกับจุดนี้ การสร้างแบบจำลองหน้าจอทำให้นักพัฒนากำหนดรูปแบบการใช้งานให้สอดคล้องกับพฤติกรรมของผู้ใช้งานได้ (Arhippainen and Tähti, 2003)

ในปี ค.ศ. 2010 Wasserman (Wasserman, 2010) ก็ได้กล่าวถึงปัจจัยอื่นที่ทำให้การพัฒนาโมบายแอพพลิเคชั่นแตกต่างจากการพัฒนาซอฟต์แวร์บนคอมพิวเตอร์ได้แก่ การใช้งานร่วมกับความสามารถของตัวอุปกรณ์เคลื่อนที่ เซนเซอร์ต่างๆ เนื้อที่การจัดเก็บข้อมูลที่จำกัด การจัดการเซนเซอร์ต่างๆ แพลตฟอร์มที่แตกต่างกันและมีจำนวนรุ่นที่มาก พลังงานของแบตเตอรี่

การพัฒนาโมบายแอพพลิเคชั่นจำเป็นจะต้องตระหนักถึงความสำคัญของประสบการณ์ใช้งานของผู้ใช้ ดังนั้นจึงจำเป็นอย่างยิ่งที่จะต้องจัดทำแบบจำลองหน้าจอของส่วนติดต่อผู้ใช้ โดยเฉพาะอย่างยิ่งเมื่อมีการพัฒนาเพื่อใช้งานบนอุปกรณ์ที่มีความแตกต่างกัน

จากการสัมภาษณ์นักพัฒนาโมบายแอพพลิเคชั่นในประเทศไทยบางรายพบว่าโดยปกติโมบายแอพพลิเคชั่นจะมีความต้องการเพื่อทำงานใดงานหนึ่งโดยเฉพาะและโดยทั่วไปนักพัฒนาหนึ่งคนจะรับผิดชอบการพัฒนาหนึ่งโครงการหรือหนึ่งแอพพลิเคชั่น ในการประมาณค่าพยายามนั้น ผู้จัดการโครงการ และนักพัฒนาจะทำการกำหนดค่าความพยายาม โดยอิงจากฟังก์ชั่นการทำงานของโมบายแอพพลิเคชั่น และใช้ประสบการณ์ของนักพัฒนาในการกำหนดค่าความพยายามในการทำงานแต่ละงาน (Task) ทำให้ยังเกิดปัญหาการประมาณการที่คลาดเคลื่อน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 3
ระเบียบวิธีวิจัย

บทนี้จะแสดงแนวทางในการตอบวัตถุประสงค์ของการวิจัย โดยเนื้อหาจะประกอบไปด้วยส่วนต่างๆได้แก่ การดำเนินงานวิจัย ตัวแปรที่ศึกษา สมมติฐานการวิจัย หน่วยทดลอง เครื่องมือที่ใช้ในการวิจัย ความถูกต้องและความน่าเชื่อถือของข้อมูล และกรอบการวิเคราะห์ข้อมูล

3.1 การดำเนินงานวิจัย

สำหรับการวิจัยในครั้งนี้มีจุดประสงค์เพื่อ เปรียบเทียบวิธีการประมาณการค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น จากวิธีการประมาณการขนาดซอฟต์แวร์ 4 วิธีการ คือ (1) วิธีฟังก์ชั่นพอยต์ (2) วิธีอ็อบเจกต์พอยต์ (3) วิธียูสเคสพอยต์ และ (4) วิธีของ Nikunj Sakhrelia ด้วยการทดสอบด้วยข้อมูลโครงการพัฒนาโมบายแอพพลิเคชั่นจริงจากบริษัทรับพัฒนาโมบายแอพพลิเคชั่นและนักพัฒนาโมบายแอพพลิเคชั่นอิสระ

หลังจากผู้วิจัยได้ทำการทบทวนวรรณกรรมและสรุปข้อมูลที่ต้องการใช้ในการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีการทั้ง 4 วิธี ได้แก่

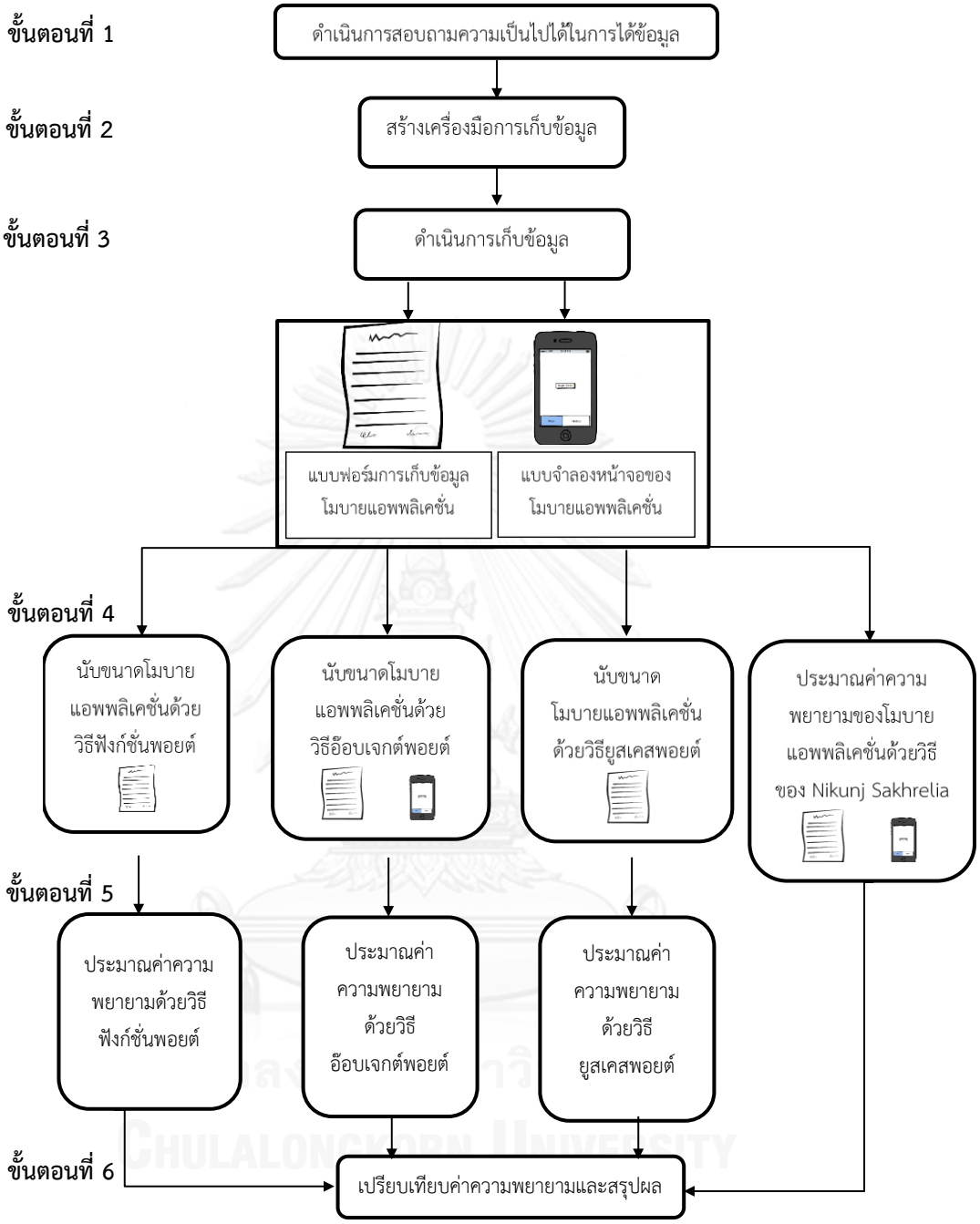
1. ข้อมูลของฟังก์ชั่นการทำงานทั้งหมดของโมบายแอพพลิเคชั่นซึ่งอธิบายขั้นตอนในการทำงาน
2. รายละเอียดของไฟล์ข้อมูลทั้งหมดที่ใช้ในโมบายบายแอพพลิเคชั่น
3. แบบจำลองหน้าจอของโมบายแอพพลิเคชั่น
4. แผนภาพยูสเคสและคำอธิบายสถานการณ์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ผู้วิจัยได้ดำเนินการวิจัยโดยมีขั้นตอนตามรูปที่ 3.1 ได้แก่

1. การดำเนินการสอบถามความเป็นไปได้ในการได้ข้อมูล ขั้นตอนนี้เป็นการติดต่อไปยังนักพัฒนาโมบายแอพพลิเคชั่นที่ผู้วิจัยคาดว่าจะให้ข้อมูลได้ โดยคำแนะนำของอาจารย์และเพื่อนของผู้วิจัย
2. สร้างเครื่องมือการเก็บข้อมูล เพื่อให้ได้ข้อมูลในรูปแบบที่ผู้วิจัยสามารถนำไปประมาณขนาดโมบายแอพพลิเคชั่นตามวิธีที่ผู้วิจัยเลือกมาใช้ในงานวิจัยนี้ได้
3. การดำเนินการเก็บข้อมูล ทำการเก็บข้อมูลจากนักพัฒนาโมบายแอพพลิเคชั่นโดยใช้เครื่องมือในการเก็บข้อมูลที่ผู้วิจัยสร้างขึ้น
4. การนำข้อมูลจากแบบฟอร์มการเก็บข้อมูลมาประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีการทั้ง 4 วิธี
5. ประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น
6. เปรียบเทียบค่าความพยายามที่ได้จากการประมาณการด้วยวิธีการทั้ง 4 วิธี

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



รูปที่ 3.1 แสดงขั้นตอนการดำเนินงานวิจัย

ขั้นตอนที่ 1 การดำเนินการสอบถามความเป็นไปได้ในการได้ข้อมูล

ผู้วิจัยได้ติดต่อไปยังนักพัฒนาโมบายแอพพลิเคชั่นที่ผู้วิจัยคาดว่าสามารถให้ข้อมูลแก่ผู้วิจัยได้ โดยได้รับคำแนะนำจากอาจารย์และเพื่อนของผู้วิจัย ในขั้นตอนนี้มีนักพัฒนาโมบายแอพพลิเคชั่นที่ผู้วิจัยได้ทำการติดต่อทั้งหมด 5 ราย เป็นนักพัฒนาโมบายแอพพลิเคชั่นอิสระ 3 ราย และนักพัฒนาจากบริษัทพัฒนาโมบายแอพพลิเคชั่นจำนวน 2 ราย โดยมีขั้นตอนในการติดต่อสอบถามดังต่อไปนี้

- ติดต่อทางโทรศัพท์เพื่อทำการนัดพบกับผู้ที่คาดว่าสามารถให้ข้อมูล รวมทั้งอธิบายวัตถุประสงค์ของงานวิจัย และอธิบายถึงข้อมูลที่ต้องการนำมาใช้ในงานวิจัย ซึ่งได้แก่
    - ฟังก์ชั่นการทำงานทั้งหมดของโมบายแอพพลิเคชั่น
    - รายละเอียดของไฟล์ข้อมูลทั้งหมดที่ใช้ในโมบายบายแอพพลิเคชั่น
    - แบบจำลองหน้าจอของโมบายแอพพลิเคชั่น
    - แผนภาพยูสเคสและคำอธิบายสถานการณ์
- นักพัฒนาแต่ละรายได้เล่าถึงกระบวนการพัฒนาโมบายแอพพลิเคชั่นของของตนเองให้กับผู้วิจัย นักพัฒนาโมบายแอพพลิเคชั่นอิสระส่วนใหญ่จะนำเอกสารความต้องการที่ได้รับจากผู้ว่าจ้างมาทำการออกแบบเป็นแบบจำลองหน้าจอ และแผนภาพอี-อาร์ (E-R diagram) ในกรณีที่เป็นบริษัทจะมีการจัดทำเอกสารการออกแบบโมบายแอพพลิเคชั่นประเภทอื่นด้วย ได้แก่ ยูสเซอร์สตอรี่ (User Story) หรือ แผนภาพกระแสข้อมูล(Data Flow Diagram) ซึ่งเอกสารเหล่านั้นนักพัฒนาไม่สามารถให้แก่ผู้วิจัยได้ แต่นักพัฒนาสามารถให้แบบจำลองหน้าจอและอธิบายการทำงานของโมบายแอพพลิเคชั่นให้แก่ผู้วิจัยได้

จากการสอบถามนักพัฒนาโมบายแอพพลิเคชั่นทั้ง 5 ราย ทำให้สรุปได้ว่า ข้อมูลที่สามารถเก็บได้จากนักพัฒนาโมบายแอพพลิเคชั่นคือ

1) แบบจำลองหน้าจอซึ่งเป็นรูปภาพหน้าจอของโมบายแอพพลิเคชั่นที่พัฒนาเสร็จแล้ว
2) ขั้นตอนการทำงานของโมบายแอพพลิเคชั่น โดยนักพัฒนาโมบายแอพพลิเคชั่นจะทำการสาธิตและอธิบายผ่านการใช้งานโมบายแอพพลิเคชั่นจริง
3) รายละเอียดของไฟล์ข้อมูลที่ใช้ในโมบายแอพพลิเคชั่นโดยนักพัฒนาให้ข้อมูลแก่ผู้วิจัยโดยอ้างอิงจากแผนภาพอี-อาร์(E-R diagram)

ส่วนเอกสารอื่นไม่สามารถเก็บข้อมูลได้ ผู้วิจัยจึงจำเป็นต้องสร้างเครื่องมือในการเก็บข้อมูลเพื่อแปลงข้อมูลให้อยู่ในรูปแบบที่ต้องการ ซึ่งประกอบไปด้วย ฟังก์ชั่นการทำงานทั้งหมดของโมบายแอพพลิเคชั่น รายละเอียดของไฟล์ข้อมูลทั้งหมดที่ใช้ในโมบายแอพพลิเคชั่น แผนภาพยูสเคส และคำอธิบายสถานการณ์หลักของแต่ละยูสเคส

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ขั้นตอนที่ 2 สร้างเครื่องมือการเก็บข้อมูล

ผู้วิจัยต้องสร้างเครื่องมือในการเก็บข้อมูลเพื่อให้ได้ข้อมูลที่ต้องการ โดยเรียกเครื่องมือในการเก็บข้อมูลว่า “แบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น” โดยใช้หลักการของการเขียนความต้องการแบบเป้าหมายและสถานการณ์ตามวิธีการของ Choi และคณะ(Choi et al., 2006) เพื่อให้ได้ข้อมูลที่จะนำมาใช้ประมาณขนาดซอฟต์แวร์ด้วยวิธีฟังก์ชั่นพอยต์และวิธียูสเคสพอยต์ ส่วนวิธีการอ็อบเจกต์พอยต์และวิธีของ Nikunj Sakhrelia จะใช้ข้อมูลเพิ่มเติมจากแบบจำลองหน้าจอ

ผู้วิจัยได้จัดทำคำอธิบายแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นเพื่อไม่ให้เกิดความคาดเคลื่อนจากการกรอกข้อมูลซึ่งอาจจะทำให้การนำไปตีความนั้นผิดพลาด เพราะการคำนวณขนาดทั้ง 4 วิธีนั้น จะต้องใช้การวิเคราะห์ตีความจากข้อมูลที่ได้รับจากผู้ให้ข้อมูล โดยเฉพาะอย่างยิ่งการวาดแผนภาพยูสเคส หากมีการเก็บข้อมูลที่ไม่ชัดเจนจะส่งผลต่อการวาดแผนภาพยูสเคสและทำให้ผลการประมาณขนาดคาดเคลื่อนได้ ซึ่งจะต้องควบคุมความผิดพลาดในส่วนนี้

ผู้วิจัยได้แบ่งแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น เป็น 3 ส่วน

- ส่วนที่ 1 สำหรับกรอกข้อมูลทั่วไปของโมบายแอพพลิเคชั่น
- ส่วนที่ 2 สำหรับกรอกข้อมูลขั้นตอนการทำงานของฟังก์ชั่นทั้งหมดของโมบายแอพพลิเคชั่น
- ส่วนที่ 3 สำหรับกรอกข้อมูลปัจจัยความซับซ้อนของระบบ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

คำอธิบายแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น

แบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น

แบบฟอร์มส่วนที่ 1

Application Name : _____

Detail : _____

_____

_____

_____

Owner: _____ Date: _____

No. of Function: _____ No. of Screen: _____

Actual Effort: _____ Man - Hour

Data File Name	Type	No. of Data Field	No.of Subgroup

รูปที่ 3.2 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นส่วนที่ 1

แบบฟอร์มส่วนที่ 1

แบบฟอร์มส่วนนี้ผู้วิจัยได้ออกแบบขึ้นเองเพื่อเก็บข้อมูลรายละเอียดของโมบายแอพพลิเคชั่นเพื่อให้เข้าใจภาพรวมและวัตถุประสงค์หลักของโมบายแอพพลิเคชั่น รวมทั้งรายละเอียดเกี่ยวกับไฟล์ข้อมูลทั้งหมดที่ใช้ในโมบายแอพพลิเคชั่นเพื่อนำไปนับขนาดด้วยวิธีฟังก์ชั่นพอยต์

โดยกรอกข้อมูลตามองค์ประกอบดังนี้

1. Mobile Application Name คือ ชื่อของโมบายแอพพลิเคชั่น
2. Detail คือ คำอธิบายวัตถุประสงค์ของโมบายแอพพลิเคชั่น
3. Date คือ วันที่เก็บข้อมูล
4. Owner คือ ชื่อของผู้ให้ข้อมูล
5. No. of Function คือ จำนวนฟังก์ชั่นการทำงานทั้งหมดของโมบายแอพพลิเคชั่น
6. No. of Screen คือ จำนวนหน้าจอทั้งหมดของโมบายแอพพลิเคชั่น
7. Actual Effort คือ ค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่นมีหน่วยเป็น คน-ชั่วโมง (Man-Hour)

8. Data File Name คือ ชื่อของไฟล์ข้อมูล ที่ใช้ในขั้นตอนทำงานของโมบายแอพพลิเคชั่น
9. Type คือ ประเภทของไฟล์ข้อมูลหรือกลุ่มข้อมูล คำที่สามารถกรอกได้ในช่องนี้ได้แก่
    - Internal Data File กรณีที่ไฟล์ข้อมูล หรือกลุ่มของข้อมูลนั้นถูกเก็บรักษาอยู่ภายในขอบเขตของระบบ ไฟล์ประเภทนี้มันจะถูกใช้งานในการกระทำลักษณะของการ Add ,Update, Delete ในกระบวนการของ EI อย่างน้อย 1 ครั้ง
    - External Data File กรณีที่ไฟล์ข้อมูล หรือกลุ่มของข้อมูลนั้นถูกอ้างอิงใช้งานมากจากแอพพลิเคชั่นอื่นๆ หรือการแสดงข้อมูลจาก URL ไฟล์ประเภทนี้มันจะถูกใช้งานในการกระทำลักษณะของการ Show
10. No. of Data Field คือจำนวนฟิลด์ข้อมูลของไฟล์ข้อมูลหรือกลุ่มข้อมูลนั้น
11. No. of Subgroup คือ จำนวนกลุ่มข้อมูลย่อย หากไม่มีให้กรอก 0

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แบบฟอร์มส่วนที่ 2

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional

รูปที่ 3.3 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นส่วนที่ 2

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แบบฟอร์มส่วนที่ 2

สำหรับบันทึกรายละเอียดขั้นตอนการทำงานของแต่ละฟังก์ชั่น แบบฟอร์มส่วนนี้ได้ยึดหลักการเขียนความต้องการแบบระบุเป้าหมายและสถานการณ์ (Choi et al, 2001) นำมาสร้างเป็นแบบฟอร์มโดยใช้องค์ประกอบในการเขียนสถานการณ์มาสร้างเป็นตาราง คือ Subject, Verb, Target, Direction เพื่อให้ได้ข้อมูลสำหรับการนำไปนับขนาดด้วยวิธีฟังก์ชั่นพอยต์และวิธียูสเคสพอยต์ และเพิ่มช่องสำหรับกรอกข้อมูล Screen Name และ Use เพื่อนำข้อมูลส่วนนี้ไปใช้ในการประมาณขนาดด้วยวิธีอ๊อบเจกต์พอยต์ วิธียูสเคสพอยต์ และวิธีของ Nikunj Sakhrelia ตามรูปที่ 3.3

ในส่วนของคำแนะนำในการกรอกข้อมูลในช่องต่างๆ นั้น ผู้วิจัยได้ทำการกำหนดขึ้นจากคำแนะนำในการเขียนความต้องการแบบระบุเป้าหมายและสถานการณ์ (Choi et al, 2001) และได้เพิ่มตัวอย่างรวมทั้งกำหนดคำที่เป็นไปได้ในการกรอกข้อมูลในช่อง Verb, Target, Direction, Use เพื่อให้เหมาะสมกับการเก็บข้อมูลโมบายแอพพลิเคชั่นและเพื่อไม่ให้เกิดความเข้าใจที่คาดเคลื่อน

โดยกรอกข้อมูลตามองค์ประกอบดังนี้

1. Seq คือ ลำดับของฟังก์ชั่นการทำงาน กรอกข้อมูลเป็นตัวเลขเลขอารบิก
2. Function Name คือ ชื่อของฟังก์ชั่นการทำงานซึ่งสื่อถึงเป้าหมายในการทำงานที่คาดหวัง
3. Screen Name คือ ชื่อของหน้าจอในแต่ละฟังก์ชั่นการทำงานที่การกระทำนั้นเกิดขึ้น หากขั้นตอนใดเกิดขึ้นในหน้าจอเดียวกันกับขั้นตอนก่อนหน้าสามารถละการกรอกข้อมูลช่องนี้ได้
4. Step No. คือ ลำดับขั้นตอนของฟังก์ชั่นการทำงาน กรอกข้อมูลเป็นตัวเลขเลขอารบิก
5. Subject คือ ผู้กระทำการในขั้นตอนนั้น ซึ่งอาจจะเป็น ผู้ใช้งาน(User) หรือ แอพพลิเคชั่น(App) หรือ แอพพลิเคชั่นภายนอก(External App)

Chulalongkorn University

ตารางที่ 3.1 อธิบายตัวเลือกในการกรอกข้อมูลช่อง Subject

ตัวเลือก Subject	ความหมาย	ตัวอย่าง
ผู้ใช้ (User)	ผู้กระทำการที่เป็นผู้ใช้งาน	"User" Select Menu Live Radio From Tab Menu
แอพพลิเคชั่น (App)	ผู้กระทำการที่เป็นตัวแอพพลิเคชั่นที่กำลังสนใจอยู่	"App" Show Screen Live Radio
แอพพลิเคชั่นภายนอก(External App)	ผู้กระทำการที่เป็นแอพพลิเคชั่นภายนอกอื่นๆ	"External App" Send Data Image Object to App

6. Verb คือ การกระทำของขั้นตอนนั้นๆ ซึ่งผู้วิจัยได้กำหนดจากตัวอย่างกิริยาของ Choi และคณะในตาราง 2.37 แล้วเพิ่มคำกิริยาที่เป็นไปได้ในการใช้งานโมบายแอพพลิเคชั่นเข้าไป โดยคำที่สามารถกรอกได้ในช่องนี้ ได้แก่ Input, Call, Send, Receive, Pin,

Tab, Slide, Scan, Tilt, Shake, Select, Add, Update, Delete, Show หากมีการกระทำใดที่ไม่ตรงกับตัวเลือกที่ให้มาให้กรอกคำว่า Other ตามด้วยคำอธิบาย คำอธิบายพร้อมตัวอย่างของการกรอกข้อมูลในช่อง Verb แสดงในตาราง 3.2

ตารางที่ 3.2 อธิบายตัวเลือกในการกรอกข้อมูลช่อง Verb

ตัวเลือก Verb	ความหมาย	ตัวอย่าง
Input	การนำเข้าข้อมูลผ่านคีย์บอร์ด หรือไมค์โครโฟน	User "Input" Data Name, Age, Amount into Screen
Call	การเรียกใช้	App "Call" External App
Send	การส่ง	External App "Send" Data Image Object to App
Receive	การรับ	App "Receive" Data Video Name From URL
Pin	กดปักหมุด ใช้สำหรับหน้าจอแผนที่	User "Pin" Data Latitude Longtitude into Screen
Tab	การแตะหรือกดลงบนอ๊อบเจกต์ต่างๆบนหน้าจอ	User "Tab" Button Start From Screen
Slide	การสไลด์ลงบนอ๊อบเจกต์ต่างๆบนหน้าจอ	User "Slide" Data Video Name From Screen
Scan	การนำเข้าข้อมูลผ่านกล้องถ่ายรูป	User "Scan" Data Barcode
Tilt	การเอียงโทรศัพท์	User "Tilt" Phone
Shake	การเขย่าโทรศัพท์	User "Shake" Phone
Select	การตัดสินใจทางเลือกต่างๆ	User "Select" Menu Item Main menu From Tab Menu
Add	การเพิ่มข้อมูล	App "Add" Data Name, Age, Amount to internal File
Update	การอัพเดทข้อมูล	App "Update" Data Name, Age, Amount to internal File
Delete	การลบข้อมูล	App "Delete" Data Name, Age, Amount to internal File
Show	การแสดงข้อมูล	App "Show" Data Name, Age, Amount From internal File

7. Target คือ จุดมุ่งหมายของการกระทำนั้นเพื่อกระทำกับสิ่งใด หรืออาจจะหมายถึงกรรมของกิริยา(Verb) โดยแบ่งการกรอกข้อมูลเป็น 2 ส่วนคือ Type และ Detail
    - Type คือ ประเภทของจุดมุ่งหมาย คำที่สามารถกรอกได้ในช่องนี้ได้แก่ Data, Button, Screen, Item หากไม่ตรงกับตัวเลือกให้กรอก Other และกรอกรายละเอียดลงในช่อง Detail
    - Detail คือ ชื่อของจุดมุ่งหมาย หากเป็น Type อ๊อบเจกต์ให้กรอกชื่อ อ๊อบเจกต์ หากเป็น Data ให้กรอก ฟิลด์ข้อมูลทั้งหมดที่เข้าหรือออกจากระบบโดยจะไม่นับฟิลด์ข้อมูลที่ผู้ใช้ไม่สนใจ

8. Direction คือ ทิศทางของการกระทำ เพื่ออธิบายถึงตำแหน่งเริ่มต้นหรือตำแหน่งสิ้นสุดของการกระทำ โดยแบ่งการกรอกข้อมูลออกเป็น 3 ส่วนคือ Type1 ,Type2 และ Name ในส่วนนี้ผู้วิจัยได้กำหนดประเภทของของทิศทางเพิ่มเติมจากคำแนะนำของ Choi และคณะ ที่ให้กรอกได้เฉพาะแหล่งข้อมูลหรือผู้กระทำการ เนื่องจากในการในการเก็บข้อมูลผู้วิจัยได้ให้ความสนใจกับองค์ประกอบบนหน้าจอของโมบายแอพพลิเคชั่น จึงเพิ่มประเภทของทิศทางที่เป็นองค์ประกอบบนหน้าจอเพิ่มเข้าไปด้วย เพื่อให้เข้าใจการทำงานของโมบายแอพพลิเคชั่นมากขึ้น
    - Type1 ส่วนนี้จะกำหนดลักษณะของทิศทาง คำที่สามารถกรอกได้ในช่องนี้ได้แก่ From, to, into
    - Type2 คือ ประเภทของทิศทาง ซึ่งเป็นได้ 3 ลักษณะคือ ผู้กระทำการ แหล่งข้อมูลหรืออ็อบเจกต์ คำที่สามารถกรอกได้ในช่องนี้ได้แก่ Internal Data File, External Data File, App, URL, User, Tab Menu, External App, Menu Bar, Table, Screen , Report หรือหากไม่ตรงกับตัวเลือกให้กรอก Other และกรอกรายละเอียดลงในช่อง Name
    - Name คือ ชื่อของทิศทาง โดยกรอกชื่อผู้กระทำการ ชื่อแหล่งข้อมูล หรือชื่ออ็อบเจกต์
9. Use คือ การเรียกใช้โมดูลหรือฟังก์ชั่นการทำงานอื่นเพื่อให้ขั้นตอนนั้นสมบูรณ์ ผู้วิจัยเพิ่มการกรอกข้อมูลในส่วนนี้เพื่อจะนำไปใช้ในการประมาณขนาดด้วยวิธีอ๊อบเจกต์พอยต์ วิธียูสเคสพอยต์ และวิธีของ Nikinj Sakhrelia โดยแบ่งการกรอกข้อมูลออกเป็น 2 ส่วนคือ Type, Name, Module Type, Optional
    - Type คือ ประเภทของสิ่งที่ถูกเรียกใช้ ซึ่งเป็นไปได้ 2 ลักษณะ ได้แก่ Function, Module โดยมีตัวเลือกตามตาราง 3.3

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 3.3 อธิบายตัวเลือกในการกรอกข้อมูลช่อง Use Type

ตัวเลือก Use Type	**ความหมาย**
F	ฟังก์ชั่นการทำงานอื่นของโมบายแอพพลิเคชั่น
M	โมดูล (ส่วนของโปรแกรมที่เขียนขึ้นเพื่อเสริมการทำงานของฐานข้อมูล)

- Name คือ ชื่อของฟังก์ชั่นการทำงานหรือโมดูล
- Module Type คือ ประเภทของโมดูล กรณีที่ระบุ Type เป็นโมดูล ให้ระบุประเภทของโมดูลโดยตัวเลือกแสดงในตาราง 3.4

ตารางที่ 3.4 อธิบายตัวเลือกในการกรอกข้อมูลช่อง Module Type

ตัวเลือก Module Type	**ความหมาย**
A	เอพีไอ (API)
B	เซอร์เวอร์เอพีไอ (Server API)
C	เอพีไอที่ต้องปรับแต่ง
D	ไลบรารี่ (Library)

- Optional คือ การเรียกใช้ฟังก์ชั่นการทำงานหรือโมดูลนั้นเป็นลักษณะทางเลือกหรือไม่ ถ้าใช่ให้กรอกเครื่องหมายถูกลงในช่อง Optional

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

แบบฟอร์มส่วนที่ 3

คำชี้แจง ระบุถึงปัจจัยที่ต้องคำนึงถึงและส่งผลต่อความซับซ้อนในการพัฒนา
โดยที่ผู้ให้ข้อมูลจะต้องนึกย้อนไปถึงอิทธิพลของปัจจัยต่างๆในช่วงของการออกแบบระบบ
ระดับอิทธิพลของปัจจัยที่เป็นไปได้ คือ 0 ถึง 5 โดยให้กรอกตามคำแนะนำในเอกสารที่แนบมาด้วย

ลำดับ	ปัจจัย	ระดับอิทธิพล	ลำดับ	ปัจจัย	ระดับอิทธิพล
F1	Data Communication		T5	Reusable code	
F2	Distribution Data Processing		T6	Easy to install	
F3	Performance		T7	Easy to use	
F4	Heavily Used Configuration		T8	Portable	
F5	Transaction Rate		T9	Easy to change	
F6	On-line Data Entry		T10	Concurrent	
F7	End-User Efficiency		T11	Security features	
F8	Online Update		T12	Access for third-parties	
F9	Complex Processing		T13	Special training required	
F10	Reusability		E1	Familiar with development method used	
F11	Installation Ease		E2	Application experience	
F12	Operational Ease		E3	Object-Oriented experience	
F13	Multiple Sites		E4	Lead analyst capability	
F14	Facilitate change		E5	Motivation	
T1	Distributed System		E6	Stable Requirements	
T2	Response adjectives		E7	Part-time workers	
T3	End-user efficiency		E8	Difficulty of programming language	
T4	Complex processing				

รูปที่ 3.4 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นส่วนที่ 3

แบบฟอร์มส่วนที่ 3

ในส่วนนี้จะถูกกรอกโดยผู้ให้ข้อมูล ซึ่งจะเป็นการเก็บข้อมูลปัจจัยต่างๆที่มีอิทธิพลต่อการพัฒนาโมบายแอพพลิเคชั่น สำหรับปัจจัย F1-F14 จะถูกนำไปใช้ในการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ ปัจจัย T1-T13 และ E1-E8 จะถูกนำไปใช้ในการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ โดยผู้กรอกข้อมูลจะอ่านคำอธิบายในการกรอกข้อมูลตามตาราง เอกสารภาคผนวก ก

ขั้นตอนที่ 3 การดำเนินการเก็บข้อมูล

ขั้นตอนในการเก็บข้อมูลผู้วิจัยได้ทำการเก็บข้อมูลนักพัฒนาโมบายแอพพลิเคชั่นทั้งหมด 7 ราย เป็นนักพัฒนาโมบายแอพพลิเคชั่นอิสระ 5 ราย และนักพัฒนาจากบริษัทพัฒนาโมบายแอพพลิเคชั่น 2 ราย โดยที่ 5 จาก 7 รายเป็นนักพัฒนาคนเดียวกับที่ผู้วิจัยได้ติดต่อไปในขั้นตอนการสอบถามความเป็นไปได้ในการได้ข้อมูล

เมื่อผู้วิจัยได้พบกับนักพัฒนาโมบายแอพพลิเคชั่น ซึ่งเป็น ผู้ให้ข้อมูลแล้ว มีขั้นตอนในการเก็บข้อมูลดังต่อไปนี้

1. ผู้วิจัยสอบถามข้อมูลเบื้องต้นของโมบายแอพพลิเคชั่น ชื่อโมบายแอพพลิเคชั่น วัตถุประสงค์ของโมบายแอพพลิเคชั่น จำนวนฟังก์ชั่นทั้งหมดของโมบายแอพพลิเคชั่น จำนวนหน้าจอทั้งหมดของโมบายแอพพลิเคชั่น และกรอกข้อมูลลงในแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 1
2. ผู้วิจัยสอบถามข้อมูลรายละเอียดไฟล์ข้อมูลที่ใช้ในโมบายแอพพลิเคชั่นนี้ ได้แก่ ชื่อไฟล์ ประเภทของไฟล์ จำนวนของฟิลด์ข้อมูล และจำนวนของกลุ่มข้อมูลย่อย ซึ่งนักพัฒนาจะอ้างอิงข้อมูลจากแผนภาพอี-อาร์ และแจ้งข้อมูลให้แก่ผู้วิจัยเพื่อกรอกข้อมูลลงในแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 1
3. ผู้วิจัยขอให้ผู้ให้ข้อมูลสาธิตการใช้งานโมบายแอพพลิเคชั่นครบทุกฟังก์ชั่น
4. ผู้วิจัยทำการสังเกตการใช้งานโมบายแอพพลิเคชั่น
5. ผู้วิจัยได้ลองใช้งานโมบายแอพพลิเคชั่นทุกฟังก์ชั่นตามที่ผู้ให้ข้อมูลสาธิตให้ชม
6. ผู้วิจัยชี้แจงแก่ผู้ให้ข้อมูลถึงรายละเอียดของข้อมูลที่ต้องการเก็บในแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 2 และ ส่วนที่ 3 พร้อมทั้งนำคำอธิบายแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นและตัวอย่างการกรอกข้อมูลให้ผู้ให้ข้อมูลอ่านทำความเข้าใจ
7. การกรอกข้อมูลลงในแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 2 จะแบ่งเป็น 2 กรณี คือ
   - กรณีผู้ให้ข้อมูลสะดวกในการกรอกข้อมูล
   - กรณีผู้ให้ข้อมูลไม่สะดวกในการกรอกข้อมูล ผู้วิจัยจะทำการกรอกข้อมูลลงในแบบฟอร์มการเก็บข้อมูลเอง
8. ผู้ให้ข้อมูลทำการกรอกข้อมูลลงในแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 3 ซึ่งเป็นการเก็บข้อมูลปัจจัยที่มีอิทธิพลในการพัฒนาโมบายแอพพลิเคชั่นนี้ ซึ่งในขั้นตอนการเก็บข้อมูลแอพพลิเคชั่นนี้ได้รับการพัฒนาเสร็จสิ้นแล้ว ดังนั้นในขั้นตอนนี้

ผู้ให้ข้อมูลจะต้องทำการนึกย้อนไปถึงช่วงก่อนการพัฒนาโมบายแอพพลิเคชั่นว่ามีปัจจัยใดที่มีอิทธิพลและผู้ให้ข้อมูลจะต้องคำนึงถึงในการพัฒนาโมบายแอพพลิเคชั่นนี้บ้าง

9. หลังจากกรอกข้อมูลครบถ้วนผู้วิจัยจะอ่านทวนข้อมูลทั้งหมดอีกครั้งเพื่อทำความเข้าใจหากมีข้อสงสัยจะทำการสอบถามกับผู้ให้ข้อมูลจนเข้าใจการทำงานทั้งหมด

ขั้นตอนที่ 4 การนำข้อมูลจากแบบฟอร์มการเก็บข้อมูลมาใช้ประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีการทั้ง 4 วิธี

หลังจากทำการเก็บรวบรวมข้อมูลผู้วิจัยจะนำข้อมูลในแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นทั้ง 3 ส่วน และแบบจำลองหน้าจอ มาทำการประมาณขนาดโมบายแอพพลิเคชั่น โดยใช้ฟิลด์ต่างๆดังแสดงในตาราง 3.5-3.6

ตารางที่ 3.5 แสดงฟิลด์ข้อมูลในแบบฟอร์มการเก็บข้อมูลที่ใช้ในการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์และวิธีอ็อบเจกต์พอยต์

วิธีประมาณขนาดซอฟต์แวร์	ฟิลด์ข้อมูลจากแบบฟอร์มการเก็บข้อมูล			แบบจำลองหน้าจอ	วัตถุประสงค์เพื่อ
	ส่วนที่ 1	ส่วนที่ 2	ส่วนที่ 3		
วิธีฟังก์ชั่นพอยต์	Data File Name, Type, No.of Data Field, No.of Subgroup				ระบุ ILF,ELF นับ DET, RET
		Subject, Verb, Target, Direction			ระบุ EI,EO,EQ
		Verb,Target, Direction			นับ DET, FTR
			ปัจจัย F1-F14		ค่าความซับซ้อนของระบบ
วิธีอ็อบเจกต์พอยต์				✓	นับจำนวนหน้าจอ
		Screen Name, Direction,			นับจำนวนแหล่งข้อมูลที่ใช้
		Use			นับจำนวนโมดูล

ตารางที่ 3.6 แสดงฟิลด์ข้อมูลในแบบฟอร์มการเก็บข้อมูลที่ใช้ในการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์และวิธีของ Nikunj Sakhrelia

วิธีประมาณขนาดซอฟต์แวร์	ฟิลด์ข้อมูลจากแบบฟอร์มการเก็บข้อมูล			แบบจำลองหน้าจอ	วัตถุประสงค์เพื่อ
	ส่วนที่ 1	ส่วนที่ 2	ส่วนที่ 3		
วิธียูสเคสพอยต์		Function Name, Use			ระบุยูสเคส และยูสเคสย่อย
		Subject, Verb, Target			ระบุผู้กระทำการ
		Subject, Verb, Target, Direction			นับจำนวนรายการ
			ปัจจัย T1-T13		ค่าความซับซ้อนเชิงเทคนิค
			ปัจจัย E1-E8		ค่าความซับซ้อนทางสภาพแวดล้อม
วิธีของ Nikunj Sakhrelia				✓	นับองค์ประกอบบนหน้าจอ
		Subject, Verb, Target, Direction, Use			พิจารณาขั้นตอนการทำงานของหน้าจอ

ในการอธิบายขั้นตอนการประมาณขนาดจะใช้การอธิบายโดยข้อมูลแอพพลิเคชั่นที่สมมติขึ้น เนื่องจากข้อมูลแอพพลิเคชั่นจริงไม่สามารถเปิดเผยได้ โดยแอพพลิเคชั่นสมมติมีรายละเอียดของโมบายแอพพลิเคชั่นตามแบบฟอร์มดังต่อไปนี้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น

แบบฟอร์มส่วนที่ 1

Application Name : แอพพลิเคชั่นตัวอย่าง

Detail : แอพพลิเคชั่นนี้มีประโยชน์สำหรับผู้ใช้งานที่ต้องการอ่านข้อมูลต่างๆ
ที่ถูกเก็บไว้ในรูปแบบคิวอาร์โค้ดไม่ว่าจะเป็น ข้อมูล URL ของเว็บไซต์
ข้อความ, เบอร์โทรศัพท์ โดยการใช้งานให้นำโทรศัพท์ไปสแกนรูปภาพคิวอาร์โค้ด
จากนั้นแอพพลิเคชั่นก็จะแสดงข้อมูลที่ถูกเก็บไว้ในคิวอาร์โค้ดให้กับผู้ใช้งาน
และผู้ใช้ยังสามารถเรียกดูข้อมูลประวัติการแสกนคิวอาร์ โค้ด ได้

Owner: นักพัฒนาตัวอย่าง Date: 20/1/2014

No. of Function: 2 No. of Screen: 4

Accual Effort: 24 Man - Hour

Data File Name	Type	No. of Data Field	No.of Subgroup
History	Internal Data File	3	0

รูปที่ 3.5 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 1
ของโมบายแอพพลิเคชั่นตัวอย่าง

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แบบฟอร์มส่วนที่ 2

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
1	สแกนคิวอาร์โค้ดด้วยกล้อง	หน้าจอหลัก	1	User	Tap	Button	Start Scan	From	Screen					
		หน้าจอกล้อง	2	User	Scan	Data	QR Code	into	Screen		M	QR Code Reader	B	
			3	App	Other: Generate	Data	QR Code				M	QR Code Reader	B	
			4	App	Add	Data	Detail, Type	to	Internal Data File	History	M	Core data	D	
			5	App	Show	Data	Detail	From	Internal Data File	History	M	Core data	D	
		หน้าจอผลการสแกน	6	User	Tap	Data	Detail	From	Screen					
			7	App	Send	Data	Detail	to	External App	Web browser				

รูปที่ 3.6 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 2 ของโมบายแอพพลิเคชั่นตัวอย่าง

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ History	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

รูปที่ 3.7 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 2 ของโมบายแอพพลิเคชั่นตัวอย่าง (ต่อ)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แบบฟอร์มส่วนที่ 3

คำชี้แจง ระบุถึงปัจจัยที่ต้องคำนึงถึงและส่งผลต่อความซับซ้อนในการพัฒนา
โดยที่ผู้ให้ข้อมูลจะต้องนึกย้อนไปถึงอิทธิพลของปัจจัยต่างๆในช่วงของการออกแบบระบบ
ระดับอิทธิพลของปัจจัยที่เป็นไปได้ คือ 0 ถึง 5 โดยให้กรอกตามคำแนะนำในเอกสารที่แนบมาด้วย

ลำดับ	ปัจจัย	ระดับอิทธิพล	ลำดับ	ปัจจัย	ระดับอิทธิพล
F1	Data Communication	0	T5	Reusable code	0
F2	Distribution Data Processing	0	T6	Easy to install	0
F3	Performance	1	T7	Easy to use	0
F4	Heavily Used Configuration	0	T8	Portable	0
F5	Transaction Rate	0	T9	Easy to change	1
F6	On-line Data Entry	5	T10	Concurrent	0
F7	End-User Efficiency	1	T11	Security features	0
F8	Online Update	0	T12	Access for third-parties	0
F9	Complex Processing	2	T13	Special training required	0
F10	Reusability	0	E1	Familiar with development method used	3
F11	Installation Ease	0	E2	Application experience	3
F12	Operational Ease	0	E3	Object-Oriented experience	3
F13	Multiple Sites	0	E4	Lead analyst capability	2
F14	Facilitate change	1	E5	Motivation	3
T1	Distributed System	0	E6	Stable Requirements	5
T2	Response adjectives	1	E7	Part-time workers	0
T3	End-user efficiency	1	E8	Difficulty of programming language	1
T4	Complex processing	2			

รูปที่ 3.8 แสดงแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น ส่วนที่ 3
ของโมบายแอพพลิเคชั่นตัวอย่าง

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

รูปที่ 3.9 แสดงหน้าจอหลัก(ซ้าย) และหน้าจอกล้อง(ขวา) ของโมบายแอพพลิเคชั่นตัวอย่าง

รูปที่ 3.10 แสดงหน้าจอผลการสแกน(ซ้าย)และหน้าจอ History(ขวา)
ของโมบายแอพพลิเคชั่นตัวอย่าง

ขั้นตอนที่ 4.1 การประมาณขนาดด้วยวิธีฟังก์ชั่นพอยต์

วิธีการนี้จะต้องระบุประเภทของฟังก์ชั่น(Function Type) ทั้ง 5 ประเภท คือ

1) ข้อมูลเชิงตรรกะภายใน(ILF)
2) ข้อมูลที่ต้องการจากภายนอก(ELF)
3) ข้อมูลนำเข้าจากภายนอก(EI)
4) ข้อมูลส่งออกภายนอก(EO)
5) การสอบถามภายนอก(EQ)

ซึ่งในการประมาณขนาดด้วยวิธีฟังก์ชั่นพอยต์จะต้องใช้ข้อมูลจากแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นส่วนที่ 1 ส่วนที่2 และส่วนที่ 3 ซึ่งมีขั้นตอนดังนี้

1. ระบุฟังก์ชั่นเชิงข้อมูล ข้อมูลเชิงตรรกะภายใน และข้อมูลที่ต้องการจากภายนอก จะพิจารณาจากข้อมูลในช่อง Type ของแบบฟอร์มส่วนที่ 1
   - หากมี Type เป็น Internal Data File ไฟล์นั้นจะถูกระบุเป็น ILF
   - หากมี Type เป็น External Data File ไฟล์นั้นจะถูกระบุเป็น ELF
2. ในการพิจารณาความซับซ้อนของฟังก์ชั่นทั้ง 2 จะต้องพิจารณาจากจำนวนองค์ประกอบข้อมูล(DET) และจำนวนเรคคอร์ดข้อมูล(RET) โดยมีวิธีการนับคือ
   - นับ DET จากข้อมูลในช่อง No.of Field
   - นับ RET จากข้อมูลในช่อง No.of Subgroup ถ้า No.of Subgroup เป็น 0 ให้นับ 1 RET ถ้ามี 1 Subgroup จะนับเป็น 2 RET

Data File Name	Type	No. of Data Field	No.of Subgroup
History	Internal Data File	3	0

ระบุ ILF หรือ EIF | นับ DET | นับ RET

รูปที่ 3.11 แสดงการระบุฟังก์ชั่นเชิงข้อมูลของโมบายแอพพลิเคชั่นตัวอย่าง

จากโมบายแอพพลิเคชั่นตัวอย่าง ไฟล์ History จะถูกระบุเป็น ILF มี 3 DET และ 1 RET

3. ระบุฟังก์ชั่นเชิงรายการ การกระทำที่จะถูกนับเป็นฟังก์ชั่นเชิงรายการจะต้องมีคุณสมบัติครบ 2 ข้อ ดังนี้

1) ต้องเป็นกระบวนการที่เป็นหน่วยเล็กที่สุดของกิจกรรมที่มีความหมายให้กับผู้ใช้ หรือเรียกว่า กระบวนการพื้นฐาน (Elementary Process)

2) จะต้องเป็นการกระทำที่มีความหมายในตัวเอง

จะพิจารณาจากข้อมูลในช่อง Verb, Target และ Direction ของขั้นตอนในกระบวนการพื้นฐานนั้น

- หากมี Verb ดังต่อไปนี้ Add, Update, Delete และมี Target เป็นประเภท Data From Direction ที่เป็น Internal File ขั้นตอนนั้นจะถูกระบุเป็น EI
- หากมี Verb ดังต่อไปนี้ Show, Send และมี Target เป็นประเภท Data โดยจะต้องผ่านการประมวลผล การคำนวณทางคณิตศาสตร์ หรือการเปลี่ยนแปลงพฤติกรรมของระบบ ขั้นตอนนั้นจะถูกระบุเป็น E
- หากมี Verb ดังต่อไปนี้ Show และมี Target เป็นประเภท Data โดยไม่มีการประมวลผลหรือการคำนวณทางคณิตศาสตร์ ขั้นตอนนั้นจะถูกระบุเป็น EQ

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ History	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

ระบุกระบวนการพื้นฐาน

ระบุ EI, EO ,EQ

รูปที่ 3.12 แสดงการระบุฟังก์ชั่นเชิงรายการของโมบายแอพพลิเคชั่นตัวอย่าง

จากโมบายแอพพลิเคชั่นตัวอย่าง

- ลำดับขั้นตอนที่ 1 และ 2 นับเป็น 1 กระบวนการพื้นฐาน ซึ่งวัตถุประสงค์หลักเพื่อแสดงข้อมูลจากไฟล์ข้อมูล History กระบวนการนี้จะถูกระบุเป็น 1 EQ
- ลำดับขั้นตอนที่ 3 ถึง 4 นับเป็น 1 กระบวนการพื้นฐาน ซึ่งวัตถุประสงค์หลักเพื่อแสดงข้อมูลจากไฟล์ข้อมูล History โดยส่งข้อมูลไปยังแอพพลิเคชั่นภายนอก กระบวนการนี้จะถูกระบุเป็น 1 EO

4. ในการพิจารณาความซับซ้อนของฟังก์ชั่นเชิงรายการ จะต้องพิจารณาจากจำนวนองค์ประกอบข้อมูล(DET) และจำนวนไฟล์ข้อมูลอ้างอิง(FTR) โดยมีวิธีการนับคือ
    - นับ 1 DET จากแต่ละข้อมูลที่ไม่ซ้ำกันในช่อง Target ในการกระทำของฟังก์ชั่นเชิงรายการนั้นๆ
    - นับ 1 DET จากแต่ละขั้นตอนที่จำเป็นจะต้องเกิดขึ้นเพื่อให้ฟังก์ชั่นเชิงรายการนั้นทำงานได้สมบูรณ์ ได้แก่ การกระทำที่จะต้องเกิดขึ้นก่อน การแสดง Confirm message หรือ Complete Message เช่น ขั้นตอนการ User Tap Button “done” และ App Show “Complete Message”
    - นับ 1 DET สำหรับ 1 ฟิลด์ข้อมูลในช่อง Target ที่ต้องกรอกเพื่อประมวลผลการทำงานของ ข้อมูลส่งออกภายนอก และการสอบถามภายนอก
    - นับ 1 FTR จากแต่ละไฟล์ข้อมูลที่เกี่ยวข้องกับขั้นตอนการทำงานของฟังก์ชั่นเชิงรายการ

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ History	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

EQ

EO

นับ DET

นับ FTR

รูปที่ 3.13 แสดงการนับ DET RET ของโมบายแอพพลิเคชั่นตัวอย่าง

จากโมบายแอพพลิเคชั่นตัวอย่าง

- EQ มี 4 DET 1 FTR
- EO มี 2 DET 1 FTR

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

5. เปรียบเทียบความซับซ้อนของฟังก์ชั่นเชิงข้อมูลกับตารางแสดงระดับความซับซ้อนของฟังก์ชั่นเชิงข้อมูล

ตารางที่ 3.6 แสดงระดับความซับซ้อนของฟังก์ชั่นเชิงข้อมูล (IFPUG, 2000)

RET	DET		
	1 - 19	**21 – 50**	**มากกว่า 50**
1	ต่ำ	ต่ำ	ปานกลาง
2-5	ต่ำ	ปานกลาง	สูง
ตั้งแต่ 6 ขึ้นไป	ปานกลาง	สูง	สูง

ตัวอย่าง ไฟล์ History เป็น ILF ที่มี 3 DET และ 1 RET จึงมีระดับความซับซ้อน “ต่ำ”

6. เปรียบเทียบความซับซ้อนของฟังก์ชั่นเชิงรายการกับตารางแสดงระดับความซับซ้อนของฟังก์ชั่นเชิงรายการ

ตารางที่ 3.7 ตารางแสดงระดับความซับซ้อนของฟังก์ชั่นเชิงรายการ EI (IFPUG,2000)

FTR	DET		
	2 - 4	**5 – 15**	**มากกว่า 15**
<2	ต่ำ	ต่ำ	ปานกลาง
2	ต่ำ	ปานกลาง	สูง
>2	ปานกลาง	สูง	สูง

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 3.8 ตารางแสดงระดับความซับซ้อนของฟังก์ชั่นเชิงรายการ EO และ EQ (IFPUG,2000)

FTR	DET		
	1- 5	**6 – 19**	**มากกว่า 19**
<2	ต่ำ	ต่ำ	ปานกลาง
2 - 3	ต่ำ	ปานกลาง	สูง
>3	ปานกลาง	สูง	สูง

จากโมบายแอพพลิเคชั่นตัวอย่าง

- ฟังก์ชั่น EQ จากขั้นตอนที่ 1 และ 2 ของฟังก์ชั่นดูข้อมูลคิวอาร์โค้ดจากไฟล์ History มีระดับความซับซ้อน ไฟล์ History มี 4 DET 1 FTR จึงมีระดับความซับซ้อน “ต่ำ”
- ฟังก์ชั่น EQ จากขั้นตอนที่ 3 และ 4 ของฟังก์ชั่นดูข้อมูลคิวอาร์โค้ดจากไฟล์ History มีระดับความซับซ้อน ไฟล์ History มี 2 DET 1 FTR จึงมีระดับความซับซ้อน “ต่ำ”

7. กรอกจำนวนของฟังก์ชั่นตามประเภทของฟังก์ชั่นและค่าปัจจัยความซับซ้อนปัจจัย F1-F14 จากแบบฟอร์มการเก็บข้อมูลส่วนที่ 3 ลงในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ตามที่แสดงในรูปที่ 3.10

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

เอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์

Application Name : แอพพลิเคชั่นตัวอย่าง

ประเภท	ระดับความซับซ้อน						รวม
	ต่ำ		ปานกลาง		สูง		
EI		x3		x4		x6	
EO	1	x4		x5		x7	4
EQ	1	x3		x4		x6	3
ILFs		x7		x10		x15	
ELFs		x5		x7		x10	

Total Unadjusted Function Points (TUFP): 7

ลำดับ	ปัจจัย	คะแนน	ลำดับ	ปัจจัย	คะแนน
F1	Data Communication	0	F8	Online Update	0
F2	Distribution Data Processing	0	F9	Complex Processing	2
F3	Performance	1	F10	Reusability	0
F4	Heavily Used Configuration	0	F11	Installation Ease	0
F5	Transaction Rate	0	F12	Operational Ease	0
F6	On-line Data Entry	5	F13	Multiple Sites	0
F7	End-User Efficiency	1	F14	Facilitate change	1

System Complexity: _____

Adjusted System Complexity = 0.65 + (0.01 * _____)

Total Adjusted Function Points (TAFP): _____ * _____ =

รูปที่ 3.14 แสดงการกรอกข้อมูลลงในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์

ขั้นตอนที่ 4.2 การประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์

วิธีการนี้จะต้องระบุอ็อบเจกต์ของซอฟต์แวร์ ทั้ง 3 ประเภท คือ

1) หน้าจอ
2) รายงาน
3) โมดูล 3 จีแอล

ซึ่งจะทำการพิจารณาจากเอกสารแบบจำลองหน้าจอ และข้อมูลในแบบฟอร์มการเก็บข้อมูลส่วนที่ 2 ซึ่งมีขั้นตอนดังนี้

1. นับ 1 หน้าจอจากแต่ละหน้าจอในเอกสารแบบจำลองหน้าจอ ตามรูปที่ 3.11

รูปที่ 3.15 แสดงหน้าจอทั้งหมดของโมบายแอพพลิเคชั่นตัวอย่าง

จากโมบายแอพพลิเคชั่นตัวอย่าง มีทั้งหมด 4 หน้าจอ คือหน้าจอหลัก หน้าจอกล้อง หน้าผลการสแกน หน้าจอ History

2. นับ 1 รายงานจากแต่ละหน้าจอในเอกสารแบบจำลองที่มีการแสดงผลในลักษณะของรายงาน

จากโมบายแอพพลิเคชั่นตัวอย่าง ไม่มีรายงาน

3. ระบุความซับซ้อนของหน้าจอและรายงาน จากส่วนของหน้าจอหรือส่วนของรายงาน และจำนวนของแหล่งข้อมูล ซึ่งมีการนับดังนี้

   - ในการนับจำนวนส่วนของหน้าจอ(View) เนื่องจากโมบายแอพพลิเคชั่นมีหน้าจอที่มีขนาดเล็กโดยปกติจะเทียบ 1หน้าจอ(Screen) เป็น 1 ส่วนของหน้าจอ ในกรณีที่จะมีส่วนของหน้าจอมากกว่า 1 เช่น กรณีที่แบ่งหน้าจอเป็นส่วนสำหรับการกรอกข้อมูลและส่วนสำหรับแสดงผลข้อมูล
   - ส่วนของรายงาน จะถูกแบ่งเป็น ส่วนหัวรายงาน ส่วนแสดงรายละเอียด และส่วนท้ายของรายงาน โดยในแต่ละส่วนอาจจะมีการแบ่งเป็นส่วนย่อยได้เช่น กรณีส่วนของรายละเอียดมีการแสดงรายละเอียดที่เป็นแผนภาพ และข้อความ กรณีนี้จะแบ่งส่วนรายละเอียดเป็น 2 ส่วน
   - การนับจำนวนแหล่งข้อมูล แต่ละแหล่งข้อมูลช่อง Direction ที่เป็น Internal Data File หรือ External Data File ของการทำงานบนหน้าจอนั้นจะถูกนับเป็น 1 แหล่งข้อมูล

รูปที่ 3.16 แสดงการนับส่วนของหน้าจอของโมบายแอพพลิเคชั่นตัวอย่าง

นับไฟล์ข้อมูลที่ใช้

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ History	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

รูปที่ 3.17 แสดงการนับจำนวนแหล่งข้อมูลข้อมูลบนหน้าจอของโมบายแอพพลิเคชั่นตัวอย่าง

จากโมบายแอพพลิเคชั่นตัวอย่าง หน้าจอ History ประกอบไปด้วย 1 ส่วนของหน้าจอ และ 1 ไฟล์ข้อมูล

4. นับ 1 โมดูลจาก ส่วนของโปรแกรมที่ถูกเขียนขึ้นเพื่อเสริมการทำงานของฐานข้อมูล โดยพิจารณาจากแบบฟอร์มการเก็บข้อมูลส่วนที่ 2 ในช่อง Module Type หากข้อมูลเป็นตัวอักษร M จะนับเป็น 1 โมดูลโดยจากจำนวนโมดูลที่ไม่ซ้ำกัน

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ History	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

นับจำนวนโมดูล

รูปที่ 3.18 แสดงการนับโมดูลของโมบายแอพพลิเคชั่นตัวอย่าง

จากโมบายแอพพลิเคชั่นตัวอย่าง นับ 2 โมดูล คือ โมดูล Core Data และโมดูล Table View

5. เปรียบเทียบความซับซ้อนของหน้าจอและรายงานจากตารางการแบ่งกลุ่มอ็อบเจกต์

ตารางที่ 3.9 การแบ่งกลุ่มอ็อบเจกต์ (Issa et al., 2007)

หน้าจอ (Screen)				**รายงาน (Report)**			
จำนวนหน้าจอที่ปรากฏ (Number of Views Contained)	**จำนวนของแหล่งข้อมูลตาราง (Source of Data Tables)**			**จำนวนส่วนของรายงาน (Number of Sections Contained)**	**จำนวนของแหล่งข้อมูลตาราง (Source of Data Tables)**		
	<4 (<2 Srvr <3 Clnt)	**≤8 (2/3 Srvr 3-5 Clnt)**	**>8 (>3 Srvr >5 Clnt)**		**<4 (<2 Srvr <3 Clnt)**	**≤8 (2/3 Srvr 3-5 Clnt)**	**>8 (>3 Srvr >5 Clnt)**
<3	ง่าย	ง่าย	ปานกลาง	0 หรือ 1	ง่าย	ง่าย	ปานกลาง
3-7	ง่าย	ปานกลาง	ยาก	2 หรือ 3	ง่าย	ปานกลาง	ยาก
≥8	ปานกลาง	ยาก	ยาก	>3	ปานกลาง	ยาก	ยาก

หมายเหตุ srvr หมายถึง เครื่องผู้ให้บริการ (Server) clnt หมายถึง เครื่องผู้รับบริการ (Client)

จากโมบายแอพพลิเคชั่นตัวอย่าง หน้าจอ History มี 1 ส่วนหน้าจอ และ 1 แหล่งข้อมูล จึงเป็นอ็อบเจกต์กลุ่ม “ง่าย”

6. กรอกจำนวนของอ็อบเจกต์ลงในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

เอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีอ๊อบเจกต์พอยต์

Application Name : แอพพลิเคชั่นตัวอย่าง

ประเภท	ระดับความซับซ้อน						รวม
	ต่ำ		ปานกลาง		สูง		
Screen	1	x1	0	x2		x3	
Report		x2	0	x5		x8	
3GL Module					2	x10	

Object Point: _____

% Reuse: 0

New Object Pont (NOP) = $\frac{(0)-100}{100} \times (_____)$ =

รูปที่ 3.19 แสดงการกรอกข้อมูลในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์

ขั้นตอนที่ 4.3 การประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

การประมาณขนาดโดยวิธีนี้จะนำข้อมูลจากแบบฟอร์มการกรอกข้อมูลส่วนที่ 2 มาวาดแผนภาพยูสเคสเพื่อให้สามารถประมาณขนาดโมบายแอพพลิเคชั่นได้จาก ผู้กระทำการ และยูสเคส ขั้นตอนในการวาดแผนภาพยูสเคสเป็นดังนี้

1. ฟังก์ชั่นทั้งหมดจะถูกกำหนดให้เป็นยูสเคสหลักของระบบ โดยหากฟังก์ชั่นใดถูกเรียกใช้ในขั้นตอนการทำงานของฟังก์ชั่นอื่น จะถูกเขียนเป็นยูสเคสย่อย ที่มีความสัมพันธ์แบบ Include หรือ Extend กับยูสเคสหลัก โดยมีเกณฑ์ในการพิจารณาดังนี้
    - ยูสเคสแบบ Include คือ ฟังก์ชั่นการทำงานที่ถูกใช้งานโดยฟังก์ชั่นอื่นเพื่อให้การทำงานของฟังก์ชั่นนั้นสมบูรณ์ และฟังก์ชั่นนี้อาจถูกเรียกใช้ซ้ำจากฟังก์ชั่นที่แตกต่างกัน ซึ่งพิจารณาจากข้อมูลในช่อง Use และ Optional หากฟังก์ชั่นใดปรากฏอยู่ในช่อง Use ในขั้นตอนของฟังก์ชั่นอื่นและไม่มีการทำเครื่องหมายถูกในช่อง Optional หมายความว่า ฟังก์ชั่นนั้นจะถูกกำหนดเป็น ยูสเคสแบบ Include

- ยูสเคสแบบ Extend คือ ฟังก์ชั่นที่ถูกเรียกใช้โดยฟังก์ชั่นอื่นในลักษณะของตัวเลือก โดยจะเรียกใช้หรือไม่ก็ได้ หากไม่เรียกใช้การทำงานของฟังก์ชั่นนั้นก็ยังสมบูรณ์ได้ ซึ่งพิจารณาจากข้อมูลในช่อง Use และ Optional หากฟังก์ชั่นใดปรากฏอยู่ในช่อง Use ในขั้นตอนของฟังก์ชั่นอื่นและมีการทำเครื่องหมายถูกในช่อง Optional หมายความว่า ฟังก์ชั่นนั้นจะถูกกำหนดเป็น ยูสเคสแบบ Extend

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
1	สแกนคิวอาร์โค้ดด้วยกล้อง	หน้าจอหลัก	1	User	Tap	Button	Start Scan	From	Screen					
		หน้าจอกล้อง	2	User	Scan	Data	QR Code	into	Screen		M	QR Code Reader	B	
			3	App	Other: Generate	Data	QR Code				M	QR Code Reader	B	
			4	App	Add	Data	Detail, Type	to	Internal Data File	History	M	Core data	D	
			5	App	Show	Data	Detail	From	Internal Data File	History	M	Core data	D	
		หน้าจอผลการสแกน	6	User	Tap	Data	Detail	From	Screen					
			7	App	Send	Data	Detail	to	External App	Web browser				

ระบุยูสเคสย่อย

ระบุยูสเคส

ระบุผู้กระทำ

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ History	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

รูปที่ 3.20 แสดงการแสดงการระบุยูสเคสและผู้กระทำการของโมบายแอพพลิเคชั่นตัวอย่าง

จากโมบายแอพพลิเคชั่นตัวอย่าง แอพพลิเคชั่นนี้มี 2 ยูสเคสหลัก คือ

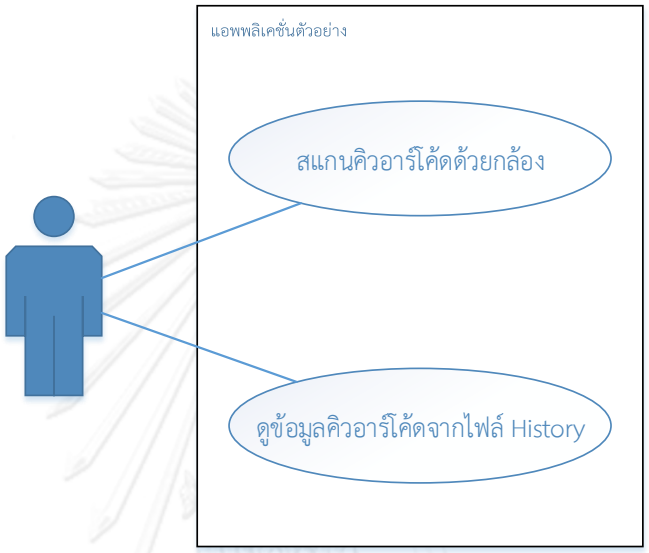
- ยูสเคสสแกนคิวอาร์โค้ดด้วยกล้อง
- ยูสเคสดูข้อมูลคิวอาร์โค้ดจากไฟล์ History

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2. Subject ที่เป็นผู้ใช้งานหลักของฟังก์ชั่นจะถูกกำหนดให้เป็นผู้กระทำการของระบบ

   จากโมบายแอพพลิเคชั่นตัวอย่าง แอพพลิเคชั่นนี้มี 1 ผู้กระทำการ คือ ผู้ใช้งาน(User)

3. นำยูสเคสและผู้กระทำการมาวาดแผนภาพยูสเคส



รูปที่ 3.21 แสดงแผนภาพยูสเคสของโมบายแอพพลิเคชั่นตัวอย่าง

4. การพิจารณาความซับซ้อนของยูสเคสจะพิจารณาจากจำนวนรายการที่เกิดภายใต้ยูสเคส ซึ่งรายการ (Transaction) จะไม่นับ 1 การกระทำหรือขั้นตอนเป็น 1 รายการแต่รายการจะถูกนับเมื่อเป็นการกระทำจากผู้ใช้ที่ระบบตอบสนองและนำเสนอผลลัพธ์จากการประมวลผลแล้วเปรียบเทียบกับตารางแสดงค่าถ่วงน้ำหนักของยูสเคส

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ history	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

นับจำนวนรายการ

รูปที่ 3.22 แสดงการนับจำนวนรายการของโมบายแอพพลิเคชั่นตัวอย่าง

จากโมบายแอพพลิเคชั่นตัวอย่าง ยูสเคส ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History มีจำนวนรายการ 2 รายการ

ตารางที่ 3.10 แสดงค่าถ่วงน้ำหนักของยูสเคส (Ribu, 2001)

ระดับความซับซ้อนของยูสเคส	จำนวนรายการ	ค่าน้ำหนัก
ง่าย	≤ 3	5
ปานกลาง	4 - 7	10
ซับซ้อน	≥ 8	15

จากโมบายแอพพลิเคชั่นตัวอย่าง ยูสเคส ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History มีความซับซ้อนระดับ “ง่าย”

5. การพิจารณาความซับซ้อนของผู้กระทำการจะพิจารณาตามตารางแสดงค่าถ่วงน้ำหนักของผู้กระทำการ

ตารางที่ 3.11 แสดงค่าถ่วงน้ำหนักของผู้กระทำการ(Ribu, 2001)

ระดับผู้กระทำการ	รายละเอียด	ค่าน้ำหนัก
ง่าย	ระบบภายนอกที่จะต้องติดต่อกับระบบผ่านทาง API	1
ปานกลาง	ระบบภายนอกที่จะต้องติดต่อกับระบบผ่านทางโปรโตคอล (e.g. TCP/IP, FTP, HTTP, database) ผู้ใช้งานที่ใช้งานระบบผ่าน text-based user interface	2
ซับซ้อน	ผู้ใช้งานที่ใช้งานระบบผ่านทาง GUI	3

จากโมบายแอพพลิเคชั่นตัวอย่าง ผู้ใช้งาน เป็นผู้กระทำการที่มีความซับซ้อนระดับ “ง่าย”

6. กรอกจำนวนของผู้กระทำการและยูสเคสและค่าปัจจัยความซับซ้อนเชิงเทคนิค T1-T13 และปัจจัยความซับซ้อนทางสภาพแวดล้อม E1 – E8 ลงในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

เอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

Application Name: แอพพลิเคชั่นตัวอย่าง

Unadjusted Actor Weighting Table

Actor Type	Description	Weighting Factor	Number	Result
Simple	External System with well-defined API	1		
Average	External System using a protocol-based interface, e.g., HTTP, TCT/IP, or a database	2		
Complex	Human	3	1	3
			Unadjusted Actor Weight Total (UAW)	3

Unadjusted Use Case Weighting Table

Use Case Type	Description	Weighting Factor	Number	Result
Simple	1 – 3 transactions	5	2	10
Average	4 – 7 transactions	10		
Complex	> 7 transactions	15		
			Unadjusted Use Case Weight Total (UUCW)	

Unadjusted Use Case Points (UUCP) = UAW + UU = _13_

รูปที่ 3.23 แสดงการกรอกข้อมูลในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

Technical Complexity Factors

Factor Number	Description	Weight	Assigned Value	Weighted Value
T1	Distributed System	2	0	0
T2	Response adjectives	1	1	1
T3	End-user efficiency	1	1	1
T4	Complex processing	1	2	2
T5	Reusable code	1	0	0
T6	Easy to install	0.5	0	0
T7	Easy to use	0.5	0	0
T8	Portable	2	0	0
T9	Easy to change	1	1	1
T10	Concurrent	1	0	0
T11	Security features	1	0	0
T12	Access for third-parties	1	0	0
T13	Special training required	1	0	0
			Technical Factor Value (TFactor)	5

Technical Complexity Factor (TCF) = 0.6 + (0.01 * TFactor) = _0.65_

รูปที่ 3.24 แสดงการกรอกข้อมูลในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์(ต่อ)

Environmental Factors

Factor Number	Description	Weight	Assigned Value	Weighted Value
E1	Familiar with development method used	1.5	3	4.5
E2	Application experience	0.5	3	1.5
E3	Object-Oriented experience	1	3	3
E4	Lead analyst capability	0.5	2	1
E5	Motivation	1	3	3
E6	Stable Requirements	2	5	10
E7	Part-time workers	-1	0	0
E8	Difficulty of programming language	-1	1	-1
			Environmental Factor Value (EFactor)	22

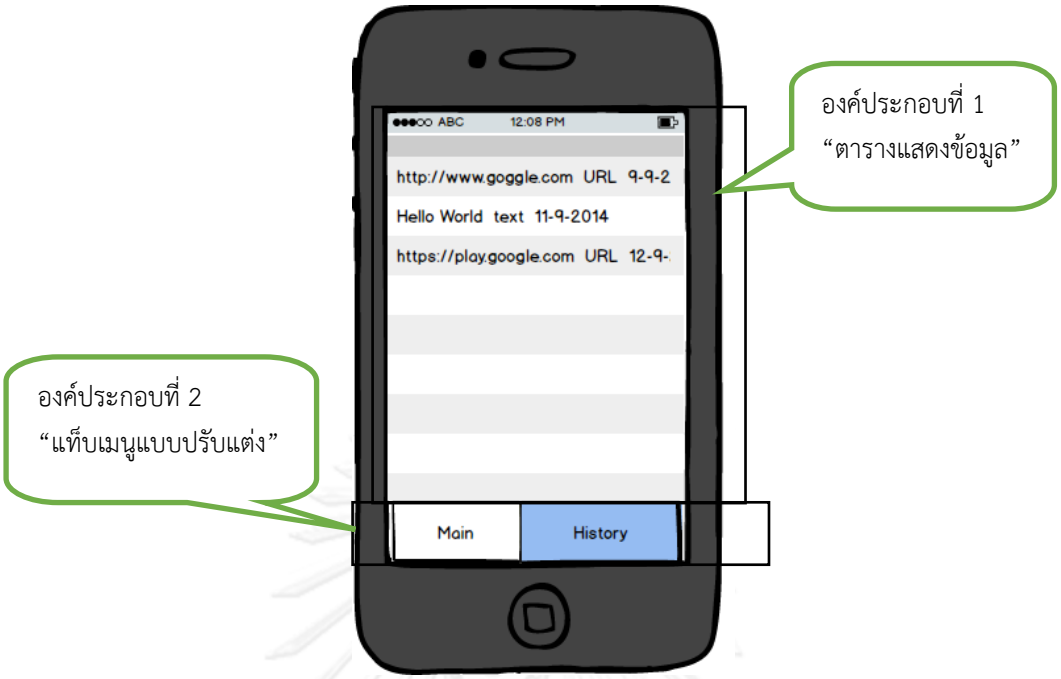
Environmental Factor (EF) = 1.4 + (-0.03 * EFactor) = _0.74_

รูปที่ 3.25 แสดงการกรอกข้อมูลในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์(ต่อ)

ขั้นตอนที่ 4.4 การประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

การประมาณขนาดโดยวิธีนี้จะใช้ข้อมูลจากเอกสารแบบจำลองหน้าจอมาพิจารณาองค์ประกอบของแต่ละหน้าจอว่ามีความซับซ้อนระดับ ง่าย ปานกลาง หรือยาก ตามคำแนะนำของ Nikunj Skhrelia ทั้งนี้พบว่าจะต้องทำการพิจารณาประกอบกับแบบฟอร์มการเก็บข้อมูลส่วนที่ 2 ด้วย เพื่อให้ทราบการขั้นตอนการทำงานของหน้าจอนั้น

1. พิจารณาองค์ประกอบของหน้าจอจากเอกสารการออกแบบหน้าจอและพิจารณาการขั้นตอนการทำงานบนหน้าจอและแหล่งข้อมูลของหน้าจอจากแบบฟอร์มการเก็บข้อมูลส่วนที่ 2 เพื่อให้เข้าใจฟังก์ชั่นการทำงานบนหน้าจอ



รูปที่ 3.26 แสดงการนับองค์ประกอบหน้าจอของโมบายแอพพลิเคชั่นตัวอย่าง

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ History	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

พิจารณาขั้นตอนการทำงานบน

รูปที่ 3.27 แสดงการพิจารณาขั้นตอนการทำงานบนหน้าจอ History ของโมบายแอพพลิเคชั่นตัวอย่าง

จากโมบายแอพพลิเคชั่นตัวอย่าง หน้าจอ History มี 2 องค์ประกอบได้แก่ แท็บเมนูและตารางแสดงข้อมูลจากขั้นตอนการทำงานทำให้ทราบว่าเป็นหน้าจอสำหรับแสดงข้อมูลจากฐานข้อมูลภายใน

CHULALONGKORN UNIVERSITY

2. เทียบกับเกณฑ์การพิจารณาความซ้อนขององค์ประกอบหน้าจอตามคำแนะนำของ Nikunj Sakhelia ซึ่งแสดงไว้เป็นตารางดังตาราง 3.12 ว่าหน้าจอนั้นมีองค์ประกอบที่มีความซับซ้อนระดับใดบ้าง

ตารางที่ 3.12 แสดงการแบ่งฟังก์ชั่นการทำงานและองค์ประกอบบนหน้าจอตามวิธีของ Nikunj Sakhrelia

ระดับความซับซ้อน	รายละเอียดฟังก์ชั่นการทำงานและ องค์ประกอบบนหน้าจอของโมบายแอพพลิเคชั่น	ค่าน้ำหนัก (คน - ชั่วโมง)
ง่าย	การแสดงข้อมูลจากแหล่งข้อมูลภายใน หน้าจอแสดงผลพร้อมปุ่มหรือเมนูอย่างง่าย การใช้งานเอพีไอที่ไม่ต้องปรับแต่ง เข้าสู่ระบบ ลืมรหัสผ่าน การแสดงข้อมูลจากเว็บไซต์โดยใช้ Rssfeed การแสดงข้อมูลทวิตเตอร์(Twitter)ผ่าน Twitter feed การแสดงข้อมูลวิดีโอจากเว็บไซต์ยูทูป(Youtube) ผ่าน Youtube video feed แท็บเมนูที่มีการปรับแต่ง	4
ปานกลาง	มีการเรียกใช้เซอร์เวอร์เอพีไอ(Server API)	8
ซับซ้อน	ตารางแสดงข้อมูลแบบปรับแต่ง การใช้งานกล้อง การใช้งานเซนเซอร์ต่างๆของอุปกรณ์ บันทึกเสียง หรือวิดีโอ การใช้งานเอพีไอที่ต้องปรับแต่ง	16

จากโมบายแอพพลิเคชั่นตัวอย่าง หน้าจอ History มี 2 องค์ประกอบ คือ

- การแสดงตารางข้อมูลจากแหล่งข้อมูลภายใน ซึ่งเป็นองค์ประกอบที่มีความซับซ้อนระดับ “ต่ำ”

- แท็บเมนูที่มีการปรับแต่ง ที่มีระดับความซับซ้อน “ต่ำ” ดังนั้นหน้าจอนี้จึงมีองค์ประกอบที่มีความซับซ้อนระดับต่ำจำนวน 2 องค์ประกอบ

3. กรอกจำนวนของหน้าจอทั้งหมดลงในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

เอกสารการประมาณขนาดซอฟต์แวร์ด้วยวิธีของ Nikunj Sakhrelia

Application Name : แอพพลิเคชั่นตัวอย่าง

ชื่อหน้าจอ	ระดับความซับซ้อน			รวม
	ต่ำ (x4)	ปานกลาง(8)	สูง(16)	
หน้าจอหลัก	2			8
หน้าจอกล้อง				
หน้าจอผลการสแกน				
หน้าจอ History				

Effort: _____

รูปที่ 3.28 แสดงการกรอกข้อมูลในเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

ตัวอย่างเพิ่มเติมของการประมาณขนาดแอพพลิเคชั่นตัวอย่างอยู่ในเอกสารภาคผนวก ข

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ขั้นตอนที่ 5 ประมาณค่าความพยายามในการพัฒนา (Development Effort)

เมื่อทำการประมาณขนาดโมบายแอพพลิเคชั่นด้วย 4 วิธีข้างต้นแล้ว ก็จะต้องนำขนาดของโมบายแอพพลิเคชั่นที่ได้มาทำการประมาณค่าความพยายาม แต่เนื่องจากวิธีการของ Nikunj Sakhrelia ได้ผลลัพธ์การคำนวณออกมาเป็นค่าความพยายามแล้วในขั้นตอน 4.4 ในขั้นตอนนี้จึงแสดงการประมาณค่าความพยายามของวิธีฟังก์ชั่นพอยต์ วิธีอ็อบเจกต์พอยต์ และวิธียูสเคสพอยต์

ขั้นตอนที่ 5.1 ประมาณค่าความพยายามในการพัฒนา (Development Effort) ของวิธีฟังก์ชั่นพอยต์

การประมาณค่าความพยายามจากวิธีฟังก์ชันพอยต์นั้น มีการนำเสนอวิธีการคำนวณที่แตกต่างกันในหลายงานวิจัย ผู้วิจัยจะเลือกวิธีการคำนวณที่มีให้ผลการคำนวณที่ใกล้เคียงค่าพยายามจริงที่สุด โดยเลือกจากวิธีการจากงานวิจัยในอดีตซึ่งมีรายละเอียดการคำนวณดังนี้

$$\text{Effort (man - hour)} = FP \times 2.2$$

จากการเปรียบเทียบความแม่นยำในการประมาณค่าความพยายามของซอฟต์แวร์เชิงวัตถุ (ชมพูนุช 2556) ได้พบว่าสมการแปลงค่าฟังก์ชั่นพอยต์ที่ให้ค่าใกล้เคียงคือสมการของ Jeffery และคณะ ในปี ค.ศ.2001 ซึ่งได้ทำการเปรียบเทียบเวลาในการพัฒนาซอฟต์แวร์โดยเปรียบเทียบจากโครงการซอฟต์แวร์ที่พัฒนาโดยบริษัทเดียวและโครงการที่พัฒนาด้วยหลายบริษัท พบว่าโครงการซอฟต์แวร์ที่พัฒนาโดยบริษัทเดียวใช้เวลาเฉลี่ยอยู่ที่ 2.2 ชั่วโมง ต่อฟังก์ชั่นพอยต์ ขณะที่โครงการซอฟต์แวร์ที่พัฒนาโดยหลายบริษัทใช้เวลาเฉลี่ยอยู่ที่ 9.5 ชั่วโมง ต่อฟังก์ชั่นพอยต์

ขั้นตอนที่ 5.2 ประมาณค่าความพยายามในการพัฒนา (Development Effort) ของวิธีอ็อบเจกต์พอยต์

การประมาณค่าความพยายามจากวิธีอ็อบเจกต์พอยต์นั้นจะนำค่าความสามารถในการพัฒนาซอฟต์แวร์ (Productivity Rate) ของนักพัฒนามาคิดคำนวณด้วย โดยจะพิจารณาค่าความสามารถในการพัฒนาซอฟต์แวร์นั้นจะพิจารณาจากฟีเจอร์ที่เกี่ยวข้อง 2 ประการ คือ ประสบการณ์และความสามารถของนักพัฒนา (Developers' Experience and Capability) ความสมบูรณ์และความสามารถของ ICASE (Integrated Computer-Aided Software Engineering: การพัฒนาซอฟต์แวร์บนเครื่องมือที่มีความสามารถในการช่วยวิเคราะห์และออกแบบระบบ) เนื่องจากในการทบทวนวรรณกรรมของวิธีอ็อบเจกต์พอยต์นั้นไม่ได้เสนอพบเกณฑ์ในการวัดค่าทั้ง 2 ผู้วิจัยจึงทำการทบทวนวรรณกรรมเพิ่มเติมพบว่าในการศึกษาของ Sudhakara

และคณะ (Sudhakara, Farooqb, and Patnaikc,2012) ได้ทำการรวบรวมวรรณกรรมเกี่ยวกับปัจจัยที่กระทบต่อความสามารถในการพัฒนาซอฟต์แวร์ ที่เน้นไปที่ปัจจัยเกี่ยวกับตัวนักพัฒนาและสภาพแวดล้อมของทีม ซึ่งมีความสอดคล้องกับปัจจัย E1-E8 จากเอกสารเก็บข้อมูลโมบายแอพพลิเคชั่นส่วนที่ 3 ที่ใช้ในการพิจารณาค่าความสามารถในการพัฒนาซอฟต์แวร์ของวิธียูสเคสพอยต์ ดังนั้นผู้วิจัยจึงนำค่าปัจจัย E1-E8 ที่ได้คำนวณค่าน้ำหนักตามตาราง 2.28 แล้ว ซึ่งมีค่าที่เป็นไปได้สูงที่สุด คือ 32.5 และต่ำที่สุดคือ -10 มาแบ่งเป็น 5 ช่วงเพื่อใช้ในการพิจารณาค่าความสามารถในการพัฒนาซอฟต์แวร์ของวิธีอ็อบเจกต์พอยต์ด้วย โดยใช้เกณฑ์พิจาณาตามตาราง 3.3

ตารางที่ 3.13 แสดงค่าความสามารถในการพัฒนาซอฟต์แวร์

ประสบการณ์และความสามารถของนักพัฒนาความสมบูรณ์และความสามารถของ ICASE	**ต่ำมาก**	**ต่ำ**	**ปานกลาง**	**สูง**	**สูงมาก**
ผลรวมค่าปัจจัยสภาพแวดล้อมในการพัฒนา(E1-E8)	≥-10	≥-1.5	≥7	≥15.5	≥24
ค่าความสามารถในการพัฒนาซอฟต์แวร์ (PROD)	4	7	13	25	50

หลังจากได้ค่าความสามารถในการพัฒนาซอฟต์แวร์จะทำการประมาณค่าความพยายามจากสมการดังนี้

$$\text{Effort} = \frac{\mathbf{NOP}}{\mathbf{PROD}}$$

เนื่องจากค่าความพยายามที่ได้จากสมการนี้มีหน่วยเป็น คน - เดือน (Man - Month) ดังนั้นจึงต้องทำการแปลงเป็น คน - ชั่วโมง (Man - Hours) เพื่อสามารถเปรียบเทียบกับค่าความพยายามที่ได้จากวิธีฟังก์ชั่นพอยต์ ด้วยการนำไปคูณกับ 20 วันทำงานต่อเดือนโดยประมาณ และ 8 ชั่วโมงทำงานต่อวัน ดังสมการ

Effort (Man - Hours) = Effort (Man –Month) * (20 * 8)

ขั้นตอนที่ 5.3 ประมาณค่าความพยายามในการพัฒนา (Development Effort) ของวิธียูสเคสพอยต์

การพิจาณาค่าความพยายามของวิธียูสเคสพอยต์นั้นจะขึ้นอยู่กับค่าปัจจัยการผลิต (Productivity Factor) ซึ่งเป็นค่าเฉลี่ยของระยะเวลาในการพัฒนาต่อ 1 ยูสเคสพอยต์ของทีมพัฒนา โดยจะได้สูตรการประมาณค่าความพยายามของวิธียูสเคสพอยต์ ดังนี้

Effort = UCP x PF

โดย Karner ได้นำเสนอค่า ปัจจัยการผลิตของวิธียูสเคสพอยต์ โดยให้เท่ากับ 20 คน-ชั่วโมง สำหรับ 1 ยูสเคสพอยต์ หลังจากนั้นการศึกษาของ Schneider และ Winters (Ribu, 2001) ได้เสนอให้มีการพิจารณาค่าปัจจัยการผลิตจากค่าปัจจัยความซับซ้อนทางสภาพแวดล้อม (Environment Complexity Factor) ตามตารางที่ 2.34 เพื่อที่จะได้ค่าที่เหมาะสมกับแต่ละโครงการ พิจารณาค่า X และ Y โดยที่ X เท่ากับผลรวมค่าของปัจจัย E1 – E6 ที่มีค่าต่ำกว่า 3 และให้ Y เท่ากับเท่ากับผลรวมค่าของปัจจัย E7 – E8 ที่มากกว่า 3 ซึ่งเงื่อนในการพิจารณาเป็นดังนี้

- หากผลรวมของ X และ Y มีค่า น้อยกว่าหรือเท่ากับ 2 ให้ใช้ค่า 20 คน-ชั่วโมง
- หากผลรวมของ X และ Y มีค่า มากกว่า 2 ให้ใช้ค่า 28 คน-ชั่วโมง

ขั้นตอนที่ 6 เปรียบเทียบค่าความพยายามที่คำนวณได้

หลังจากที่ได้ ค่าความพยายามจากวิธีการประมาณขนาดทั้ง 4 วิธี คือ (1) วิธีฟังก์ชั่นพอยต์ (2) วิธีอ็อบเจกต์พอยต์ (3)วิธียูสเคสพอยต์ และ (4) วิธีของ Nikunj Sakhrelia การเปรียบเทียบความค่าความพยายามทีคำนวณได้จากทั้ง 4 วิธี

3.2 ตัวแปรที่ทำการศึกษา

งานวิจัยนี้มีวัตถุประสงค์เพื่อเปรียบเทียบวิธีการประมาณค่าความพยายามของโมบายแอพพลิเคชั่น ดังนั้นตัวแปรต้นของการวิจัยครั้งนี้ คือ วิธีการประมาณการขนาดซอฟต์แวร์ทั้ง 4 วิธี คือ

1. วิธีฟังก์ชั่นพอยต์
2. วิธีอ็อบเจกต์พอยต์
3. วิธียูสเคสพอยต์
4. วิธีของ Nikunj Sakhrelia

ประชากร

ประชากรของงานวิจัยนี้คือ โมบายแอพพลิเคชั่นที่ได้รับการพัฒนาเสร็จสิ้นแล้ว

กลุ่มตัวอย่าง

กลุ่มตัวอย่างในการวิจัยครั้งนี้ คือ โมบายแอพพลิเคชั่นที่ได้รับการพัฒนาโดยนักพัฒนาในประเทศไทยเนื่องจากมีกระบวนการพัฒนาและองค์ประกอบไม่แตกต่างจากประชากรที่จะศึกษา ทำให้มีคุณสมบัติที่สามารถเป็นตัวแทนของประชากรได้ ทั้งนี้ยังเป็นการสะดวกแก่ผู้วิจัยในการเก็บข้อมูล

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ขนาดตัวอย่าง

เนื่องจากผู้วิจัยไม่สามารถทราบจำนวนประชากรที่แน่นอนได้ แต่ทราบว่ามีจำนวนมาก และไม่ทราบสัดส่วนของจำนวนประชากร และเนื่องจากข้อจำกัดในการเก็บข้อมูลซึ่งเป็นข้อมูลที่ค่อนข้างเป็นความลับของผู้พัฒนา ดังนั้นผู้วิจัยจึงกำหนดจำนวนหน่วยทดลองที่เป็นไปได้สำหรับงานวิจัยนี้ คือ 20 หน่วย

การเลือกตัวอย่าง

เทคนิคในการการสุ่มหน่วยตัวอย่างที่ดีสุด คือการเลือกตัวอย่างที่ใช้ความน่าจะเป็น ซึ่งเลือกหน่วยตัวอย่างที่เป็นตัวแทนของประชากรโดยยึดหลักว่าทุกหน่วยของประชากรมีโอกาสที่จะได้รับการถูกเลือกมาเป็นหน่วยตัวอย่างเท่าๆ กัน แต่ด้วยเงื่อนไขของการเลือกตัวอย่างตามแบบใช้ความน่าจะเป็นนั้น จะต้องประกอบด้วย (1) ต้องทราบขนาดประชากรว่ามีกี่หน่วย และ (2) ต้องมีกรอบตัวอย่าง (Sampling Frame) กล่าวคือ รายชื่อของแต่ละหน่วยในประชากรพร้อมรายละเอียด (กัลยา วานิชย์บัญชา, 2542) และในงานวิจัยนี้ผู้วิจัยไม่สามารถทราบขนาดของประชากรและรายชื่อของแต่ละหน่วยประชากรทั้งหมดได้ อีกทั้งข้อมูลเหล่านี้เป็นข้อมูลที่เข้าถึงได้ยากเพราะบริษัทที่พัฒนามักจะไม่เปิดเผยข้อมูลแก่บุคคลภายนอก ผู้วิจัยจึงต้องใช้การเลือกตัวอย่างแบบไม่ใช้ความน่าจะเป็น (Non-probability Sampling) โดยเป็นการสุ่มตัวอย่างแบบเจาะจง (Purposive sampling)

3.3 สมมติฐานการวิจัย

ตามวัตถุประสงค์ของงานวิจัยนี้ คือ เปรียบเทียบวิธีการประมาณค่าความพยายามของโมบายแอพพลิเคชั่น และเนื่องจากค่าความพยายามขึ้นกับการประมาณการขนาดซอฟต์แวร์ ผู้วิจัยจึงเปรียบเทียบการประมาณค่าความพยายาม จากวิธีการประมาณการขนาดซอฟต์แวร์ 4 วิธีได้แก่ (1) วิธีฟังก์ชั่นพอยต์ (2) วิธีอ็อบเจกต์พอยต์ (3) วิธียูสเคสพอยต์ และ (4) วิธีของ Nikunj Sakhrelia การประมาณค่าความพยายามของทั้ง 4 วิธี ได้มากจากการประมาณขนาดด้วยวิธีการทางทฤษฎี 3 วิธี ได้แก่ วิธีฟังก์ชั่นพอยต์ (2) วิธีอ็อบเจกต์พอยต์ (3) วิธียูสเคสพอยต์ และวิธีที่ไม่อิงกับทฤษฎีอีก 1 วิธี ได้แก่ วิธีของ Nikunj Sakhrelia ซึ่งมีวิธีการคำควณและใช้ข้อมูลในการคำนวณที่แตกต่างกัน ผู้วิจัยจึงคาดว่า วิธีการประมาณขนาดซอฟต์แวร์ที่ต่างกันจะให้ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นต่างกัน ซึ่งผู้วิจัยได้ตั้งสมมติฐานการวิจัยไว้ดังนี้

เปรียบเทียบการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบาแอพพลิเคชั่นด้วยวิธีการประมาณขนาดที่แตกต่างกัน

กำหนดให้

μ_1 คือ ค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ในหน่วย คน – ชั่วโมง

μ_2 คือ ค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ในหน่วย คน – ชั่วโมง

μ_3 คือ ค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ในหน่วย คน – ชั่วโมง

μ_4 คือ ค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น ด้วยวิธีของ Nikunj Sakhrelia ในหน่วย คน – ชั่วโมง

H_0: $\mu_1 = \mu_2 = \mu_3 = \mu_4$

H_1: $\mu_i \neq \mu_j$ เมื่อ $i \neq j$

การยอมรับ H_0 หมายถึง วิธีการประมาณขนาดซอฟต์แวร์ที่แตกต่างกันไม่ส่งผลต่อการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่น

การปฏิเสธ H_0 หมายถึง วิธีการประมาณขนาดซอฟต์แวร์ที่แตกต่างกันส่งผลต่อการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่น

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.4 ความถูกต้องและความน่าเชื่อถือของข้อมูล

ผู้วิจัยได้พยายามอย่างที่สุดเพื่อให้ได้ข้อมูลที่ดีเพื่อนำไปสู่ผลการวิจัยที่ถูกต้อง (Valid) และเชื่อถือได้ (Reliable) ซึ่งจำเป็นต้องทำการควบคุมปัจจัยต่างๆที่เกี่ยวข้อง ในการเลือกและเก็บรวบรวมข้อมูลโครงการพัฒนาโมบายแอพพลิเคชั่น ซึ่งมีปัจจัยที่ต้องควบคุมดังนี้

1. การเก็บข้อมูลที่ใช้ในการวิจัย จะใช้ข้อมูลจากโครงการพัฒนาโมบายแอพพลิเคชั่นซึ่งเกิดขึ้นจริง และทำการเก็บข้อมูลกับผู้จัดการโครงการหรือนักพัฒนาโดยตรง ถึงแม้ผู้วิจัยจะ

ไม่สามารถเข้าถึงเอกสารความต้องการของโครงการได้ด้วยนโยบายการรักษาความลับของบริษัทพัฒนาซอฟต์แวร์ แต่ได้รับความอนุเคราะห์จากนักพัฒนาโมบายแอพพลิเคชั่นในการสละเวลาให้ข้อมูลสาธิตและอธิบายวิธีการทำงานของโมบายแอพพลิเคชั่นให้แก่ผู้วิจัย ทำให้ได้ข้อมูลรายละเอียดฟังก์ชั่นการทำงานของโมบายแอพพลิเคชั่นที่มีรายละเอียดการทำงานอย่างถูกต้องและครบถ้วน

2. วิธีการประมาณขนาดด้วยวิธีฟังก์ชั่นพอยต์ และวิธีอ็อบเจกต์พอยต์ผู้วิจัยได้ปฏิบัติตามขั้นตอนวิธีการที่เผยแพร่ในเอกสารคู่มือของไอเอฟพียูจี (IFPUG) และเอกสารคู่มือคำอธิบายโมเดลโคโคโม่ 2 ของมหาวิทยาลัยเซาท์เทิร์นแคลิฟอร์เนีย ตามลำลำดับ ซึ่งได้มีการอธิบายรายละเอียดขั้นตอนต่างๆไว้อย่างชัดเจนครบถ้วน ดังนั้นขนาดของซอฟต์แวร์ที่วัดได้จึงถูกต้องและตรงมาตรฐาน

3.5 กรอบการวิเคราะห์ข้อมูล

การวิจัยครั้งนี้ศึกษาตัวแปรเพียงตัวแปรเดียว คือ ค่าเฉลี่ยของการประมาณค่าความพยายามจากหน่วยทดลอง ด้วยวิธีการประมาณขนาด 4 วิธีการ คือ (1) วิธีฟังก์ชั่นพอยต์ (2) วิธีอ็อบเจกต์พอยต์ (3) วิธียูสเคสพอยต์ และ (4) วิธีของ Nikunj Sakhrelia โดยมีสมมติฐานว่า วิธีการประมาณขนาดซอฟต์แวร์ที่แตกต่างกันไม่ส่งผลต่อการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่น ผู้วิจัยจึงเลือกใช้การวิเคราะห์ความแปรปรวน (Analysis of Variance: ANOVA) ด้วยการวิเคราะห์ความแปรปรวนแบบทางเดียว (One-Way ANOVA) ในการวิเคราะห์ความแตกต่างของค่าเฉลี่ยระหว่างกลุ่มข้อมูลทั้ง 4 กลุ่ม

ทั้งนี้ในการวิเคราะห์ความแปรปรวนจะต้องทำการทดสอบการแจกแจงของข้อมูลว่ามีการแจกแจงปกติหรือไม่ หากมีการแจกแจงแบบปกติ ผู้วิจัยจะเลือกใช้สถิติพาราเมตริก ในการทดสอบสมมติฐาน แต่หากข้อมูลไม่มีการแจกแจงแบบปกติ จะใช้สถิตินอนพาราเมตริก ในการทดสอบแทน

บทที่ 4
ผลการวิเคราะห์ข้อมูล

บทนี้จะแสดงผลการวิเคราะห์ข้อมูลตามแนวทางการตอบวัตถุประสงค์ของการวิจัย โดยประกอบไปด้วย ผลการเก็บรวบรวมข้อมูล เอกสารที่เก็บจากแหล่งข้อมูล วิธีการคำนวณขนาดและค่าความพยายามการทดสอบสมมติฐาน ความแม่นยำของวิธีการประมาณค่าความพยายามทั้ง 4 วิธี การเปรียบเทียบความแม่นยำของวิธีการประมาณค่าความพยายามทั้ง 4 วิธี สรุปผลการวิเคราะห์ข้อมูล ซึ่งในการคำนวณและสรุปผลทางสถิติโดยใช้โปรแกรม SPSS เป็นเครื่องมือในการคำนวณ โดยมีรายละเอียดดังต่อไปนี้

4.1 ผลการเก็บรวบรวมข้อมูล

จากการกำหนดจำนวนชุดข้อมูลสำหรับงานวิจัยนี้ไว้ที่ 20 ชุดข้อมูล ผู้วิจัยได้ดำเนินการเก็บข้อมูลเพื่อใช้ในงานวิจัยให้ได้ตามเป้าหมายโดยใช้เอกสารการเก็บข้อมูลตามที่กล่าวไว้ในบทที่ 3.1 ผู้วิจัยได้ติดต่อไปยังแหล่งข้อมูลต่างๆทั้งบริษัทพัฒนาโมบายแอพพลิเคชั่นและนักพัฒนาโมบายแอพพลิเคชั่นอิสระที่รับพัฒนาโมบายแอพพลิเคชั่นให้กับบริษัทต่างๆ หรือที่พัฒนาวางขายในร้านขายโมบายแอพพลิเคชั่นออนไลน์ โดยได้ทำการสัมภาษณ์และนัดพบเพื่อทำการเก็บข้อมูล แต่เนื่องด้วยข้อมูลที่ต้องการจะต้องใช้การกรอกข้อมูลอย่างละเอียดและมีปริมาณค่อนข้างมากทำให้หลายแหล่งข้อมูลได้ขาดการติดต่อ ทั้งนี้ทำให้ในงานวิจัยนี้ผู้วิจัยสามารถเก็บข้อมูลได้น้อยกว่าที่ตั้งเป้าหมายไว้โดยได้จำนวนข้อมูลทั้งหมด 17 ชุดข้อมูล ซึ่งมาจากแหล่งข้อมูลต่างๆกัน ได้แก่ บริษัทพัฒนาซอฟต์แวร์ 2 บริษัท และ นักพัฒนาโมบายแอพพลิเคชั่นอิสระ 5 คน ซึ่งโมบายแอพพลิเคชั่นทั้งหมดถูกพัฒนาด้วยภาษาอ็อบเจกทีฟ-ซี (Objective-C) โดยแบ่งจำนวนชุดข้อมูล (โมบายแอพพลิเคชั่น) ตามแหล่งข้อมูลได้ดังนี้

แอพพลิเคชั่นจากบริษัทที่ 1 จำนวน 2 แอพพลิเคชั่น

แอพพลิเคชั่นจากบริษัทที่ 2 จำนวน 3 แอพพลิเคชั่น

แอพพลิเคชั่นจากนักพัฒนาโมบายแอพพลิเคชั่นอิสระที่ 1 จำนวน 6 แอพพลิเคชั่น

แอพพลิเคชั่นจากนักพัฒนาโมบายแอพพลิเคชั่นอิสระที่ 2 จำนวน 2 แอพพลิเคชั่น

แอพพลิเคชั่นจากนักพัฒนาโมบายแอพพลิเคชั่นอิสระที่ 3 จำนวน 2 แอพพลิเคชั่น

แอพพลิเคชั่นจากนักพัฒนาโมบายแอพพลิเคชั่นอิสระที่ 4 จำนวน 1 แอพพลิเคชั่น

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

แอพพลิเคชั่นจากนักพัฒนาโมบายแอพพลิเคชั่นอิสระที่ 5 จำนวน 1 แอพพลิเคชั่น

ตาราง 4.1 แสดงผลการเก็บข้อมูลโมบายแอพพลิเคชั่น

ลำดับ	ชื่อแอพพลิเคชั่น	จำนวนหน้าจอ	จำนวนฟังก์ชั่น	จำนวนไฟล์ข้อมูล
1	แอพพลิเคชั่นคำนวณค่าโดยสารรถไฟฟ้า BTS	5	4	3
2	แอพพลิเคชั่นแสดงหุ้น	5	4	3
3	แอพพลิเคชั่นนำทาง	3	2	2
4	แอพพลิเคชั่นข้อมูลข่าวสาร	6	6	4
5	แอพพลิเคชั่นข้อมูลภาพยนตร์	5	5	4
6	แอพพลิเคชั่นข้อมูลการท่องเที่ยว	9	8	6
7	แอพพลิเคชั่นดิกชันนารี	6	7	2
8	แอพพลิเคชั่นสำรองข้อมูลรายชื่อติดต่อ	6	7	3
9	แอพพลิเคชั่นรายงานผลฟุตบอล	10	11	9
10	แอพพลิเคชั่นดูวิดีโอออนไลน์	10	10	5
11	แอพพลิเคชั่นข้อมูลซีรีย์	6	5	4
12	แอพพลิเคชั่นบันทึกรายรับ-รายจ่าย	11	6	2
13	แอพพลิเคชั่นตรวจสอบสัญญาณไวไฟ(wi-fi)	2	2	1
14	แอพพลิเคชั่นข้อมูลโรงพยาบาล	11	7	4
15	แอพพลิเคชั่นฟังวิทยุออนไลน์	6	6	7
16	แอพพลิเคชั่นดูโฆษณาออนไลน์	2	2	1
17	แอพพลิเคชั่น QR Reader	4	4	1

โมบายแอพพลิเคชั่นที่ใช้ในการวิจัยครั้งนี้มีจำนวนหน้าจอเฉลี่ยอยู่ที่ 6 หน้าจอ 6 ฟังก์ชั่น และ 4 ไฟล์ข้อมูล แอพพลิเคชั่นที่มีจำนวนหน้าจอสูงที่สุดคือ 11 หน้าจอ แอพพลิเคชั่นที่มีจำนวนฟังก์ชั่นสูงที่สุดคือ 11 ฟังก์ชั่น แอพพลิเคชั่นที่มีจำนวนไฟล์ข้อมูลสูงที่สุดคือ 9 ไฟล์

4.2 วิธีการคำนวณขนาดและค่าความพยายาม

หลังจากได้ข้อมูลจากแหล่งข้อมูลผู้วิจัยได้นำข้อมูลทั้งหมดมาทำการคำนวณขนาดของโมบายแอพพลิเคชั่นด้วยวิธีการทั้ง 4 วิธีซึ่งจะได้ค่าจากการคำนวณดังนี้ วิธีฟังก์ชั่นพอยต์จะได้ค่า Total Adjusted Function Point (TAFP) หรือผลรวมค่าฟังก์ชั่นพอยต์ที่ปรับค่าแล้ว วิธีอ็อบเจกต์พอยต์จะได้ค่า New Object Point (NOP) หรือค่าอ็อบเจกต์พอยต์ใหม่ วิธียูสเคสพอยต์จะได้ค่า Use Case Point (UCP) หรือค่ายูสเคสพอยต์ ส่วนวิธีของ Nikunj Sakhrelia จะได้ผลลัพธ์การคำนวณออกมาเป็นค่าความพยายาม ซึ่งผลการคำนวณขนาดโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ วิธีอ็อบเจกต์พอยต์ และวิธียูสเคสพอยต์ แสดงในตาราง 4.2

หลังจากได้ขนาดของโมบายแอพพลิเคชั่นก็จะนำมาทำการประมาณค่าความพยายามตามวิธีการที่ได้อธิบายไว้ในหัวข้อ 3.1 โดยหลังจากทำการเปรียบเทียบพบว่า ค่าความพยายามจากวิธียูสเคสพอยต์ให้ค่าสูงที่สุดและค่าความพยายามจากวิธีของ Nikunj Sakhrelia ให้ค่าน้อยที่สุดจากค่าการประมาณการของทั้ง 4 วิธี ตามที่แสดงไว้ในตาราง 4.3

จากตาราง 4.3 วิธีฟังก์ชั่นพอยต์(FP) นั้นประมาณค่าความพยายามได้ต่ำกว่าค่าความพยายามจริง(Under Estimate) ทั้งสิ้น 14 แอพพลิเคชั่นและประมาณค่าความพยายามสูงกว่าค่าความพยายามจริง(Over Estimate) จำนวน 3 แอพพลิเคชั่น

สำหรับวิธีอ็อบเจกต์พอยต์(OP) นั้นประมาณค่าความพยายามได้ต่ำกว่าค่าความพยายามจริง (Underestimate) ทั้งสิ้น 8 แอพพลิเคชั่นและประมาณค่าความพยายามสูงกว่าค่าความพยายามจริง (Over Estimate) จำนวน 9 แอพพลิเคชั่น

วิธียูสเคสพอยต์(UCP) นั้นประมาณค่าความพยายามได้ต่ำกว่าค่าความพยายามจริง (Underestimate) ทั้งสิ้น 2 แอพพลิเคชั่นและประมาณค่าความพยายามสูงกว่าค่าความพยายามจริง (Over Estimate) จำนวน 15 แอพพลิเคชั่น

ส่วนวิธีการของ Nikunj Sakhrelia(NS) นั้นประมาณค่าความพยายามได้ต่ำกว่าค่าความพยายามจริง(Underestimate) ทั้งสิ้น 12 แอพพลิเคชั่นและประมาณค่าความพยายามสูงกว่าค่าความพยายามจริง(Over Estimate) จำนวน 5 แอพพลิเคชั่น

ในการพิจารณาค่ามัธยฐานและค่าเฉลี่ยพบว่าค่าความพยายามจากวิธีฟังก์ชั่นพอยต์ วิธีอ็อบเจกต์พอยต์ และวิธีของ Nikunj Sakhrelia นั้นมีค่าที่ต่ำกว่าค่าความพยายามที่ใช้จริงส่วนวิธียูสเคสพอยต์นั้นให้ค่าความพยายามที่สูงกว่าค่าความพยายามที่ใช้จริง

ตาราง 4.2 แสดงผลการคำนวณขนาดโมบายแอพพลิเคชั่น

ชื่อแอพพลิเคชั่น	ฟังก์ชั่นพอยต์	อ็อบเจกต์พอยต์	ยูสเคสพอยต์
	(TAFP)	(NOP)	(UCP)
แอพพลิเคชั่นคำนวณค่าโดยสารรถไฟฟ้า BTS	30.00	45.00	23.00
แอพพลิเคชั่นแสดงหุ้น	29.00	35.00	28.00
แอพพลิเคชั่นนำทาง	14.00	33.00	18.00
แอพพลิเคชั่นข้อมูลข่าวสาร	36.00	26.00	33.00
แอพพลิเคชั่นข้อมูลภาพยนตร์	48.00	35.00	33.00
แอพพลิเคชั่นข้อมูลการท่องเที่ยว	74.00	42.00	48.00
แอพพลิเคชั่นดิกชันนารี	38.00	17.00	33.00
แอพพลิเคชั่นสำรองข้อมูลรายชื่อติดต่อ	37.00	36.00	38.00
แอพพลิเคชั่นรายงานผลฟุตบอล	101.00	72.00	58.00
แอพพลิเคชั่นดูวิดีโอออนไลน์	61.00	60.00	43.00
แอพพลิเคชั่นข้อมูลซีรีย์	32.00	36.00	28.00
แอพพลิเคชั่นบันทึกรายรับ-รายจ่าย	54.00	38.00	43.00
แอพพลิเคชั่นตรวจสอบสัญญาณไวไฟ(wi-fi)	10.00	12.00	13.00
แอพพลิเคชั่นข้อมูลโรงพยาบาล	47.00	50.00	33.00
แอพพลิเคชั่นฟังวิทยุออนไลน์	48.00	56.00	28.00
แอพพลิเคชั่นดูโฆษณาออนไลน์	10.00	32.00	13.00
แอพพลิเคชั่นQR Reader	34.00	24.00	23.00
ค่าต่ำสุด	**10.00**	**12.00**	**13.00**
ค่าสูงสุด	**101.00**	**72.00**	**58.00**
ค่าเฉลี่ย	**41.35**	**38.18**	**31.53**
ค่ามัธยฐาน	**37.00**	**36.00**	**33.00**

ตาราง 4.3 แสดงผลการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น

ชื่อแอพพลิเคชั่น	ค่าความพยายามจากการประมาณการ (คน - ชั่วโมง)				ค่าความพยายามจริง (คน - ชั่วโมง)
	(FP)	(OP)	(UCP)	(NS)	
แอพพลิเคชั่นคำนวณค่าโดยสารรถไฟฟ้า BTS	76.56	144.00	340.95	88.00	192.00
แอพพลิเคชั่นแสดงหุ้น	72.73	224.00	496.05	84.00	272.00
แอพพลิเคชั่นนำทาง	33.88	105.60	234.04	40.00	288.00
แอพพลิเคชั่นข้อมูลข่าวสาร	93.46	166.40	515.82	92.00	116.00
แอพพลิเคชั่นข้อมูลภาพยนตร์	129.89	224.00	566.64	136.00	372.00
แอพพลิเคชั่นข้อมูลการท่องเที่ยว	193.73	268.80	867.22	188.00	504.00
แอพพลิเคชั่นดิกชันนารี	61.86	54.40	282.89	28.00	28.00
แอพพลิเคชั่นสำรองข้อมูลรายชื่อติดต่อ	64.31	230.40	424.61	76.00	176.00
แอพพลิเคชั่นรายงานผลฟุตบอล	222.20	460.80	757.57	248.00	364.00
แอพพลิเคชั่นดูวิดีโอออนไลน์	144.94	192.00	829.26	212.00	120.00
แอพพลิเคชั่นข้อมูลซีรีย์	74.62	115.20	513.91	80.00	22.00
แอพพลิเคชั่นบันทึกรายรับ-รายจ่าย	86.72	243.20	717.95	116.00	224.00
แอพพลิเคชั่นตรวจสอบสัญญาณไวไฟ(wi-fi)	16.94	76.80	155.99	28.00	24.00
แอพพลิเคชั่นข้อมูลโรงพยาบาล	106.50	320.00	442.13	260.00	200.00
แอพพลิเคชั่นฟังวิทยุออนไลน์	105.60	179.20	346.75	116.00	160.00
แอพพลิเคชั่นดูโฆษณาออนไลน์	18.70	102.40	113.57	44.00	120.00
แอพพลิเคชั่นQR Reader	62.08	76.80	216.75	60.00	96.00
ค่าต่ำสุด	**16.94**	**54.40**	**113.57**	**28.00**	**22.00**
ค่าสูงสุด	**222.20**	**460.80**	**867.22**	**260.00**	**504.00**
ค่าเฉลี่ย	**92.04**	**187.29**	**460.12**	**111.53**	**192.82**
ค่ามัธยฐาน	**76.56**	**179.20**	**442.13**	**88.00**	**176.00**

4.3 การทดสอบสมมติฐาน

หลังจากได้ประมาณค่าความพยายามของทั้ง 17 หน่วยตัวอย่าง ด้วยวิธีการทั้ง 4 วิธีครบแล้ว ซึ่งได้ค่าความพยายามตามที่แสดงในตาราง 4.3

ผู้วิจัยจึงทำการตรวจสอบความแตกต่างของค่าความพยายามจากทั้ง 4 วิธี ด้วยการวิเคราะห์ความแปรปรวน ทั้งนี้ในการวิเคราะห์ความแปรปรวนจะต้องทำการทดสอบการแจกแจงของข้อมูลว่ามีการแจกแจงปกติหรือไม่ หากมีการแจกแจงแบบปกติ ผู้วิจัยจะเลือกใช้สถิติพาราเมตริก ในการทดสอบสมมติฐาน แต่หากข้อมูลไม่มีการแจกแจงแบบปกติ จะใช้สถิตินอนพาราเมตริก ในการทดสอบแทน

4.3.1. การทดสอบการแจกแจงปกติของข้อมูล

ค่าความพยายามที่ได้จากวิธีฟังก์ชั่นพอยต์ ใช้สมมติฐานในการทดสอบการแจกแจงของข้อมูล คือ

H_0: ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์มีการแจงแจกปกติ

H_1: ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ไม่มีการแจงแจกปกติ

กำหนดระดับนัยสำคัญเป็น 0.05 (α = 0.05)

โดยงานวิจัยนี้มีจำนวนหน่วยตัวอย่าง 17 หน่วย ซึ่งน้อยกว่า 50 หน่วย ผู้วิจัยจึงเลือกใช้การทดสอบ Shapiro-Wilk โดยจะปฏิเสธ H_0 หากค่า Sig. ของการทดสอบน้อยกว่า ระดับนัยสำคัญที่กำหนด ซึ่งคือ 0.05 ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ดังตาราง 4.4

ตารางที่ 4.4 ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์

	ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์
ค่า Sig.	0.156
การแจกแจง	มีการแจกแจงปกติ

ค่าความพยายามที่ได้จากวิธีอ็อบเจกต์พอยต์ ใช้สมมติฐานในการทดสอบการแจกแจงของข้อมูล คือ

H_0: ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์มีการแจงแจกปกติ

H_1: ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ไม่มีการแจงแจกปกติ

กำหนดระดับนัยสำคัญเป็น 0.05 (α = 0.05)

ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ดังตาราง 4.5

ตารางที่ 4.5 ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์

	ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์
ค่า Sig.	0.154
การแจกแจง	มีการแจกแจงปกติ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ค่าความพยายามที่ได้จากวิธียูสเคสพอยต์ ใช้สมมติฐานในการทดสอบการแจกแจงของข้อมูล คือ

H_0: ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์มีการแจงแจกปกติ

H_1: ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ไม่มีการแจงแจกปกติ

กำหนดระดับนัยสำคัญเป็น 0.05 (α = 0.05)

ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ดังตาราง 4.6

ตารางที่ 4.6 ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

	ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์
ค่า Sig.	0.544
การแจกแจง	มีการแจกแจงปกติ

ค่าความพยายามที่ได้จากวิธีของ Nikunj Sakhrelia ใช้สมมติฐานในการทดสอบการแจกแจงของข้อมูล คือ

H_0: ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia มีการแจงแจกปกติ

H_1: ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia ไม่มีการแจงแจกปกติ

กำหนดระดับนัยสำคัญเป็น 0.05 (α = 0.05)

ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia ดังตาราง 4.7

ตารางที่ 4.7 ผลสรุปการทดสอบการแจกแจงปกติของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

	ค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia
ค่า Sig.	0.04 *
การแจกแจง	ไม่มีการแจกแจงปกติ

4.3.2. เปรียบเทียบค่าความพยายามจากวิธีการที่แตกต่างกัน

ผู้วิจัยเปรียบเทียบค่าเฉลี่ยค่าความพยายามของวิธีการทั้ง 4 วิธี ด้วยการวิเคราะห์ความแปรปรวน (Analysis of Variance: ANOVA) และเนื่องจากมีข้อมูลค่าความพยายามของหนึ่งวิธี ไม่มีการแจกแบบปกติ จึงเลือกใช้ วิธีการทดสอบ Kruskal –Wallis ในการทดสอบ โดยมีสมมติฐานการทดสอบ คือ

H_0: วิธีการประมาณขนาดซอฟต์แวร์ที่แตกต่างกันไม่ส่งผลต่อการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่น ($\mu_1 = \mu_2 = \mu_3 = \mu_4$)

H_1: หมายถึง วิธีการประมาณขนาดซอฟต์แวร์ที่แตกต่างกันส่งผลต่อการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่น ($\mu_i \neq \mu_j$ เมื่อ i $\neq$ j)

กำหนดระดับนัยสำคัญเป็น 0.05 (α = 0.05)

μ_1 คือ ค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์

μ_2 คือ ค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์

μ_3 คือ ค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

μ_4 คือ ค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

โดยจะปฏิเสธ H_0 ถ้าค่า Sig. (Significance) จากการทดสอบมีค่าน้อยกว่าระดับนัยสำคัญที่กำหนด ในงานวิจัยนี้กำหนดระดับนัยสำคัญที่ 0.05

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตาราง 4.8 ผลการทดสอบ Kruskal –Wallis ของค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นทั้ง 4 วิธี จากข้อมูลทั้งหมด

Methods	N	Mean	Mean Rank	Chi-Square	df	Sig.
FP	17	92.04	19.97	35.067	3	.000*
OP	17	187.29	37.44			
UCP	17	460.12	56.41			
NS	17	111.53	24.18			

จากผลการทดสอบค่า Sig. (Significance) มีค่าเท่ากับ 0.000 ซึ่งน้อยกว่าค่านัยสำคัญที่งานวิจัยนี้กำหนด (0.05) ทำให้สามารถปฏิเสธ H_0 นั่นหมายความว่า มีค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นอย่างน้อยหนึ่งคู่ไม่เท่ากัน กล่าวคือ วิธีการประมาณขนาดซอฟต์แวร์ที่แตกต่างกันส่งผลต่อการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่น

เนื่องจากพบว่า มีค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นอย่างน้อยหนึ่งคู่ไม่เท่ากัน จึงทำการทดสอบเปรียบเทียบค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นรายคู่ด้วยการทดสอบของแมนและวิทนีย์(The Mann-Whitney U Test)สำหรับคู่ที่ข้อมูลไม่มีการแจกแจงปกติและทดสอบด้วยการทดสอบที(T-Test)สำหรับคู่ที่ข้อมูลมีการแจกแจงปกติ

โดยทดสอบเปรียบเทียบรายคู่ปรากฏว่า ไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.05 ระหว่าง 1 คู่ทดสอบ คือ

1) วิธีฟังก์ชั่นพอยต์ และวิธีของ Nikunj Sakhrelia ค่า Sig. 0.502 มากกว่าค่านัยสำคัญ (0.05)

การทดสอบพบความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.05 ระหว่าง 5 คู่ คือ

1) วิธีฟังก์ชั่นพอยต์และวิธีอ็อบเจกต์พอยต์ ค่า Sig. 0.003 น้อยกว่าค่านัยสำคัญ

2) วิธีฟังก์ชั่นพอยต์และวิธียูสเคสพอยต์ ค่า Sig. 0.000 น้อยกว่าค่านัยสำคัญ

3) วิธีอ็อบเจกต์พอยต์ และ วิธียูสเคสพอยต์ ค่า Sig. 0.000 น้อยกว่าค่านัยสำคัญ

4) วิธีอ็อบเจกต์พอยต์และวิธีของ Nikunj Sakhrelia ค่า Sig. 0.024 น้อยกว่าค่านัยสำคัญ

5) วิธียูสเคสพอยต์ และวิธีของ Nikunj Sakhrelia ค่า Sig. 0.000 น้อยกว่าค่านัยสำคัญ

ซึ่งให้ผลการทดสอบตามตาราง 4.9-4.10

ตาราง 4.9 ผลการเปรียบเทียบค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น สำหรับคู่ที่ข้อมูลมีการแจงแจกปกติ

การทดสอบ		จำนวน	f	Sig.	t	Sig.
FP	OP	17	4.236	0.048	-3.348	0.003*
	UCP	17	19.733	0.000	-6.369	.000*
OP	UCP	17	9.485	0.004	-4.436	.000*

ตาราง 4.10 ผลการเปรียบเทียบค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น สำหรับคู่ที่ข้อมูลไม่มีการแจงแจกปกติ

การทดสอบ		จำนวน	Mann-Whitney U Value	Sig.
NS	FP	17	125	0.502
	OP	17	79	0.024*
	UCP	17	15	.000*

4.4 ความแม่นยำของวิธีการประมาณค่าความพยายามทั้ง 4 วิธี

ในการเปรียบเทียบค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นของทั้ง 4 วิธีนั้น ทำให้ได้ผลว่าวิธีการแต่ละวิธีการนั้นให้ค่าความพยายามที่แตกต่างกัน แต่ทั้งนี้ผู้วิจัยไม่สามารถสรุปได้ว่าวิธีการใดที่มีความเหมาะสมในการนำไปใช้จริงในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น ผู้วิจัยจึงทำการทดสอบความแม่นยำของค่าความพยายามที่ได้จากวิธีการทั้งสี่วิธีโดยนำไปเปรียบเทียบกับค่าความพยายามจริงที่ได้จากแหล่งข้อมูลที่เก็บมาโดยใช้หน่วยคน-ชั่วโมง (Man-Hour) โดยในการทดสอบความแม่นยำผู้วิจัยใช้วิธีการเปรียบเทียบค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง (Mean Magnitude of Relative Error: MMRE) และ และอัตราร้อยละของระบบที่ประมาณค่าความพยายามได้ถูกต้อง (Percentage of predictions: PRED) ในการพิจารณาความแม่นยำ

ค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง (Mean Magnitude of Relative Error: MMRE) (Conte et al., 1986) ได้ถูกคิดค้นขึ้นเพื่อแก้ปัญหาค่าความคลาดเคลื่อนที่แปรผันกับขนาดของระบบ Conte และคณะ จึงได้นำเสนอวิธีการหารเฉลี่ยขนาดของระบบจริง เพื่อลดความผิดพลาดในส่วนนี้ซึ่งสามารถหาได้ด้วยสมการ ดังนี้

$$MMRE = \frac{1}{n} x \sum_{i=1}^{n} MRE$$

$$MRE = \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

โดย

y_i คือ ค่าความพยายามที่ใช้ในการพัฒนาระบบจริง

$\hat{y}_i$ คือ ค่าความพยายามที่ได้จากวิธีการประมาณขนาดระบบ

อัตราร้อยละของระบบที่ประมาณค่าความพยายามได้ถูกต้อง (Percentage of predictions: PRED) คือ ร้อยละของระบบทั้งหมดที่มีค่าความคลาดเคลื่อนจากการประมาณค่าความ พยายามได้ต่ำกว่าค่าขีดแบ่ง (Threshold) ซึ่งส่วนใหญ่จะใช้ 0.25 หรือให้ความคลาดเคลื่อนไม่เกิน 25% (Conte et al., 1986) สามารถคำนวณได้จากสมการ

$$PRED\ (q) = \frac{k}{n}$$

q คือ ค่า MRE ที่ตั้งไว้ ค่าขีดแบ่ง (threshold) ในงานวิจัยส่วนใหญ่จะยอมรับที่ค่า 0.25

k คือ จำนวนระบบที่ MRE มีค่าน้อยกว่า หรือเท่ากับ q

n คือ จำนวนระบบทั้งหมด

ผู้วิจัยจึงเลือกการวิเคราะห์ข้อมูล PRED(0.25) ซึ่งหมายถึง วัดอัตราร้อยละของระบบที่ประมาณค่าความพยายามได้คลาดเคลื่อนน้อยกว่า 25%

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.4.1. ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์

จากข้อมูลการประมาณค่าความพยายามของข้อมูลทั้ง 17 ชุดข้อมูลด้วยวิธีฟังก์ชั่นพอยต์ พบว่า ค่าความพยายาม มีค่าตั้งแต่ 16.94 - 222.20 คน-ชั่วโมง ค่าเฉลี่ยและค่ามัธยฐานของค่าความพยายามอยู่ที่ 92.04 และ 76.56 คน-ชั่วโมง ตามลำดับ เมื่อนำมาคำนวณค่าความคาดเคลื่อนโดยเทียบเคียง จะได้ค่าดังตาราง 4.11

จากตารางจะพบว่าค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีฟังก์ชั่นพอยต์(MRE) ในแต่ละชุดข้อมูลมีค่าตั้งแต่ 0.194 – 2.392 และมีค่าเฉลี่ยอยู่ที่ 0.672 นั่นหมายความว่าวิธีฟังก์ชั่นพอยต์ให้ค่าเบี่ยงเบนไปจากข้อมูลจริงร้อยละ 67.2 แสดงให้เห็นว่ามีความคาดเคลื่อนค่อนข้างมาก และมี อัตราร้อยละของระบบที่ประมาณค่าความพยายามได้ถูกต้อง เท่ากับ 0.11765 หมายถึงมีจำนวนร้อยละ 11.765 ของชุดข้อมูลทั้งหมดที่ประมาณค่าความพยายามผิดพลาดไม่เกินร้อยละ 25 (PRED(0.25))

ผู้วิจัยทำการทดสอบความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ในทางสถิติ ทั้งนี้ การประมาณขนาดซอฟต์แวร์ที่ยอมรับได้จะประมาณค่าความพยายามในการพัฒนาซอฟต์แวร์ได้ผิดพลาดไม่เกินร้อยละ 25 หรือมีค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง(MRE) เท่ากับ 0.25 (Conte et al., 1986) โดยทางทำการทดสอบการแจกแจงปกติของข้อมูลค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีฟังก์ชั่นพอยต์ หากมีการแจกแจงแบบปกติ ผู้วิจัยจะเลือกใช้สถิติพาราเมตริก ในการทดสอบสมมติฐาน แต่หากข้อมูลไม่มีการแจกแจงแบบปกติ จะใช้สถิตินอนพาราเมตริก ในการทดสอบแทน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 4.11 แสดงผลการคำนวณค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีฟังก์ชั่นพอยต์

ลำดับ	ชื่อแอพพลิเคชั่น	ค่าความพยายามจากวิธี FP (Man-hour)	ค่าความพยายามจริง (Man-hour)	MRE
1	แอพพลิเคชั่นคำนวณค่าโดยสารรถไฟฟ้า BTS	76.56	192	0.601
2	แอพพลิเคชั่นแสดงหุ้น	72.732	272	0.733
3	แอพพลิเคชั่นนำทาง	33.88	288	0.882
4	แอพพลิเคชั่นข้อมูลข่าวสาร	93.456	116	0.194
5	แอพพลิเคชั่นข้อมูลภาพยนตร์	129.888	372	0.651
6	แอพพลิเคชั่นข้อมูลการท่องเที่ยว	193.732	504	0.616
7	แอพพลิเคชั่นดิกชันนารี	61.864	28	1.209
8	แอพพลิเคชั่นสำรองข้อมูลรายชื่อติดต่อ	64.306	176	0.635
9	แอพพลิเคชั่นรายงานผลฟุตบอล	222.2	364	0.390
10	แอพพลิเคชั่นดูวิดีโอออนไลน์	144.936	120	0.208
11	แอพพลิเคชั่นข้อมูลซีรีย์	74.624	22	2.392
12	แอพพลิเคชั่นบันทึกรายรับ-รายจ่าย	86.724	224	0.613
13	แอพพลิเคชั่นตรวจสอบสัญญาณไวไฟ(wi-fi)	16.94	24	0.294
14	แอพพลิเคชั่นข้อมูลโรงพยาบาล	106.502	200	0.467
15	แอพพลิเคชั่นฟังวิทยุออนไลน์	105.6	160	0.340
16	แอพพลิเคชั่นดูโฆษณาออนไลน์	18.7	120	0.844
17	แอพพลิเคชั่นQR Reader	62.084	96	0.353
ค่าเฉลี่ย (MMRE)				**0.672**

การทดสอบการแจกแจงปกติของข้อมูล

ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีฟังก์ชั่นพอยต์ ใช้สมมติฐานในการทดสอบการแจกแจงของข้อมูล คือ

H_0: ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ (MRE_{FP}) มีการแจงแจกปกติ

H_1: ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ (MRE_{FP}) ไม่มีการแจงแจกปกติ

กำหนดระดับนัยสำคัญเป็น 0.05 (α = 0.05)

โดยงานวิจัยนี้มีจำนวนหน่วยตัวอย่าง 17 หน่วย ซึ่งน้อยกว่า 50 หน่วย ผู้วิจัยจึงเลือกใช้การทดสอบ Shapiro-Wilk โดยจะปฏิเสธ H_0 หากค่า Sig. ของการทดสอบน้อยกว่า ระดับนัยสำคัญที่กำหนด ซึ่งคือ 0.05 ผลสรุปการทดสอบการแจกแจงปกติของค่าเฉลี่ยของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์ดังตาราง 4.12

ตาราง 4.12 แสดงผลการทดสอบการแจกแจงข้อมูลของวิธีฟังก์ชั่นพอยต์

	ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์
ค่า Sig.	0.000*
การแจกแจง	ไม่มีการแจกแจงปกติ

จากการทดสอบพบว่าข้อมูลไม่มีการแจกแจงปกติดังนั้นผู้วิจัยจึงเลือกการทดสอบโดยใช้สถิตินอนพาราเมตริก ด้วยการทดสอบด้วยสถิติวิลคอกซัน (Wilcoxon Signed Rank Test) โดยใช้ค่ากลางของข้อมูลในการทดสอบ กำหนดสมมติฐานดังนี้

H_0: ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์มีความคาดเคลื่อนน้อยกว่าหรือเท่ากับร้อยละ 25

H_1: ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์มีความคาดเคลื่อนมากกว่าร้อยละ 25

โดยจะปฏิเสธ H_0 ถ้าค่า Sig. (Significance) จากการทดสอบมีค่าน้อยกว่าระดับนัยสำคัญที่กำหนด ในงานวิจัยนี้กำหนดระดับนัยสำคัญที่ 0.05

ตาราง 4.13 ผลการทดสอบความแม่นยำในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์

Null Hypothesis	Test	sig.
ค่ากลาง MRE น้อยกว่าหรือเท่ากับ 0.25	One-Sample Wilcoxon Sined Rank Test	0.0005*

ผลการทดสอบพบว่า ค่า Sig.(2-tailed) มีค่า 0.001 แต่เนื่องจากการทดสอบนี้ใช้การทดสอบสมมติฐานแบบทางเดียวทำให้ค่า Sig. ที่ได้ต้องนำมาหารด้วย 2 จึงได้ค่า Sig. เท่ากับ 0.0005 มีค่าน้อยกว่าระดับนัยสำคัญ คือ 0.05 จึงปฏิเสธ H_0 หมายความว่า ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์มีความคาดเคลื่อนมากกว่าร้อยละ 25

จึงสรุปว่าวิธีฟังก์ชั่นพอยต์น่าจะไม่เหมาะสมในการใช้ประมาณการค่าความพยายามของข้อมูลในงานวิจัยชิ้นนี้

4.4.2. ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์

จากข้อมูลการประมาณค่าความพยายามของข้อมูลทั้ง 17 ชุดข้อมูลด้วยวิธีอ็อบเจกต์พอยต์ พบว่า ค่าความพยายาม มีค่าตั้งแต่ 54.40 - 460.80 คน-ชั่วโมง ค่าเฉลี่ยและค่ามัธยฐานของค่าความพยายามอยู่ที่ 187.29 และ 179.20 คน-ชั่วโมง ตามลำดับ เมื่อนำมาคำนวณค่าความคาดเคลื่อนโดยเทียบเคียง จะได้ค่าดังตาราง 4.13

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 4.14 แสดงผลการคำนวณค่า MRE ของวิธีอ็อบเจกต์พอยต์

ลำดับ	ชื่อแอพพลิเคชั่น	ค่าความพยายามจากวิธี OP (Man-hour)	ค่าความพยายามจริง (Man-hour)	MRE
1	แอพพลิเคชั่นคำนวณค่าโดยสารรถไฟฟ้า BTS	144	192	0.250
2	แอพพลิเคชั่นแสดงหุ้น	224	272	0.176
3	แอพพลิเคชั่นนำทาง	105.6	288	0.633
4	แอพพลิเคชั่นข้อมูลข่าวสาร	166.4	116	0.434
5	แอพพลิเคชั่นข้อมูลภาพยนตร์	224	372	0.398
6	แอพพลิเคชั่นข้อมูลการท่องเที่ยว	268.8	504	0.467
7	แอพพลิเคชั่นดิกชันนารี	54.4	28	0.943
8	แอพพลิเคชั่นสำรองข้อมูลรายชื่อติดต่อ	230.4	176	0.309
9	แอพพลิเคชั่นรายงานผลฟุตบอล	460.8	364	0.266
10	แอพพลิเคชั่นดูวิดีโอออนไลน์	192	120	0.600
11	แอพพลิเคชั่นข้อมูลซีรีย์	115.2	22	4.236
12	แอพพลิเคชั่นบันทึกรายรับ-รายจ่าย	243.2	224	0.086
13	แอพพลิเคชั่นตรวจสอบสัญญาณไวไฟ(wi-fi)	76.8	24	2.200
14	แอพพลิเคชั่นข้อมูลโรงพยาบาล	320	200	0.600
15	แอพพลิเคชั่นฟังวิทยุออนไลน์	179.2	160	0.120
16	แอพพลิเคชั่นดูโฆษณาออนไลน์	102.4	120	0.147
17	แอพพลิเคชั่นQR Reader	76.8	96	0.200
ค่าเฉลี่ย (MMRE)				**0.710**

จากตารางจะพบว่าค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีอ็อบเจกต์พอยต์ ในแต่ละชุดข้อมูลมีค่าตั้งแต่ 0.086 - 4.236 และมีค่าเฉลี่ยอยู่ที่ 0.710 นั่นหมายความว่าวิธีอ็อบเจกต์พอยต์ให้ค่าเบี่ยงเบนไปจากข้อมูลจริงร้อยละ 71.0 แสดงให้เห็นว่ามีความคาดเคลื่อนค่อนข้างมาก และมีอัตราร้อยละของระบบที่ประมาณค่าความพยายามได้ถูกต้อง เท่ากับ 0.294 หมายถึงมีจำนวนร้อยละ 29.41 ของชุดข้อมูลทั้งหมดที่ประมาณค่าความพยายามผิดพลาดไม่เกินร้อยละ 25 (PRED(0.25))

ผู้วิจัยทำการทดสอบความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ในทางสถิติ ทั้งนี้ การประมาณขนาดซอฟต์แวร์ที่ยอมรับได้จะประมาณค่าความพยายามในการพัฒนาซอฟต์แวร์ได้ผิดพลาดไม่เกินร้อยละ 25 หรือมีค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง(MRE) เท่ากับ 0.25 (Conte et al., 1986) โดยทางทำการทดสอบการแจกแจงปกติของข้อมูลค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีอ็อบเจกต์พอยต์ หากมีการแจกแจงแบบปกติ ผู้วิจัยจะเลือกใช้สถิติพาราเมตริก ในการทดสอบสมมติฐาน แต่หากข้อมูลไม่มีการแจกแจงแบบปกติ จะใช้สถิตินอนพาราเมตริก ในการทดสอบแทน

การทดสอบการแจกแจงปกติของข้อมูล

ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีอ็อบเจกต์พอยต์ ใช้สมมติฐานในการทดสอบการแจกแจงของข้อมูล คือ

H_0: ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ (MRE_{OP}) มีการแจงแจกปกติ

H_1: ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ (MRE_{OP}) ไม่มีการแจงแจกปกติ

กำหนดระดับนัยสำคัญเป็น 0.05 (α = 0.05)

โดยงานวิจัยนี้มีจำนวนหน่วยตัวอย่าง 17 หน่วย ซึ่งน้อยกว่า 50 หน่วย ผู้วิจัยจึงเลือกใช้การทดสอบ Shapiro-Wilk โดยจะปฏิเสธ H_0 หากค่า Sig. ของการทดสอบน้อยกว่า ระดับนัยสำคัญที่กำหนด ซึ่งคือ 0.05 ผลสรุปการทดสอบการแจกแจงปกติของค่าเฉลี่ยของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์ดังตาราง 4.15

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตาราง 4.15 แสดงผลการทดสอบการแจกแจงข้อมูลของวิธีอ็อบเจกต์พอยต์

	ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์
ค่า Sig.	0.000*
การแจกแจง	ไม่มีการแจกแจงปกติ

จากการทดสอบพบว่าข้อมูลไม่มีการแจกแจงปกติดังนั้นผู้วิจัยจึงเลือกการทดสอบโดยใช้สถิตินอนพาราเมตริก ด้วยการทดสอบด้วยสถิติวิลคอกซัน (Wilcoxon Signed Ranks Test) โดยใช้ค่ากลางของข้อมูลในการทดสอบ กำหนดสมมติฐานดังนี้

H_0: ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์มีความคาดเคลื่อนน้อยกว่าหรือเท่ากับร้อยละ 25

H_1: ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์มีความคาดเคลื่อนมากกว่าร้อยละ 25

โดยจะปฏิเสธ H_0 ถ้าค่า Sig. (Significance) จากการทดสอบมีค่าน้อยกว่าระดับนัยสำคัญที่กำหนด ในงานวิจัยนี้กำหนดระดับนัยสำคัญที่ 0.05

ตาราง 4.16 ผลการทดสอบความแม่นยำในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์

Null Hypothesis	Test	sig.
ค่ากลาง MRE น้อยกว่าหรือเท่ากับ 0.25	One-Sample Wilcoxon Sined Rank Test	0.013*

ผลการทดสอบพบว่า ค่า Sig.(2-tailed) มีค่า 0.026 แต่เนื่องจากการทดสอบนี้ใช้การทดสอบสมมติฐานแบบทางเดียวทำให้ค่า Sig. ที่ได้ต้องนำมาหารด้วย 2 จึงได้ค่า Sig. เท่ากับ 0.013 มีค่าน้อยกว่าระดับนัยสำคัญ คือ 0.05 จึงปฏิเสธ H_0 หมายความว่า ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์มีความคาดเคลื่อนมากกว่าร้อยละ 25

จึงสรุปว่าวิธีอ็อบเจกต์พอยต์น่าจะไม่เหมาะสมในการใช้ประมาณการค่าความพยายามของข้อมูลในงานวิจัยชิ้นนี้

4.4.3. ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

จากข้อมูลการประมาณค่าความพยายามของข้อมูลทั้ง 17 ชุดข้อมูลด้วยวิธียูสเคสพอยต์พบว่าค่าความพยายาม มีค่าตั้งแต่ 113.57 – 867.22 คน-ชั่วโมง ค่าเฉลี่ยและค่ามัธยฐานของค่าความพยายามอยู่ที่ 460.12 และ 442.13 คน-ชั่วโมง ตามลำดับ เมื่อนำมาคำนวณค่าความคาดเคลื่อนโดยเทียบเคียง จะได้ค่าดังตาราง 4.16

จากตาราง 4.16 จะพบว่าค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธียูสเคสพอยต์ ในแต่ละชุดข้อมูลมีค่าตั้งแต่ 0.054 - 22.360 และมีค่าเฉลี่ยอยู่ที่ 3.396 นั่นหมายความว่าวิธียูสเคสพอยต์ให้ค่าเบี่ยงเบนไปจากข้อมูลจริงถึงร้อยละ 339.6 แสดงให้เห็นว่ามีความคาดเคลื่อนมาก และมีอัตราร้อยละของระบบที่ประมาณค่าความพยายามได้ถูกต้อง เท่ากับ 0.1176

หมายถึงมีจำนวนร้อยละ 11.76 ของชุดข้อมูลทั้งหมดที่ประมาณค่าความพยายามผิดพลาดไม่เกินร้อยละ 25 (PRED(0.25))

ผู้วิจัยทำการทดสอบความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ในทางสถิติ ทั้งนี้ การประมาณขนาดซอฟต์แวร์ที่ยอมรับได้จะประมาณค่าความพยายามในการพัฒนาซอฟต์แวร์ได้ผิดพลาดไม่เกินร้อยละ 25 หรือมีค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง(MRE) เท่ากับ 0.25 (Conte et al., 1986) โดยทางทำการทดสอบการแจกแจงปกติของข้อมูลค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธียูสเคสพอยต์ หากมีการแจกแจงแบบปกติ ผู้วิจัยจะเลือกใช้สถิติพาราเมตริก ในการทดสอบสมมติฐาน แต่หากข้อมูลไม่มีการแจกแจงแบบปกติ จะใช้สถิตินอนพาราเมตริก ในการทดสอบแทน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 4.17 แสดงผลการคำนวณค่า MRE ของวิธียูสเคสพอยต์

ลำดับ	ชื่อแอพพลิเคชั่น	ค่าความพยายามจากวิธี UCP (Man-hour)	ค่าความพยายามจริง (Man-hour)	MRE
1	แอพพลิเคชั่นคำนวณค่าโดยสารรถไฟฟ้า BTS	340.952	192	0.776
2	แอพพลิเคชั่นแสดงหุ้น	496.048	272	0.824
3	แอพพลิเคชั่นนำทาง	234.036	288	0.187
4	แอพพลิเคชั่นข้อมูลข่าวสาร	515.823	116	3.447
5	แอพพลิเคชั่นข้อมูลภาพยนตร์	566.643	372	0.523
6	แอพพลิเคชั่นข้อมูลการท่องเที่ยว	867.216	504	0.721
7	แอพพลิเคชั่นดิกชันนารี	282.8925	28	9.103
8	แอพพลิเคชั่นสำรองข้อมูลรายชื่อติดต่อ	424.612	176	1.413
9	แอพพลิเคชั่นรายงานผลฟุตบอล	757.567	364	1.081
10	แอพพลิเคชั่นดูวิดีโอออนไลน์	829.255	120	5.910
11	แอพพลิเคชั่นข้อมูลซีรีย์	513.912	22	22.360
12	แอพพลิเคชั่นบันทึกรายรับ-รายจ่าย	717.9452	224	2.205
13	แอพพลิเคชั่นตรวจสอบสัญญาณไวไฟ(wi-fi)	155.987	24	5.499
14	แอพพลิเคชั่นข้อมูลโรงพยาบาล	442.134	200	1.211
15	แอพพลิเคชั่นฟังวิทยุออนไลน์	346.752	160	1.167
16	แอพพลิเคชั่นดูโฆษณาออนไลน์	113.568	120	0.054
17	แอพพลิเคชั่นQR Reader	216.752	96	1.258
ค่าเฉลี่ย (MMRE)				**3.396**

การทดสอบการแจกแจงปกติของข้อมูล

ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธียูสเคสพอยต์ ใช้สมมติฐานในการทดสอบการแจกแจงของข้อมูล คือ

H_0: ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ (MRE_{UCP}) มีการแจงแจกปกติ

H_1: ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ (MRE_{UCP}) ไม่มีการแจงแจกปกติ

กำหนดระดับนัยสำคัญเป็น 0.05 (α = 0.05)

โดยงานวิจัยนี้มีจำนวนหน่วยตัวอย่าง 17 หน่วย ซึ่งน้อยกว่า 50 หน่วย ผู้วิจัยจึงเลือกใช้การทดสอบ Shapiro-Wilk โดยจะปฏิเสธ H_0 หากค่า Sig. ของการทดสอบน้อยกว่า ระดับนัยสำคัญที่กำหนด ซึ่งคือ 0.05 ผลสรุปการทดสอบการแจกแจงปกติของค่าเฉลี่ยของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์ดังตาราง 4.18

ตาราง 4.18 แสดงผลการทดสอบการแจกแจงข้อมูลของวิธียูสเคสพอยต์

	ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์
ค่า Sig.	0.000*
การแจกแจง	ไม่มีการแจกแจงปกติ

จากการทดสอบพบว่าข้อมูลไม่มีการแจกแจงปกติดังนั้นผู้วิจัยจึงเลือกการทดสอบโดยใช้สถิตินอนพาราเมตริก ด้วยการทดสอบด้วยสถิติวิลคอกซัน (Wilcoxon Signed Ranks Test) โดยใช้ค่ากลางของข้อมูลในการทดสอบ กำหนดสมมติฐานดังนี้

H_0: ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์มีความคาดเคลื่อนน้อยกว่าหรือเท่ากับร้อยละ 25

H_1: ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์มีความคาดเคลื่อนมากกว่าร้อยละ 25

โดยจะปฏิเสธ H_0 ถ้าค่า Sig. (Significance) จากการทดสอบมีค่าน้อยกว่าระดับนัยสำคัญที่กำหนด ในงานวิจัยนี้กำหนดระดับนัยสำคัญที่ 0.05

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตาราง 4.19 ผลการทดสอบความแม่นยำในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

Null Hypothesis	Test	sig.
ค่ากลาง MRE น้อยกว่าหรือเท่ากับ 0.25	One-Sample Wilcoxon Sined Rank Test	0.0005*

ผลการทดสอบพบว่า ค่า Sig.(2-tailed) มีค่า 0.001 แต่เนื่องจากการทดสอบนี้ใช้การทดสอบสมมติฐานแบบทางเดียวทำให้ค่า Sig. ที่ได้ต้องนำมาหารด้วย 2 ซึ่งได้ค่า Sig. เท่ากับ 0.0005 มีค่าน้อยกว่าระดับนัยสำคัญ คือ 0.05 จึงปฏิเสธ H_0 หมายความว่า ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์มีความคาดเคลื่อนมากกว่าร้อยละ 25

จึงสรุปว่าวิธียูสเคสพอยต์น่าจะไม่เหมาะสมในการใช้ประมาณการค่าความพยายามของข้อมูลในงานวิจัยชิ้นนี้

4.4.4. ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

จากข้อมูลการประมาณค่าความพยายามของข้อมูลทั้ง 17 ชุดข้อมูลด้วยวิธีของ Nikunj Sakhrelia พบว่า ค่าความพยายาม มีค่าตั้งแต่ 28 - 260 คน-ชั่วโมง ค่าเฉลี่ยและค่ามัธยฐานของค่าความพยายามอยู่ที่ 111.53 และ 88.00 คน-ชั่วโมง ตามลำดับ เมื่อนำมาคำนวณค่าความคาดเคลื่อนโดยเทียบเคียง จะได้ค่าดังตาราง 4.20

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 4.20 แสดงผลการคำนวณค่า MRE ของวิธีของ Nikunj Sakhrelia

ลำดับ	ชื่อแอพพลิเคชั่น	ค่าความพยายามจากวิธี NS (Man-hour)	ค่าความพยายามจริง (Man-hour)	MRE
1	แอพพลิเคชั่นคำนวณค่าโดยสารรถไฟฟ้า BTS	88	192	0.542
2	แอพพลิเคชั่นแสดงหุ้น	84	272	0.691
3	แอพพลิเคชั่นนำทาง	40	288	0.861
4	แอพพลิเคชั่นข้อมูลข่าวสาร	92	116	0.207
5	แอพพลิเคชั่นข้อมูลภาพยนตร์	136	372	0.634
6	แอพพลิเคชั่นข้อมูลการท่องเที่ยว	188	504	0.627
7	แอพพลิเคชั่นดิกชันนารี	28	28	0.000
8	แอพพลิเคชั่นสำรองข้อมูลรายชื่อติดต่อ	76	176	0.568
9	แอพพลิเคชั่นรายงานผลฟุตบอล	248	364	0.319
10	แอพพลิเคชั่นดูวิดีโอออนไลน์	212	120	0.767
11	แอพพลิเคชั่นข้อมูลซีรีย์	80	22	2.636
12	แอพพลิเคชั่นบันทึกรายรับ-รายจ่าย	116	224	0.482
13	แอพพลิเคชั่นตรวจสอบสัญญาณไวไฟ(wi-fi)	28	24	0.167
14	แอพพลิเคชั่นข้อมูลโรงพยาบาล	260	200	0.300
15	แอพพลิเคชั่นฟังวิทยุออนไลน์	116	160	0.275
16	แอพพลิเคชั่นดูโฆษณาออนไลน์	44	120	0.633
17	แอพพลิเคชั่นQR Reader	60	96	0.375
ค่าเฉลี่ย (MMRE)				**0.593**

จากตารางจะพบว่าค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีของ Nikunj Sakhrelia ในแต่ละชุดข้อมูลมีค่าตั้งแต่ 0 – 2.636 และมีค่าเฉลี่ยอยู่ที่ 0.593 นั่นหมายความว่าวิธีของ Nikunj Sakhreliaให้ค่าเบี่ยงเบนไปจากข้อมูลจริงร้อยละ 59.3 แสดงให้เห็นว่ามีความคาดเคลื่อนค่อนข้างมาก และมีอัตราร้อยละของระบบที่ประมาณค่าความพยายามได้ถูกต้องเท่ากับ 0.176 หมายถึงมีจำนวนร้อยละ 17.64 ของชุดข้อมูลทั้งหมดที่ประมาณค่าความพยายามผิดพลาดไม่เกินร้อยละ 25 (PRED(0.25))

ผู้วิจัยทำการทดสอบความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhreliaในทางสถิติ ทั้งนี้ การประมาณขนาดซอฟต์แวร์ที่ยอมรับ

ได้จะประมาณค่าความพยายามในการพัฒนาซอฟต์แวร์ได้ผิดพลาดไม่เกินร้อยละ 25 หรือมีค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง(MRE) น้อยกว่าหรือเท่ากับ 0.25 (Conte et al., 1986) โดยทางทำการทดสอบการแจกแจงปกติของข้อมูลค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีของ Nikunj Sakhrelia หากมีการแจกแจงแบบปกติ ผู้วิจัยจะเลือกใช้สถิติพาราเมตริก ในการทดสอบสมมติฐาน แต่หากข้อมูลไม่มีการแจกแจงแบบปกติ จะใช้สถิตินอนพาราเมตริก ในการทดสอบแทน

การทดสอบการแจกแจงปกติของข้อมูล

ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามที่ได้จากวิธีของ Nikunj Sakhrelia ใช้สมมติฐานในการทดสอบการแจกแจงของข้อมูล คือ

H_0: ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia (MRE_{NS}) มีการแจงแจกปกติ

H_1: ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia (MRE_{NS}) ไม่มีการแจงแจกปกติ

กำหนดระดับนัยสำคัญเป็น 0.05 (α = 0.05)

โดยงานวิจัยนี้มีจำนวนหน่วยตัวอย่าง 17 หน่วย ซึ่งน้อยกว่า 50 หน่วย ผู้วิจัยจึงเลือกใช้การทดสอบ Shapiro-Wilk โดยจะปฏิเสธ H_0 หากค่า Sig. ของการทดสอบน้อยกว่า ระดับนัยสำคัญที่กำหนด ซึ่งคือ 0.05 ผลสรุปการทดสอบการแจกแจงปกติของค่าเฉลี่ยของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia ดังตาราง 4.21

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตาราง 4.21 แสดงผลการทดสอบการแจกแจงข้อมูลของวิธีของ Nikunj Sakhrelia

	ค่าความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia
ค่า Sig.	0.000*
การแจกแจง	ไม่มีการแจกแจงปกติ

จากการทดสอบพบว่าข้อมูลไม่มีการแจกแจงปกติดังนั้นผู้วิจัยจึงเลือกการทดสอบโดยใช้สถิตินอนพาราเมตริก ด้วยการทดสอบด้วยสถิติวิลคอกซัน (Wilcoxon Signed Ranks Test) โดยใช้ค่ากลางของข้อมูลในการทดสอบ กำหนดสมมติฐานดังนี้

H_0: ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhreliaมีความคาดเคลื่อนน้อยกว่าหรือเท่ากับร้อยละ 25

H_1: ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhreliaมีความคาดเคลื่อนมากกว่าร้อยละ 25

โดยจะปฏิเสธ H_0 ถ้าค่า Sig. (Significance) จากการทดสอบมีค่าน้อยกว่าระดับนัยสำคัญที่กำหนด ในงานวิจัยนี้กำหนดระดับนัยสำคัญที่ 0.05

ตาราง 4.22 ผลการทดสอบความแม่นยำในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

Null Hypothesis	Test	sig.
ค่ากลาง MRE น้อยกว่าหรือเท่ากับ 0.25	One-Sample Wilcoxon Sined Rank Test	0.002*

ผลการทดสอบพบว่า ค่า Sig.(2-tailed) มีค่า 0.004 แต่เนื่องจากการทดสอบนี้ใช้การทดสอบสมมติฐานแบบทางเดียวทำให้ค่า Sig. ที่ได้ต้องนำมาหารด้วย 2 จึงได้ค่า Sig. เท่ากับ 0.002 มีค่าน้อยกว่าระดับนัยสำคัญ คือ 0.05 จึงปฏิเสธ H_0 หมายความว่า ความแม่นยำของการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia มีความคาดเคลื่อนมากกว่าร้อยละ 25

จึงสรุปว่าวิธีของ Nikunj Sakhrelia น่าจะไม่เหมาะสมในการใช้ประมาณการค่าความพยายามของข้อมูลในงานวิจัยชิ้นนี้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.5 การเปรียบเทียบความแม่นยำของวิธีการประมาณการค่าความพยายามทั้ง 4 วิธี

การเปรียบเทียบความแม่นยำจะใช้การเปรียบเทียบค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง (MMRE) ของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นของวิธีการทั้ง 4 ด้วยการวิเคราะห์ความแปรปรวน (Analysis of Variance: ANOVA) และเนื่องจากมีข้อมูลที่ไม่มีการแจกแบบปกติ จึงเลือกใช้ วิธีการทดสอบ Kruskal –Wallis ในการทดสอบ โดยมีสมมติฐานการทดสอบ คือ

H_0: วิธีการประมาณขนาดซอฟต์แวร์ที่แตกต่างกันไม่ส่งผลต่อความแม่นยำในการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่น ($\mu_1 = \mu_2 = \mu_3 = \mu_4$)

H_1: วิธีการประมาณขนาดซอฟต์แวร์ที่แตกต่างกันส่งผลต่อความแม่นยำการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่น ($\mu_i \neq \mu_j$ เมื่อ $i \neq j$)

กำหนดระดับนัยสำคัญเป็น 0.05 ($\alpha = 0.05$)

μ_1 คือ ค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง($MMRE_{FP}$) ของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นของด้วยวิธีฟังก์ชั่นพอยต์

μ_2 คือ ค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง($MMRE_{OP}$) ของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์

μ_3 คือ ค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง($MMRE_{UCP}$) ของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

μ_4 คือ ค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง($MMRE_{NS}$) ของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

โดยจะปฏิเสธ H_0 ถ้าค่า Sig. (Significance) จากการทดสอบมีค่าน้อยกว่าระดับนัยสำคัญที่กำหนด ในงานวิจัยนี้กำหนดระดับนัยสำคัญที่ 0.05

ตาราง 4.23 แสดงผลการเปรียบเทียบความแม่นยำของวิธีการประมาณค่าความพยายามทั้ง 4 วิธี

Methods	N	Mean Rank	Chi-Square	df	Sig.
FP	17	33.91	15.067	3	0.002*
OP	17	25.18			
UCP	17	49.71			
NS	17	29.21			

จากผลการทดสอบค่า Sig. (Significance) มีค่าเท่ากับ 0.002 ซึ่งน้อยกว่าค่านัยสำคัญที่งานวิจัยนี้กำหนด (0.05) ทำให้สามารถปฏิเสธ H_0 นั่นหมายความว่า มีค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นอย่างน้อยหนึ่งคู่ไม่เท่ากัน กล่าวคือ วิธีการประมาณขนาดซอฟต์แวร์ที่แตกต่างกันส่งผลต่อความแม่นยำการประมาณค่าความพยายามที่ใช้ในการพัฒนาโมบายแอพพลิเคชั่น

เนื่องจากพบว่า มีค่าเฉลี่ยค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นอย่างน้อยหนึ่งคู่ไม่เท่ากัน จึงทำการทดสอบเปรียบเทียบค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียงของค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นรายคู่ ด้วยการทดสอบของแมนและวิทนีย์ (The Mann-Whitney U Test)

โดยผลการทดสอบด้วยการทดสอบของแมนและวิทนีย์ (The Mann-Whitney U Test) ปรากฏว่า ไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.05 ระหว่าง 3 คู่ทดสอบ คือ

1) วิธีฟังก์ชั่นพอยต์และวิธีอ็อบเจกต์พอยต์ ค่า Sig. มีค่า 0.113 มากกว่า ค่านัยสำคัญ

2)วิธีฟังก์ชั่นพอยต์และวิธีของ Nikunj Sakhrelia ค่า Sig. มีค่า 0.418 มากกว่าค่านัยสำคัญ

3) วิธีอ็อบเจกต์พอยต์และวิธีของ Nikunj Sakhrelia ค่า Sig. มีค่า 0.389 มากกว่าค่านัยสำคัญ

การทดสอบพบความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.05 ระหว่าง 3 คู่ คือ

1) วิธีฟังก์ชั่นพอยต์และวิธียูสเคสพอยต์ ค่า Sig. มีค่า 0.006 น้อยกว่า ค่านัยสำคัญ

2) วิธีอ็อบเจกต์พอยต์และวิธียูสเคสพอยต์ ค่า Sig. มีค่า 0.003 น้อยกว่า ค่านัยสำคัญ

3)วิธียูสเคสพอยต์และวิธีของ Nikunj Sakhrelia ค่า Sig. มีค่า 0.002 น้อยกว่าค่านัยสำคัญ

ซึ่งให้ผลการทดสอบตามตาราง 4.24

ตาราง 4.24 แสดงผลการทดสอบความแม่นยำของวิธีการประมาณค่าความพยายามรายคู่

การทดสอบ		**จำนวน**	**Mann-Whitney U Value**	**Sig.**
FP	OP	17	98.5	0.113
	UCP	17	65	0.006*
	NS	17	121	0.418
OP	UCP	17	57	0.003*
	NS	17	119.5	0.389
UCP	NS	17	53	0.002*

4.6 สรุปผลการวิเคราะห์ข้อมูล

งานวิจัยนี้ได้ทำการทดสอบวิธีการประมาณการค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น 4 วิธี โดยการเปรียบเทียบค่าเฉลี่ยค่าความพยายามที่ได้จากการประมาณการ และความแม่นยำของการประมาณการโดยพิจารณาค่าเฉลี่ยความคาดเคลื่อนโดยเทียบเคียง ซึ่งให้ผลการทดสอบว่าดังตาราง 4.25

ตาราง 4.25 สรุปผลการเปรียบเทียบวิธีการประมาณค่าความพยายามทั้ง 4 วิธี

คู่ทดสอบ		ผลการทดสอบ	
		ค่าความพยายาม	ค่าความแม่นยำ
FP	OP	แตกต่าง	ไม่แตกต่าง
	UCP	แตกต่าง	แตกต่าง
	NS	ไม่แตกต่าง	ไม่แตกต่าง
OP	UCP	แตกต่าง	แตกต่าง
	NS	แตกต่าง	ไม่แตกต่าง
UCP	NS	แตกต่าง	แตกต่าง

การพิจารณาเปรียบเทียบค่าความพยายามของทั้ง 4 วิธีนั้นพบว่า วิธีการประมาณขนาดที่แตกต่างกันจะส่งผลต่อค่าความพยายามในการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น เมื่อทดสอบรายคู่พบว่าวิธีฟังก์ชั่นพอยต์และวิธีของ Nikunj Sakhrelia นั้นให้ผลการประมาณค่าความพยายามที่ไม่แตกต่างกัน และให้ผลแตกต่างกับวิธีอ็อบเจกต์พอยต์ และวิธียูสเคสพอยต์

ในส่วนของความแม่นยำนั้นวิธีฟังก์ชั่นพอยต์ วิธีอ็อบเจกต์พอยต์ และวิธีของ Nikunj Sakhrelia นั้นให้ค่าไม่แตกต่างกัน แต่เมื่อพิจารณาค่าอัตราร้อยละที่ทำนายได้ถูกต้องพบว่า ไม่มีวิธีการใดที่ประมาณการได้แม่นยำเกินค่าที่ยอมรับได้ หรือหมายถึงไม่มีวิธีการใดที่มีค่าเฉลี่ยความคลาดเคลื่อนโดยเทียบเคียง(MMRE) น้อยกว่าหรือเท่ากับ 0.25

บทที่ 5
สรุปผลการวิจัย

5.1 สรุปผลการวิจัย

งานวิจัยนี้เป็นการเปรียบเทียบวิธีการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นด้วยวิธีการที่แตกต่างกัน 4 วิธี คือ วิธีฟังก์ชั่นพอยต์ วิธีอ็อบเจกต์พอยต์ วิธียูสเคสพอยต์ และ วิธีของ Nikunj Sakhrelia โดยใช้ข้อมูลแอพพลิเคชั่นที่พัฒนาขึ้นจริงเป็นข้อมูลในการทดสอบวิธีการทั้ง 4 วิธี โดยใช้ข้อมูล 17 ชุดข้อมูล ให้ผลการทดสอบพบว่า วิธีการประมาณการขนาดซอฟต์แวร์ที่แตกต่างกันส่งผลต่อการประมาณค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่น เมื่อแยกพิจารณาเปรียบเทียบค่าความพยายามของแต่ละวิธีเป็นรายคู่ ทั้งหมด 6 คู่ทดสอบ ได้แก่

1) วิธีฟังก์ชั่นพอยต์และวิธีอ็อบเจกต์พอยต์
2) วิธีฟังก์ชั่นพอยต์และวิธียูสเคสพอยต์
3) วิธีฟังก์ชั่นพอยต์และวิธีของ Nikunj Sakhrelia
4) วิธีอ็อบเจกต์พอยต์และวิธียูสเคสพอยต์
5) วิธีอ็อบเจกต์พอยต์และวิธีของ Nikunj Sakhrelia
6) วิธียูสเคสพอยต์และวิธีของ Nikunj Sakhrelia

พบว่า มีเพียงคู่ทดสอบที่ 3 คือคู่ระหว่างวิธีฟังก์ชั่นพอยต์และวิธีของ Nikunj Sakhrelia เท่านั้นที่ให้ค่าการประมาณการค่าความพยายามไม่แตกต่างกัน และยังพบว่าวิธียูสเคสพอยต์จะให้ค่าการประมาณการค่าความพยายามสูงกว่าวิธีอื่น

แต่หากพิจารณาความแม่นยำในการประมาณการค่าความพยายามพบว่า คู่ทดสอบที่ 1 3 และ 5 ให้ค่าความแม่นยำที่ไม่แตกต่างกัน นั่นหมายถึงว่ามีเพียงวิธีการยูสเคสพอยต์วิธีการเดียวที่มีความแม่นยำแตกต่างจากวิธีการอื่น

ผลสรุปที่ได้สอดคล้องกับการวิจัยของ Issa และคณะในปี ค.ศ. 2007 ที่กล่าวว่าความแม่นยำในการประมาณการค่าความพยายามของวิธีฟังก์ชั่นพอยต์และวิธีอ็อบเจกต์พอยต์ ไม่มีความแตกต่างกันอย่างมีนัยสำคัญ(Issa et al., 2007)

5.2 อภิปรายผล

งานวิจัยนี้ได้เลือกใช้ข้อมูลโมบายแอพพลิเคชั่นที่พัฒนาขึ้นจริงและในการประมาณการค่าความพยายามการเก็บข้อมูลซึ่งเป็นความลับ ทำให้การเก็บข้อมูลของงานวิจัยนี้เป็นไปได้ยาก ทำให้

ข้อมูลที่ได้มีปริมาณน้อยเมื่อเทียบกับโมบายแอพพลิเคชั่นทั้งหมดที่มีวางจำหน่าย ทำให้ข้อมูลที่ใช้ในงานวิจัยนี้ยังไม่มีความหลากหลายในประเภทของโมบายแอพพลิเคชั่น จึงทำให้ผู้วิจัยไม่สามารถเปรียบเทียบค่าประมาณการตามประเภทของโมบายแอพพลิเคชั่นที่มีทั้งหมด ซึ่งอาจจะทำให้เห็นถึงความแตกต่างของผลการทดสอบในแอพพลิเคชั่นที่มีประเภทแตกต่างกัน

ในการเก็บข้อมูลนั้นจำเป็นต้องเก็บข้อมูลปัจจัยที่มีอิทธิพลต่อการพัฒนาโมบายแอพพลิเคชั่น ซึ่งโดยปกติข้อมูลเหล่านี้จะต้องเก็บในช่วงต้นของการพัฒนา แต่ในการวิจัยครั้งนี้เป็นการเก็บข้อมูลของโมบายแอพพลิเคชั่นที่ได้รับการพัฒนาเสร็จแล้ว จึงทำให้นักพัฒนาผู้ให้ข้อมูลจำเป็นต้องนึกย้อนไปถึงช่วงต้นของการพัฒนา และเนื่องจากระยะเวลาผ่านมาไปทำให้ทัศนคติของผู้ให้ข้อมูลอาจจะเปลี่ยนไปทำให้ข้อมูลเหล่านี้อาจมีความคาดเคลื่อน

ข้อมูลของโมบายแอพพลิเคชั่นที่นำมาใช้ในงานวิจัยนี้เป็นข้อมูลของโมบายแอพพลิเคชั่นที่พัฒนาเสร็จแล้ว ทำให้ในการประมาณการค่าความพยายามซึ่งโดยปกติจะต้องใช้ข้อมูลจากขั้นตอนการออกแบบนั้นอาจจะเกิดความคาดเคลื่อนขึ้นได้จากข้อมูลที่อาจจะมีรายละเอียดที่แตกต่างกันระหว่างข้อมูลของโมบายแอพพลิเคชั่นที่พัฒนาเสร็จแล้วและข้อมูลโมบายแอพพลิเคชั่นในช่วงการออกแบบ

การประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีอ๊อบเจกต์พอยต์และวิธีของ Nikunj Sakhrelia นั้นเป็นวิธีการที่ค่อนข้างหยาบ การระบุองค์ประกอบต่างๆในการประมาณขนาดมีเกณฑ์ในการพิจารณาที่ไม่ละเอียดมาก จึงทำให้ในการประมาณขนาดนั้นอาจจะเกิดความคาดเคลื่อนของค่าประมาณการที่เกิดจากมุมมองของผู้วิจัยได้

สำหรับวิธียูสเคสนั้นผู้วิจัยจำเป็นต้องนำข้อมูลมาวาดเป็นแผนภาพยูสเคสซึ่งในการวาดแผนภาพยูสเคสอาจจะมีความแตกต่างกันได้จากมุมมองของผู้วาดแผนภาพ ซึ่งความแตกต่างของแผนภาพยูสเคสจะส่งผลต่อค่าความพยายามที่ประมาณการได้จากวิธียูสเคสพอยต์ แต่ทั้งนี้ผู้วิจัยได้ส่งแผนภาพยูสเคสที่ได้ทำการวาดให้แก่นักพัฒนาเพื่อตรวจสอบความถูกต้อง เพื่อลดความผิดพลาดในส่วนนี้

วิธีการยูสเคสพอยต์นั้นในการวิจัยนี้ผู้เลือกใช้ค่าความสามารถในการพัฒนาซอฟต์แวร์ตามงานวิจัยของ Schneider และ Winters อาจเพราะเมื่อเทคโนโลยีเปลี่ยนไป เครื่องมือในการพัฒนามีการอำนวยความสะดวกมากขึ้น ทำให้ค่าความสามารถในการพัฒนาซอฟต์แวร์ของนักพัฒนาดีขึ้น ค่าดังกล่าวอาจจะไม่เหมาะสมแล้วในปัจจุบัน

ชุดข้อมูลของโมบายแอพพลิเคชั่นที่นำมาใช้ในงานวิจัยนี้เป็นโมบายแอพพลิเคชั่นที่พัฒนาขึ้นใหม่ทั้งหมด ซึ่งในการพัฒนาซอฟต์แวร์เชิงวัตถุมักจะมีการใช้โค้ดซ้ำ (Reuse Code) ทำให้ใช้เวลาในการพัฒนาน้อยลง ซึ่งในการนำวิธีการประมาณค่าความพยายามที่ใช้ในงานวิจัยนี้ไปใช้กับโมบายแอพพลิเคชั่นที่มีการใช้โค้ดซ้ำอาจจะให้ค่าการประมาณค่าความพยายามที่แตกต่างออกไป

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

วิธีการของ Nikunj Sakhrelia ซึ่งเป็นวิธีการที่พัฒนามาจากประสบการณ์ในการพัฒนาโมบายแอพพลิเคชั่นของโดยเฉพาะ ทำให้ค่าน้ำหนักขององค์ประกอบแต่ละประเภทจึงอยู่บนพื้นฐานของค่าความสามารถในการพัฒนาซอฟต์แวร์ของผู้คิดค้นวิธีการนี้ ซึ่งอาจจะมีความเหมาะสมทำให้วิธีการนี้มีค่าความคาดเคลื่อนน้อยที่สุด หรืออาจจะหมายถึงในปัจจุบันเครื่องมือในการพัฒนาโมบายแอพพลิเคชั่นหรือเทคโนโลยีที่เกิดขึ้นใหม่ ทำให้นักพัฒนาแต่ละคนมีค่าความสามารถในการพัฒนาซอฟต์แวร์ที่ไม่แตกต่างกัน

5.3 การนำงานวิจัยไปใช้

ข้อค้นพบในงานวิจัยนี้ ช่วยต่อยอดองค์ความรู้ในการประมาณการค่าความพยายาม โดยมีรายละเอียดดังนี้

1. ในการตัดสินใจเลือกใช้วิธีการประมาณการค่าความพยายามในการพัฒนาโมบายแอพพลิเคชั่นนั้น ขึ้นอยู่กับนักพัฒนาว่ามีวิธีการออกแบบแอพพลิเคชั่นอย่างไรหากนักพัฒนาออกแบบโมบายแอพพลิเคชั่นโดยใช้แบบจำลองหน้าจอ ก็ควรเลือกวิธีอ็อบเจกต์พอยต์หรือวิธีของ Nikunj Sakhrelia หากมีการเขียนความต้องการแบบระบุเป้าหมายและสถานการณ์ก็สามารถเลือกใช้วิธีฟังก์ชั่นพอยต์ได้ เพราะวิธีการทั้ง 3 วิธีให้ความแม่นยำไม่แตกต่างกัน
2. วิธีฟังก์ชั่นพอยต์และวิธียูสเคสพอยต์ มีขั้นตอนในการประมาณค่าความพยายามมากกว่าวิธีการอื่น ถ้าหากเลือกใช้ 2 วิธีนี้ ก็จะทำให้ต้องเสียเวลามาก

การเรียนรู้และทำความเข้าใจวิธีการฟังก์ชั่นพอยต์และวิธียูสเคสพอยต์นั้นจะต้องใช้เวลาทำความเข้าใจมากเพราะมีเงื่อนไขในการระบุองค์ประกอบต่างๆค่อนข้างละเอียด ส่วนวิธีการอ็อบเจกต์พอยต์และวิธีของ Nikunj Sakhrelia นั้นมีขั้นตอนน้อยทำให้เป็นวิธีการที่เข้าใจได้ง่ายกว่า

Chulalongkorn University

รายการอ้างอิง

ภาษาไทย

กัลยา วานิชย์บัญชา. 2551. หลักสถิติ . พิมพ์ครั้งที่ 8. กรุงเทพฯ: โรงพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย.

ชมพูนุช เผ่าประพัธน์.การเปรียบเทียบความแม่นยำของวิธีการประมาณขนาดซอฟต์แวร์เชิงวัตถุด้วยวิธีฟังก์ชันพอยต์ ฟังก์ชันพอยต์เชิงวัตถุ และคลาสพอยต์. วิทยานิพนธ์ ปริญญามหาบัณฑิต, สาขาวิชาการพัฒนาซอฟต์แวร์ด้านธุรกิจ ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย, 2555.

วงหทัย ตันชีวะวงศ์.ปัจจัยที่มีผลต่อการใช้โมบายแอพพลิเคชั่นบนสมาร์ทโฟนและแท็บเล็ต. [ออนไลน์].2557.แหล่งที่มา: http://203.131.210.100/research/?p=252 [14 กุมพาพันธ์ 2557]

วรัฎฐา เจริญสถาพงษ์. วิธีการประมาณต้นทุนสำหรับการพัฒนาเว็บแอพพลิเคชั่น. วิทยานิพนธ์ปริญญามหาบัณฑิต, สาขาวิชาการพัฒนาซอฟต์แวร์ด้านธุรกิจ ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย, 2549.

สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ. 2556. แผนยุทธศาสตร์การส่งเสริมอุตสาหกรรมซอฟต์แวร์(พ.ศ.2555-2558) ฉบับปรับปรุง ปี พ.ศ.2556. สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ กระทรวงวิทยาศาสตร์และเทคโนโลยี.

สุชาดา พลาชัยภิรมย์ศิล.แนวโน้มการใช้โมบายแอพพลิเคชั่น Usages Trend of Mobile Application. วารสารนักบริหาร 31 : 110-115.

อัษฎาพร ทรัพย์สมบูรณ์.การวิเคราะห์และออกแบบเชิงวัตถุ (Object Oriented Analysis and Design). กรุงเทพ:เคทีพี คอมพ์ แอนด์ คอนซัลท์,2554.312 หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาษาอังกฤษ

Albrecht, AJ., and Gaffney, JE. 1983. Software function, source lines of code, and development effort prediction: a software science validation. IEEE Transactions on Software Engineering SE-9(6): 639–648.

Arhippainen, L., and Tähti, M.2003. Empirical evaluation of user experience in two adaptive mobile application prototypes. In Proceedings of the 2nd international conference on mobile and ubiquitous multimedia (pp. 27-34).

Banker, R.D., Kauffman, R.J., Wright, C., and Zweig, D. Automating output size and reuse metrics in a repository-based computer-aided software engineering (CASE) environment. IEEE Transactions on Software Engineering Vol.20 No.3 (1994): 169–187.

Choi,S., Park,S., and Sugumaran,V. Function Point Extraction Method from Goal and Scenario Based Requirements Text, Natural Language Processing and Information Systems, pp. 12-24, 2006.

Conte, S., Dunsmore, H., and Shen, V. 1986. Software Engineering Metrics and Models. Menlo Park, Benjamin/Cummings

Fetcke. T., Abran, A., and Nguyen, T. 1997. Mapping the OO-Jacobson Approach into Function Point Analysis. Proceedings of TOOLS-23'97, 28 July – 1 August 1997

Framingham, M. Android Pushes Past 80% Market Share While Windows Phone Shipments Leap 156.0% Year over Year in the Third Quarter[Online].2013.http://www.idc.com/getdoc.jsp?containerId=prUS24442013 [2014, February 27]

Gartner, Inc. Market Share Analysis: Mobile Phones Worldwide, 1Q13.Market Analysis and statics [Online]. 2013. Available from: http://www.gartner.com/newsroom/id/2482816 [2013, July 16]

Google Developer Groups (GDGs). Mobile Apps Categories. [Online].2014.Available from: https://developers.google.com/adwords/api/docs/appendix/mobileappcategories [2014, Feb 22]

Holzinger, A., Treitler, P., and Slany, W. 2012. Making apps useable on multiple different mobile platforms: On interoperability for business application development on smart phones. In: Quirchmayr G, Basl J, You I et al. (eds). Multidisciplinary research and practice for information systems. Springer, Berlin Heidelberg, pp 176–189

IFPUG. 2000. Function Point Counting Practices Manual Release 4.1.1. International Function Point Users Group (IFPUG).

Issa, A., Odeh, M., and Coward,D. Can Function Points Be Mapped to Object Points?. International Arab Journal of Information Technology 4 (1) (2007): 1.

Joorabchi, M. E., Mesbah, A., and Kruchten, P.2013. Real Challenges in Mobile App Development. In Empirical Software Engineering and Measurement, 2013 ACM/IEEE International Symposium on (pp. 15-24). IEEE.

Jørgensen, M. 2014. Failure factors of small software projects at a global outsourcing marketplace. Journal of Systems and Software.

Jørgensen, M., and Sjøberg, D. I. (2001). Impact of effort estimates on software project work. Information and software technology, 43(15), 939-948.

Kang, K.C., Kim, S., Lee, J., Kim, K., Shin, E., and Huh, M. FORM: A feature–oriented reuse method with domain–specific reference architectures. Annals of Software Engineering, 5:143–168, 1998.

Karner, G. Resource Estimation for Objectory Projects. Objective Systems. , 1993 อ้างถึงใน ชมพูนุช เผ่าประพัธน์.การเปรียบเทียบความแม่นยำของวิธีการประมาณขนาดซอฟต์แวร์เชิงวัตถุด้วยวิธีฟังก์ชันพอยต์ ฟังก์ชันพอยต์เชิงวัตถุ และคลาสพอยต์. วิทยานิพนธ์ ปริญญามหาบัณฑิต, สาขาวิชาการพัฒนาซอฟต์แวร์ด้านธุรกิจ ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย, 2555.

Kim, J., Park, S., and Sugumaran, V. 2004. A linguistics-based approach for use case driven analysis using goal and scenario authoring. In Natural Language Processing and Information Systems (pp. 159-170). Springer Berlin Heidelberg.

Linberg, K. R. 1999. Software developer perceptions about software project failure: A Case study.
Journal of Systems and Software 49. อ้างถึงใน Verner, J.M., Sampson, J., Cerpa, N., 2008. What factors lead to software project failure? In: Research Challenges in Information Science, Marrakech, Morocco, pp.71–80.

Longstreet,D. Function Point Analysis Training Course. Charleston. South Carolina. Longstreet Consulting Inc., 2004, อ้างถึงใน วรัฏฐา เจริญสถาพงษ์. วิธีการประมาณต้นทุนสำหรับการพัฒนาเว็บแอพพลิเคชั่น. วิทยานิพนธ์ปริญญามหาบัณฑิต, สาขาวิชาการพัฒนาซอฟต์แวร์ด้านธุรกิจ ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย, 2549.

Malathi, S. and Sridhar, S. ANALYSIS OF SIZE METRICS AND EFFORT PERFORMANCE CRITERION IN SOFTWARE COST ESTIMATION. Indian Journal of Computer Science and Engineering (IJCSE) (Vol. 3 No. 1 Feb-Mar 2012): 24-31

Portio Research. Mobile Application Features 2013-2017[Online].2013.Available from:http://www.portioresearch.com/en/major-reports/current-portfolio/mobile-applications-futures-2013-2017.aspx [2013, March 14]

Ribu, K. Estimating object-oriented software projects with use cases, Master's thesis, University of Oslo, Department of Informatics, 2001.

Rubinstein, D. 2007. Standish group report: There's less development chaos today.SD Times, [Online]. Available from: http://www.sdtimes.com/link/30247l [2013 July 16]

Sakhrelia, N. 2012. How to estimate mobile app project effort?, [Online]. Available from: http://developer.appcelerator.com/question/128655/how-to-estimate-mobile-app-project-effort- [2013 August 28]

Saxena,V., and Shrivastava,M. Performance of function point analysis through UML modeling. ACM SIGSOFT Software Engineering Notes Volume 34 Number 2 (March 2009): 1-4.

Sudhakara, G. P., Farooqb, A., and Patnaikc, S. (2012). Measuring productivity of software development teams. Serbian Journal of Management, 7(1)

Tracy, K.W. 2012. Mobile Application Development Experiences on Apple's iOS and Android OS. IEEE Potentials, 31(4), pp.30–34.

Verner, J.M., Sampson, J., and Cerpa, N. 2008. What factors lead to software project failure? In: Research Challenges in Information Science, Marrakech, Morocco, pp.71–80.

Wasserman, T. 2010. Software engineering issues for mobile application development. FoSER 2010.

Zheng, Y., Conghui, L., Niemi,V., and Guoliang, Y. Effects of Displaying Trust Information on Mobile Application Usage. Proceedings of 7th International Conference ATC 2010 Xi'an.107-1

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก ก

แบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น

แบบฟอร์มส่วนที่ 1

Application Name : _____

Detail : _____

_____

_____

_____

Owner: _____ Date: _____

No. of Function: _____ No. of Screen: _____

Actual Effort: _____ Man - Hour

Data File Name	Type	No. of Data Field	No.of Subgroup

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แบบฟอร์มส่วนที่ 2

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แบบฟอร์มส่วนที่ 3

คำชี้แจง ระบุถึงปัจจัยที่ต้องคำนึงถึงและส่งผลต่อความซับซ้อนในการพัฒนา
โดยที่ผู้ให้ข้อมูลจะต้องนึกย้อนไปถึงอิทธิพลของปัจจัยต่างๆในช่วงของการออกแบบระบบ
ระดับอิทธิพลของปัจจัยที่เป็นไปได้ คือ 0 ถึง 5 โดยให้กรอกตามคำแนะนำในเอกสารที่แนบมาด้วย

ลำดับ	ปัจจัย	ระดับอิทธิพล	ลำดับ	ปัจจัย	ระดับอิทธิพล
F1	Data Communication		T5	Reusable code	
F2	Distribution Data Processing		T6	Easy to install	
F3	Performance		T7	Easy to use	
F4	Heavily Used Configuration		T8	Portable	
F5	Transaction Rate		T9	Easy to change	
F6	On-line Data Entry		T10	Concurrent	
F7	End-User Efficiency		T11	Security features	
F8	Online Update		T12	Access for third-parties	
F9	Complex Processing		T13	Special training required	
F10	Reusability		E1	Familiar with development method used	
F11	Installation Ease		E2	Application experience	
F12	Operational Ease		E3	Object-Oriented experience	
F13	Multiple Sites		E4	Lead analyst capability	
F14	Facilitate change		E5	Motivation	
T1	Distributed System		E6	Stable Requirements	
T2	Response adjectives		E7	Part-time workers	
T3	End-user efficiency		E8	Difficulty of programming language	
T4	Complex processing				

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

คำแนะนำในการกรอกแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นส่วนที่ 3

F1.Data Communication จะมองเรื่องของจำนวนของสิ่งที่ช่วยอำนวยความสะดวกในการถ่ายโอนข้อมูล หรือแลกเปลี่ยนข้อมูลระหว่างซอฟต์แวร์หรือระบบ	
0	ซอฟต์แวร์ทั้งหมดรันแบบ Batch หรือ standalone PC
1	ซอฟต์แวร์เป็นแบบ Batch แต่มีการใช้ลักษณะของ Remote Data Entry หรือ Remote Printing
2	ซอฟต์แวร์เป็นแบบ Batch แต่มีการใช้ลักษณะของ Remote Data Entry และ Remote Printing
3	ซอฟต์แวร์เป็นแบบ Online Data Collection หรือ *Teleprocessing จากส่วน Front-end สู่ Batch
4	ซอฟต์แวร์มีลักษณะของ Teleprocessing ซึ่งมากกว่าในส่วนของ Front-end แต่สนับสนุนการติดต่อ Teleprocessing Protocol เพียงชนิดเดียว
5	ซอฟต์แวร์มีลักษณะของ Teleprocessing ซึ่งมากกว่าในส่วนของ Front-end แต่สนับสนุนการติดต่อ Teleprocessing Protocol ที่หลากหลายชนิดมากขึ้น

F2. Distributed Data Processing จะมองในเรื่องการจัดการข้อมูลในเชิง Distributed และขั้นตอนหรือหน้าที่ในการจัดการ	
0	ซอฟต์แวร์สามารถช่วยในการถ่ายข้อมูลหรือช่วยในการดำเนินการระหว่างกันได้
1	ซอฟต์แวร์สามารถเตรียมข้อมูลสำหรับ End User เพื่อดำเนินขั้นตอนต่อไปในส่วนอื่นๆ เช่น PC Spreadsheets และ PC DBMS
2	ซอฟต์แวร์สามารถเตรียมข้อมูลเพื่อถ่ายโอนและถ่ายโอนไปยังขั้นตอนอื่นๆ ในระบบเพื่อประมวลผลได้ (ไม่เพียงแต่ถ่ายโอนสำหรับ End User)
3	ซอฟต์แวร์มีลักษณะของ Distributed processing และการถ่ายโอนข้อมูลเป็นแบบออนไลน์แต่เป็นในทิศทางเดียว
4	ซอฟต์แวร์มีลักษณะของ Distributed processing และการถ่ายโอนข้อมูลเป็นแบบออนไลน์แต่เป็นใน 2 ทิศทาง
5	ซอฟต์แวร์มีการจัดการการถ่ายโอนข้อมูลแบบไดนามิกขึ้นอยู่กับแต่ละส่วนของระบบ

F3. Performance จะมองในเรื่องของ Response Time ประสิทธิภาพของซอฟต์แวร์นั้นจะได้รับการประเมินโดย User ซึ่งเป็นผลสืบเนื่องจากการออกแบบพัฒนาการติดตั้งและรองรับซอฟต์แวร์นั้นๆ	
0	ไม่มีสภาพของประสิทธิภาพที่ User ต้องการเป็นพิเศษ
1	ประสิทธิภาพและการออกแบบตามความต้องการของ User ได้รับการทบทวนแต่ไม่มีความต้องการกระทำอะไรเป็นพิเศษ
2	Response Time และผลลัพธ์ของการทำงานเป็นเรื่องสำคัญและจำเป็นระหว่างช่วงเวลาที่มีการใช้งานสูงๆ การประมวลผลต้องให้เรียบร้อยก่อนวัดถัดไป
3	Response Time และผลลัพธ์ของการทำงานเป็นเรื่องที่สำคัญและจำเป็นระหว่างช่วงเวลาที่มีการใช้งานในธุรกิจ ไม่มีการออกแบบสำหรับ CPU ที่ต้องการใช้งานเฉพาะที่มีการประมวลผลที่มีความต้องการเฉพาะ รวมทั้งการออกแบบอินเตอร์เฟสมีการระบุเฉพาะ
4	เพิ่มในส่วนของความต้องการของ User ในเรื่องของประสิทธิภาพตั้งแต่ขั้นตอนของการวิเคราะห์ระบบงาน
5	เพิ่มในส่วนของการนำเครื่องมือในการวิเคราะห์ประสิทธิภาพเข้ามาใช้ในขั้นตอนการออกแบบ พัฒนา และนำไปใช้เพื่อที่จะให้ตรงกับประสิทธิภาพที่ User มี

F4. Heavily Used Configuration จะมองในเรื่องของการใช้ที่เกี่ยวกับฮาร์ดแวร์ และ Platform ซึ่งเป็นคุณลักษณะหนึ่งของซอฟต์แวร์ เช่น User ต้องการรันซอฟต์แวร์บนอุปกรณ์หรือเครื่องที่มีการใช้งานอย่างหนัก	
0	ไม่มีความชัดเจนในเรื่องของความเข้มงวดในความสามารถในการจัดการให้สามารถใช้งานได้
1	มีความเข้มงวดในการจัดการให้ใช้งานได้แต่ความเข้มงวดก็ยังน้อยในซอฟต์แวร์ซึ่งเป็นตัวอย่าง
2	มีระบบรักษาความปลอดภัยหรือการคำนึงถึงข้อจำกัดด้านเวลารวมอยู่ด้วย
3	มีการเจาะจงถึง Processor (เช่น Centrino P4) สำหรับซอฟต์แวร์
4	มีการระบุถึงความเข้มงวดอย่างยิ่งในการประมวลผลซอฟต์แวร์ต่อหน่วยประมวลผลกลาง (Central Processor) ซึ่งมีลักษณะที่เฉพาะเจาะจง
5	มีข้อจำกัดพิเศษบนซอฟต์แวร์ในส่วนประกอบแบบ Distributed ของระบบ

F5. Transaction Rate จะมองในเรื่องความถี่ในการประมวลผล Transaction ว่าเป็นแบบรายวัน รายสัปดาห์ รายเดือน	
0	ไม่มีการคาดถึงช่วงที่มีระดับสูงสุดของ Transaction
1	ช่วงที่ระดับของจำนวน Transaction เพิ่มสูงสุดนั้นได้ถูกคาดคะเน (รายเดือน รายปี)
2	มีการคาดคะเนว่าจำนวน Transaction สูงสุดเป็นรายสัปดาห์
3	มีการคาดคะเนว่าจำนวน Transaction สูงสุดเป็นรายวัน
4	มีอัตราของ Transaction สูงมาก Requirement ต้องการมากเพียงพอเพื่อที่จะมีการวิเคราะห์ประสิทธิภาพของงานในขั้นตอนการออกแบบ
5	มีอัตราของ Transaction สูงมาก Requirement ต้องมากเพียงพอเพื่อที่จะมีการวิเคราะห์ประสิทธิภาพของงานในขั้นตอนการออกแบบ รวมไปถึงขั้นตอนของการพัฒนาและการติดตั้ง

F6. On-Line Data Entry จะมองในเรื่องของเปอร์เซ็นต์ของข้อมูลที่ผ่านมาทางออนไลน์	
0	ทุกๆ Transaction จะเป็นรูปแบบของ Batch
1	1% ถึง 7% ของ Transaction เป็น Interactive Data Entry
2	8% ถึง 15% ของ Transaction เป็น Interactive Data Entry
3	16% ถึง 23% ของ Transaction เป็น Interactive Data Entry
4	24% ถึง 30% ของ Transaction เป็น Interactive Data Entry
5	> 30% ของ Transaction เป็น Interactive Data Entry

F7. End-User Efficiency จะมองในเรื่องของผู้ใช้งานระบบว่าสามารถใช้งานระบบได้อย่างมีประสิทธิภาพหรือไม่ ซึ่งลักษณะของปัจจัยที่ส่งผลต่อ User Friendly การให้ความสำคัญกับประสิทธิภาพประกอบด้วย

- Navigation Aids
- Menu
- Online Help and Document
- Automated Cursor Movement
- Scrolling
- Remote Printing (Via Online Transaction)
- Reassigned Function Keys
- Batched Job Submitted From Online Transactions
- Cursor Selection Of Screen Data

Chulalongkorn University

- Heavy Use Of Reverse Video Highlighting Color Underling And Other Indicators - Hard Copy User Documentation Of Online Transactions - Mouse Interface - Pop Up Windows - As Few Screen As Possible To Accomplish A Business Function - Bilingual Support - Multilingual Support	
0	ไม่มีการให้ความสำคัญกับประสิทธิภาพตามข้อกำหนดข้างต้น
1	มีการให้ความสำคัญกับประสิทธิภาพ 1-3 ข้อจากข้อกำหนดข้างต้น
2	มีการให้ความสำคัญกับประสิทธิภาพ 4-5 ข้อจากข้อกำหนดข้างต้น
3	มากกว่า 6 ข้อจากข้อกำหนดข้างต้นโดย User ไม่มีความต้องการที่เฉพาะในเรื่อง Efficiency
4	มากกว่า 6 ข้อจากข้อกำหนดข้างต้นโดย User มีการใช้ความสนใจกับเรื่องของ Efficiency โดยต้องมีการออกแบบซอฟต์แวร์อย่างเฉพาะเพื่อสนับสนุน Efficiency
5	มากกว่า 6 ข้อจากข้อกำหนดข้างต้นโดย User มีการใช้ความสนใจกับเรื่องของ Efficiency โดยต้องมีการใช้งานอุปกรณ์และขั้นตอนที่พิเศษซึ่งจะทำให้การทำงานนั้นบรรลุตามวัตถุประสงค์

F8. Online Update จะมองในเรื่องของ Internal Logical File ว่ามีจำนวนมากน้อยเพียงใด ที่ได้รับการอัพเดทผ่าน Online Transaction	
0	ไม่มีการอัพเดทออนไลน์
1	การอัพเดทเป็นการอัพเดท 1-3 ไฟล์โดยการอัพเดทนั้นเป็นไปได้ช้า
2	การอัพเดทเป็นการอัพเดท 4 ไฟล์ขึ้นไป โดยการอัพเดทนั้นเป็นไปได้ช้า
3	สามารถออนไลน์อัพเดทได้ในส่วนหลักๆของ Internal Logical Files
4	เพิ่มเติมการป้องกันข้อมูลสูญหายในขณะทำ Online Update ซึ่งได้มีการออกแบบและสร้างอย่างพิเศษในซอฟต์แวร์
5	เพิ่มเติมการป้องกันข้อมูลสูญหายของข้อมูลรวมทั้งมีการจัดการระบบกู้ข้อมูลอัตโนมัติ

F9. Complex Processing เป็นส่วนของคุณลักษณะของซอฟต์แวร์ โดยส่วนประกอบมีดังนี้ - มีส่วนขยายของ Logical Processing - มีส่วนขยายของ Mathematical Processing - มีการประมวลผลที่ซับซ้อนเพื่อจัดการกับ Input และ Output เช่นการมี Multimedia - ข้อผิดพลาดที่เกิดจากการประมวลผลก่อให้เกิด Transaction ที่ไม่สมบูรณ์นั้นต้องทำการประมวลผลอีกครั้ง - Sensitive Control เช่นความปลอดภัยในการประมวลผลข้อมูล	
0	ไม่มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน
1	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน 1 ข้อ
2	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน 2 ข้อ
3	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน 3 ข้อ
4	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบน 4 ข้อ
5	มีคุณลักษณะของซอฟต์แวร์ตามลักษณะดังที่กล่าวได้ด้านบนครบ 5 ข้อ

F10. Reusability จะมองในเรื่องของซอฟต์แวร์ที่ได้ทำการพัฒนานั้นสามารถนำไปปรับใช้หรือสามารถนำไปพัฒนาต่อไปในโครงการอื่นๆ ได้หรือไม่	
0	ไม่มีการ Reuse โค้ด
1	การ Reuse มีเพียงภายในซอฟต์แวร์
2	มีการนำโค้ดไป Reuse น้อยกว่า 10% ในหลายๆ งาน
3	มีการนำโค้ดไป Reuse มากกว่า 10% ในหลายๆ งาน
4	ทั้งซอฟต์แวร์และเอกสารประกอบการพัฒนาค่อนข้างเอื้อต่อการนำไป Reuse
5	ทั้งแอพพลิเคชั่นและเอกสารประกอบการพัฒนาเอื้อต่อการนำไป Reuse

F11. Installation Ease จะมองในเรื่องของการติดตั้งซอฟต์แวร์	
0	ไม่มีลักษณะพิเศษในการติดตั้ง
1	มีลักษณะพิเศษในการติดตั้ง แต่ไม่มีความต้องการเพิ่มเติมจาก User
2	การติดตั้งมีการกำหนดจาก User มีคู่มือในการติดตั้งและปรับเปลี่ยนโดยคู่มือที่จัดทำนั้นต้องผ่านการตรวจสอบและจะไม่คำนึงถึงผลกระทบจากการติดตั้ง
3	การติดตั้งมีการกำหนดจาก User มีคู่มือในการติดตั้งและปรับเปลี่ยนโดยคู่มือที่จัดทำนั้นต้องผ่านการตรวจสอบและคำนึงถึงผลกระทบจากการติดตั้ง
4	เพิ่มจากระดับของปัจจัยที่ 2 โดยมีระบบติดตั้งและเปลี่ยนแปลงอัตโนมัติ
5	เพิ่มจากระดับของปัจจัยที่ 3 โดยมีระบบติดตั้งและเปลี่ยนแปลงอัตโนมัติ

Chulalongkorn University

F12. Operation Ease จะมองในเรื่องของประสิทธิภาพหรือระบบอัตโนมัติในการ Start Up Back Up และขั้นตอนการ Recovery และมีการทดสอบอย่างมีประสิทธิภาพ โดยต้องมีคู่มือประกอบซอฟต์แวร์ที่ได้จัดทำขึ้น (Manual)	
0	ไม่มีลักษณะพิเศษในการจัดทำ Start Up Back Up และ Recovery
1	มีลักษณะพิเศษที่มีประสิทธิภาพในการจัดทำ Start Up Back Up และ Recovery
2	
3	
4	
5	ไม่ต้องมีผู้จัดการซอฟต์แวร์ โดยซอฟต์แวร์นั้นสามารถจัดการเองได้ รวมทั้งสามารถย้อนเพื่อแก้ไขข้อผิดพลาด

F13. Multiple Sites จะมองในเรื่องของซอฟต์แวร์ว่าสามารถพัฒนาและสนับสนุนการติดตั้ง ใช้งานได้หลายพื้นที่ หรือหลายองค์กร	
0	User ไม่มีความต้องการพิเศษในการทำ Multiple Site
1	ความจำเป็นที่จะต้องมี Multiple Site หรือไม่นั้นต้องมีการพิจารณา ซึ่งถ้าเป็นแบบ Multiple Site แล้วนั้น ซอฟต์แวร์จะถูกพัฒนาภายใต้ลักษณะของ ฮาร์ดแวร์และซอฟต์แวร์เดียวกัน
2	ความจำเป็นที่จะต้องมี Multiple Site หรือไม่นั้นต้องมีการพิจารณา ซึ่งถ้าเป็นแบบ Multiple Site แล้วนั้น ซอฟต์แวร์จะถูกพัฒนาภายใต้ลักษณะของ ฮาร์ดแวร์และซอฟต์แวร์ที่เหมือนกัน
3	ความจำเป็นที่จะต้องมี Multiple Site หรือไม่นั้นต้องมีการพิจารณา ซึ่งถ้าเป็นแบบ Multiple Site แล้วนั้น ซอฟต์แวร์จะถูกพัฒนาภายใต้ลักษณะของ ฮาร์ดแวร์และซอฟต์แวร์ที่แตกต่างกัน
4	เอกสารประกอบได้มีการจัดการทดสอบและสนับสนุนซอฟต์แวร์แบบ Multiple Sites ภายใต้ระดับของปัจจัยที่ 1 หรือ 2
5	เอกสารประกอบได้มีการจัดการทดสอบและสนับสนุนซอฟต์แวร์แบบ Multiple Sites ภายใต้ระดับของปัจจัยที่ 3

F14. Facilitate change จะมองในเรื่องของซอฟต์แวร์ว่าสามารถออกแบบและพัฒนาเพื่อสามารถเปลี่ยนซอฟต์แวร์ได้ตามความสะดวกหรือไม่ - การ Query ข้อมูลที่ยืดหยุ่นและสิ่งที่ช่วยออกแบบรายงานให้ง่ายขึ้น โดยสามารถดึงข้อมูลอย่างง่ายๆ ที่ต้องการได้ เช่น ตรรกะแบบ And/Or ซึ่งใช้งานกับ 1 Internal Logical File - การ Query ข้อมูลที่ยืดหยุ่นและสิ่งที่ช่วยออกแบบรายงานให้ง่ายขึ้น โดยสามารถดึงข้อมูลที่ซับซ้อนที่ต้องการได้ เช่น ตรรกะแบบ And/Or ซึ่งใช้งานมากกว่า 1 Internal Logical File - การ Query ข้อมูลที่ยืดหยุ่นและสิ่งที่ช่วยออกแบบรายงานให้ง่ายขึ้น โดยสามารถดึงข้อมูลที่ซับซ้อนได้ เช่น ตรรกะแบบ And/Or ซึ่งใช้งานได้มากกว่า 1 Internal Logical File - ข้อมูลที่ใช้ในการควบคุมทางธุรกิจนั้นจะถูกเก็บไว้ในตารางโดย User จะทำการ Maintain โดยการ Online Interactive Processes การเปลี่ยนแปลงใดๆ ที่เกิดขึ้นนั้นจะกระทบกับวันที่มาทำธุรกิจในวันถัดไปได้ - ข้อมูลที่ใช้ในการควบคุมทางธุรกิจนั้นจะถูกเก็บไว้ในตารางโดย User จะทำการ Maintain โดยการ Online Interactive Processes การเปลี่ยนแปลงใดๆ ที่เกิดขึ้นจะส่งผลทันทีทันใด	
0	ไม่มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน
1	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 1 ข้อ
2	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 2 ข้อ
3	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 3 ข้อ
4	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 4 ข้อ
5	มีคุณลักษณะของ Facilitate ตามที่กล่าวมาด้านบน 5 ข้อ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

*** ในการกรอกข้อมูลปัจจัย T1-T13 กรอกค่า 0 หมายถึงปัจจัยนั้นไม่มีอิทธิพล กรอกค่า 5 หมายถึงปัจจัยนั้นมีอิทธิพลสูง**

T1. Distributed System แอพพลิเคชั่นมีฐานข้อมูลแบบรวมศูนย์(Centralized) หรือแบบกระจาย (Distributed)	
ให้ระดับอิทธิพลสูง	หมายถึงมีการใช้สถาปัตยกรรมที่มีความซับซ้อน

T2. Response adjectives ความรวดเร็วในการตอบสนองเป็นปัจจัยสำคัญสำหรับผู้ใช้งาน	
ให้ระดับอิทธิพลสูง	หมายถึงจำเป็นต้องให้ความสำคัญกับเวลาในการตอบสนอง

T3. End-user efficiency แอพพลิเคชั่นถูกพัฒนาขึ้นเพื่อเพิ่มความสามารถของผู้ใช้งาน	
ให้ระดับอิทธิพลสูง	หมายถึงแอพพลิเคชั่นให้ความสำคัญกับการเพิ่มประสิทธิภาพให้แก่ผู้ใช้

T4. Complex processing ความซับซ้อนของอัลกอริทึ่มในการพัฒนา	
ให้ระดับอิทธิพลสูง	หมายถึงแอพพลิเคชั่นมีอัลกอริทึ่มที่ซับซ้อน

T5. Reusable code ปริมาณการรียูสโค้ด	
ให้ระดับอิทธิพลสูง	หมายถึงไม่มีการรียูสโค้ด

T6. Easy to install ความง่ายในการติดตั้ง	
ให้ระดับอิทธิพลสูง	หมายถึงต้องมีขั้นตอนการติดตั้งที่ง่าย

T7. Easy to use ความง่ายในการใช้งาน	
ให้ระดับอิทธิพลสูง	หมายถึงต้องออกแบบให้มีการใช้งานง่าย

T8. Portable มีการออกแบบให้ใช้งานได้หลายแพลตฟอร์ม	
ให้ระดับอิทธิพลสูง	หมายถึงสามารถใช้งานได้หลายแพลตฟอร์ม

T9. Easy to change คำนึงถึงการปรับค่าหรือเปลี่ยนแปลงของแอพพลิเคชั่นในอนาคต	
ให้ระดับอิทธิพลสูง	หมายถึงการพัฒนาต้องรองรับการเปลี่ยนแปลงของความต้องการในอนาคต

T10. Concurrent ปัญหา Database Locking หรือในการเข้าใช้งานพร้อมๆกัน	
ให้ระดับอิทธิพลสูง	หมายถึงต้องให้ความสำคัญกับปัญหานี้

T11. Security features สามารถใช้งานโซลูชั่นเกี่ยวกับความปลอดภัยที่มีอยู่หรือจะต้องพัฒนาเพิ่มเติม	
ให้ระดับอิทธิพลสูง	หมายถึงจำเป็นต้องพัฒนาเพิ่มเติม

Chulalongkorn University

T12. Access for third-parties มีการใช้งาน Third-Part API หรือ Library หรือไม่	
ให้ระดับอิทธิพลสูง	หมายถึงไม่มีการใช้งาน Third-Part API หรือ Library

T13. Special training required แอพพลิเคชั่นมีการใช้งานที่ซับซ้อนหรือไม่	
ให้ระดับอิทธิพลสูง	หมายถึงผู้ใช้งานจำเป็นจะต้องใช้เวลามากในการทำความเข้าใจการใช้งาน

E1.Familiar with development method used เป็นการวัดประสบการณ์ของทีมงานเกี่ยวกับกระบวนการพัฒนาที่ใช้ในโครงการ	
0	ทีมงานไม่รู้จักกระบวนการพัฒนาที่ใช้ในโครงการนี้
1	ทีมงานมีความรู้ในเชิงทฤษฎีเกี่ยวกับกระบวนการพัฒนาที่ใช้ในโครงการนี้
2	สมาชิกในทีมอย่างน้อยหนึ่งคนเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้เพียงครั้งเดียวหรือไม่กี่ครั้ง
3	สมาชิกของทีมอย่างน้อยครึ่งหนึ่งเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้ในหลายโครงการ
4	สมาชิกของทีมอย่างน้อยหนึ่งคนเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้ในหลายโครงการ
5	ทีมงานทั้งหมดมีเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้ในหลายโครงการ

E2. Application experience เป็นการวัดประสบการณ์ของทีมงานเกี่ยวกับประเภทของโมบายแอพพลิเคชั่นที่กำลังจะพัฒนา หรือประสบการณ์ในการพัฒนาแอพพลิเคชั่นที่หลากหลายประเภท	
0	ทุกคนในทีมเป็นมือใหม่
1 หรือ 2	บางคนในทีมมีประสบการณ์เล็กน้อย 1 ปีถึง 1 ปี ครึ่ง สมาชิกที่เหลือเป็นมือใหม่
3	ทุกคนในทีมมีประสบการณ์มากกว่า 1 ปีครึ่ง
4	สมาชิกในทีมส่วนใหญ่มีประสบการณ์ 2 ปี
5	สมาชิกทุกคนมีประสบการณ์ 2 ปี

E3.Object-Oriented experience เป็นการวัดประสบการณ์ของทีมงานเกี่ยวกับการวิเคราะห์และออกแบบเชิงวัตถุ เช่นการสร้างแบบจำลองยูสเคส(Use Case modeling)ในขั้นตอนการวิเคราะห์ และการสร้างแบบจำลองคอมโพเน้นท์(Component modeling)ในขั้นตอนการออกแบบ	
0	ทีมงานไม่มีความคุ้นเคยกับการวิเคราะห์และออกแบบเชิงวัตถุเลย
1	ทุกคนในทีมมีประสบการณ์น้อยกว่า 1 ปี
2 หรือ 3	ทุกคนในทีมมีประสบการณ์ 1 ปี ถึง 1 ปี ครึ่ง
4	สมาชิกส่วนใหญ่ในทีมมีประสบการณ์มากกว่า 2 ปี
5	สมาชิกของทีมอย่างน้อยหนึ่งคนเคยใช้กระบวนการพัฒนาเดียวกันกับโครงการนี้ในหลายโครงการ

E4. Lead analyst capability เป็นการวัดประสบการณ์ในการวิเคราะห์ความต้องการและสร้างแบบจำลอง	
0	นักวิเคราะห์เป็นมือใหม่
1 หรือ 2	นักวิเคราะห์มีประสบการณ์ไม่กี่โครงการ
3 หรือ 4	นักวิเคราะห์มีประสบการณ์อย่างน้อย 2 ปี จากหลายโครงการ
5	นักวิเคราะห์มีประสบการณ์อย่างน้อย 3 ปี กับรูปแบบโครงการที่หลากหลาย

E5. Motivation ปัจจัยนี้เป็นการอธิบายถึงแรงจูงใจของทีมงานโดยรวม	
0	ไม่มีแรงจูงใจ
1 หรือ 2	มีแรงจูงใจเล็กน้อย
3 หรือ 4	ทีมงานมีแรงจูงใจในการดำเนินงานที่ดี
5	ทีมงานมีแรงจูงใจและแรงกระตุ้นอย่างมาก

E6. Stable requirements ปัจจัยนี้เป็นการวัดระดับของการเปลี่ยนแปลงความต้องการและความไม่ชัดเจนของความต้องการ	
0	ความต้องการไม่แน่นอนและมีการเปลี่ยนแปลงความต้องการอย่างต่อเนื่อง
1 หรือ 2	ความต้องการไม่แน่นอน ลูกค้ามีการเปลี่ยนแปลงความต้องการในหลายช่วงของโครงการ
3 หรือ 4	มีการเปลี่ยนแปลงความต้องการเล็กน้อยเท่าที่จำเป็น
5	ความต้องการมีเสถียรภาพตลอดช่วงเวลาของโครงการ

E7. Part-time workers มีการสอนงานสมาชิกใหม่ของทีมหรือหมายถึงสมาชิกของทีมไม่แน่นอนซึ่งส่งผลต่อความสามารถในการพัฒนาซอฟต์แวร์(Productivity)ในทางลบ	
0	ไม่มีสมาชิกในทีมที่ทำงานแบบพาร์ทไทม์(Part-Time)
1 หรือ 2	บางส่วนของทีม (ประมาณ 20 เปอร์เซ็น) เป็นพนักงานแบบพาร์ทไทม์(Part-time)
3 หรือ 4	ครึ่งหนึ่งของทีมงานเป็นพนักงานแบบพาร์ทไทม์(Part-time)
5	ทุกคนในทีมเป็นพนักงานแบบพาร์ทไทม์(Part-time)

E8. Difficult programming language ปัจจัยนี้บ่งบอกถึงความเชี่ยวชาญของทีมงานในการใช้เครื่องมือและภาษาของการพัฒนาที่ใช้ในโครงการนี้	
0	ทุกคนในทีมมีประสบการณ์ในการเขียนโปรแกรมมากกว่า 2 ปี
1	สมาชิกทีมส่วนใหญ่มีประสบการณ์ในการเขียนโปรแกรมมากกว่า 2 ปี
2	ทุกคนในทีมมีประสบการณ์ในการเขียนโปรแกรมมากกว่า 1 ปีครึ่ง
3	สมาชิกในทีมส่วนใหญ่มีประสบการณ์ในการเขียนโปรแกรมมากกว่า 1 ปี
4	สมาชิกในทีมบางคนมีประสบการณ์ในการเขียนโปรแกรมประมาณ 1 ปี ที่เหลือเป็นมือใหม่
5	ทุกคนในทีมเป็นมือใหม่

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก ข

ตัวอย่างการประมาณขนาดโมบายแอพพลิเคชั่นของโมบายแอพพลิเคชั่นตัวอย่าง

โมบายแอพพลิเคชั่นตัวอย่างมีข้อมูลตามเอกสารแบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่นดังนี้

แบบฟอร์มการเก็บข้อมูลโมบายแอพพลิเคชั่น

แบบฟอร์มส่วนที่ 1

Application Name : แอพพลิเคชั่นตัวอย่าง

Detail : แอพพลิเคชั่นนี้มีประโยชน์สำหรับผู้ใช้งานที่ต้องการอ่านข้อมูลต่างๆ ที่ถูกเก็บไว้ในรูปแบบคิวอาร์โค้ดไม่ว่าจะเป็น ข้อมูล URL ของเว็บไซต์ ข้อความ, เบอร์โทรศัพท์ โดยการใช้งานให้นำโทรศัพท์ไปสแกนรูปภาพคิวอาร์โค้ด จากนั้นแอพพลิเคชั่นก็จะแสดงข้อมูลที่ถูกเก็บไว้ในคิวอาร์โค้ดให้กับผู้ใช้งาน และผู้ใช้ยังสามารถเรียกดูข้อมูลประวัติการแสกนคิวอาร์ โค้ด ได้

Owner: นักพัฒนาตัวอย่าง Date: 20/1/2014

No. of Function: 2 No. of Screen: 4

Accual Effort: 24 Man - Hour

Data File Name	Type	No. of Data Field	No.of Subgroup
History	Internal Data File	3	0

แอพพลิเคชั่นนี้มีทั้งหมด 2 ฟังก์ชั่นการทำงาน คือ

- สแกนคิวอาร์โค้ดด้วยกล้อง
- ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ฟังก์ชั่นสแกนคิวอาร์โค้ดด้วยกล้อง

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
1	สแกนคิวอาร์โค้ดด้วยกล้อง	หน้าจอหลัก	1	User	Tap	Button	Start Scan	From	Screen					
		หน้าจอกล้อง	2	User	Scan	Data	QR Code	into	Screen		M	QR Code Reader	B	
			3	App	Other: Generate	Data	QR Code				M	QR Code Reader	B	
			4	App	Add	Data	Detail, Type	to	Internal Data File	History	M	Core data	D	
			5	App	Show	Data	Detail	From	Internal Data File	History	M	Core data	D	
		หน้าจอผลการสแกน	6	User	Tap	Data	Detail	From	Screen					
			7	App	Send	Data	Detail	to	External App	Web browser				

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ฟังก์ชั่นดูข้อมูลคิวอาร์โค้ดจากไฟล์ History

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ History	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แบบฟอร์มส่วนที่ 3

คำชี้แจง ระบุถึงปัจจัยที่ต้องคำนึงถึงและส่งผลต่อความซับซ้อนในการพัฒนา
โดยที่ผู้ให้ข้อมูลจะต้องนึกย้อนไปถึงอิทธิพลของปัจจัยต่างๆในช่วงของการออกแบบระบบ
ระดับอิทธิพลของปัจจัยที่เป็นไปได้ คือ 0 ถึง 5 โดยให้กรอกตามคำแนะนำในเอกสารที่แนบมาด้วย

ลำดับ	ปัจจัย	ระดับอิทธิพล	ลำดับ	ปัจจัย	ระดับอิทธิพล
F1	Data Communication	0	T5	Reusable code	0
F2	Distribution Data Processing	0	T6	Easy to install	0
F3	Performance	1	T7	Easy to use	0
F4	Heavily Used Configuration	0	T8	Portable	0
F5	Transaction Rate	0	T9	Easy to change	1
F6	On-line Data Entry	5	T10	Concurrent	0
F7	End-User Efficiency	1	T11	Security features	0
F8	Online Update	0	T12	Access for third-parties	0
F9	Complex Processing	2	T13	Special training required	0
F10	Reusability	0	E1	Familiar with development method used	3
F11	Installation Ease	0	E2	Application experience	3
F12	Operational Ease	0	E3	Object-Oriented experience	3
F13	Multiple Sites	0	E4	Lead analyst capability	2
F14	Facilitate change	1	E5	Motivation	3
T1	Distributed System	0	E6	Stable Requirements	5
T2	Response adjectives	1	E7	Part-time workers	0
T3	End-user efficiency	1	E8	Difficulty of programming language	1
T4	Complex processing	2			

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แอพพลิเคชั่นนี้มีทั้งหมด 4 หน้าจอ คือ

- หน้าจอหลัก
- หน้าจอกล้อง
- หน้าจอ History
- หน้าจอผลการสแกน

หน้าจอหลัก (ซ้าย) และหน้าจอกล้อง(ขวา)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้าผลการสแกน (ซ้าย) และหน้าจอ History (ขวา)

การประมาณขนาดด้วยวิธีฟังก์ชั่นพอยต์

ระบุ ILF และ ELF ด้วยข้อมูลจากแบบฟอร์มส่วนที่ 1

Data File Name	Type	No. of Data Field	No.of Subgroup
History	Internal Data File	3	0

จากข้อมูลช่อง Type จึงระบุ ไฟล์นี้เป็น ILF

นับ DET และ RET

จากข้อมูลช่อง No.of Data Field และ No.of Subgroup ไฟล์นี้มี 3 DET 1 RET

ระบุ EI EO และ EQ ด้วยข้อมูลจากแบบฟอร์มส่วนที่ 2

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
1	สแกนคิวอาร์โค้ดด้วยกล้อง	หน้าจอหลัก	1	User	Tap	Button	Start Scan	From	Screen					
		หน้าจอกล้อง	2	User	Scan	Data	QR Code	into	Screen		M	QR Code Reader	B	
			3	App	Other: Generate	Data	QR Code				M	QR Code Reader	B	
			4	App	Add	Data	Detail, Type	to	Internal Data File	History	M	Core data	D	
			5	App	Show	Data	Detail	From	Internal Data File	History	M	Core data	D	
		หน้าจอผลการสแกน	6	User	Tap	Data	Detail	From	Screen					
			7	App	Send	Data	Detail	to	External App	Web browser				

ฟังก์ชั่นที่ 1 มีกระบวนการพื้นฐาน 3 กระบวนการพื้นฐานคือ

- ขั้นตอนการบันทึกข้อมูล QR Code ที่สแกนผ่านกล้องไปยังไฟล์ History ซึ่งมีขั้นตอนการทำงานหลัก คือ ลำดับขั้นตอนที่ 4
    - กระบวนการนี้จะถูกนับเป็น “EI” เนื่องจากเป็นการเพิ่มข้อมูลไปยังไฟล์ History ซึ่งเป็น ILF
    - นับ 3 DET จาก 2 ข้อมูล (Detail, Type) + 1 ขั้นตอนเพื่อเริ่มกระบวนการ EI (ขั้นตอนที่ 2)
    - นับ 1 FTR จาก 1 ไฟล์ข้อมูล (History)
- ขั้นตอนการแสดงข้อมูลของ QR Code ที่บันทึกไปยังไฟล์ History แล้ว ให้แก่ผู้ใช้งาน ซึ่งมีขั้นตอนการทำงานหลัก คือ ลำดับขั้นตอนที่ 5
    - กระบวนการนี้จะถูกนับเป็น “EQ” เนื่องจากเป็นการแสดงข้อมูลจากไฟล์ History ซึ่งเป็น ILF แก่ผู้ใช้งานโดยไม่มีการประมวลผล
    - นับ 1 DET จาก 1 ข้อมูล (Detail)
    - นับ 1 FTR จาก 1 ไฟล์ข้อมูล (History)
- ขั้นตอนการส่งค่า Detail ของ QR Code ที่เลือกออกไปยัง Web Browser เพื่อแสดงข้อมูล ซึ่งมีขั้นตอนการทำงานหลัก คือ ลำดับขั้นตอนที่ 7
    - กระบวนการนี้จะถูกนับเป็น “EO” เนื่องจากเป็นการส่งค่าข้อมูลจากไฟล์ History ซึ่งเป็น ILF แก่ไปยังระบบภายนอกโดยต้องประมวลผลข้อมูลส่งไปในลักษณะ URL
    - นับ 2 DET จาก 1 ข้อมูล (Detail) + 1 ขั้นตอนขั้นตอนเพื่อเริ่มกระบวนการ EO (ขั้นตอนที่ 6)
    - นับ 1 FTR จาก 1 ไฟล์ข้อมูล (History)

Seq	Function Name	Screen Name	Step Seq.	Subject	verb	Target		Direction			Use			
						Type	Detail	Type1	Type2	Name	Type	Name	Module Type	Optional
2	ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History		1	User	Select	item	History	From	Tab Menu					
		หน้าจอ History	2	App	Show	Data	Detail, Type, Date	From	Internal Data File	History	M	Core data,Table View	D,D	
			3	User	Select	Data	Detail, Type, Date	From	Table		M	Core data,Table View	D,D	
			4	App	Send	Data	Detail	to	External App	Web browser				

ฟังก์ชั่นที่ 2 มีกระบวนการพื้นฐาน 2 กระบวนการพื้นฐานคือ

- ขั้นตอนการแสดงข้อมูล Detail, Type, Date จากไฟล์ History ซึ่งมีขั้นตอนการทำงานหลัก คือ ลำดับขั้นตอนที่ 2
    - กระบวนการนี้จะถูกนับเป็น EQ เนื่องจากเป็นการแสดงข้อมูลจากไฟล์ History ซึ่งเป็น ILF
    - นับ 4 DET จาก 3 ข้อมูล (Detail, Type, Date) + 1 ขั้นตอนขั้นตอนเพื่อเริ่มกระบวนการ EQ (ขั้นตอนที่ 1)
    - นับ 1 FTR จาก 1 ไฟล์ข้อมูล (History)
- ขั้นตอนการส่งค่า Detail ของ QR Code ที่เลือกออกไปยัง Web Browser เพื่อแสดงข้อมูล ซึ่งมีขั้นตอนการทำงานหลัก คือ ลำดับขั้นตอนที่ 4
    - กระบวนการนี้จะไม่ถูกนับเป็นฟังก์ชั่นเชิงรายการเนื่องจากมีตรรกะการทำงานและไฟล์ข้อมูลที่ใช้ซ้ำซ้อนกับกระบวนการอื่น

หลังจากนั้นกรอกข้อมูลลงแบบฟอร์มการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

เอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีฟังก์ชั่นพอยต์

Application Name : แอพพลิเคชั่นตัวอย่าง

ประเภท	ระดับความซับซ้อน						รวม
	ต่ำ		ปานกลาง		สูง		
EI	1	x3		x4		x6	3
EO	1	x4		x5		x7	4
EQ	2	x3		x4		x6	6
ILFs	1	x7		x10		x15	7
ELFs		x5		x7		x10	

Total Unadjusted Function Points (TUFP): 20

ลำดับ	ปัจจัย	คะแนน	ลำดับ	ปัจจัย	คะแนน
F1	Data Communication	0	F8	Online Update	0
F2	Distribution Data Processing	0	F9	Complex Processing	2
F3	Performance	1	F10	Reusability	0
F4	Heavily Used Configuration	0	F11	Installation Ease	0
F5	Transaction Rate	0	F12	Operational Ease	0
F6	On-line Data Entry	5	F13	Multiple Sites	0
F7	End-User Efficiency	1	F14	Facilitate change	1

System Complexity: 10

Adjusted System Complexity = 0.65 + (0.01 * ___10___)

Total Adjusted Function Points (TAFP): ___0.75___ * ____20____ = 15

ประมาณค่าความพยายามจากสมการแปลงค่า

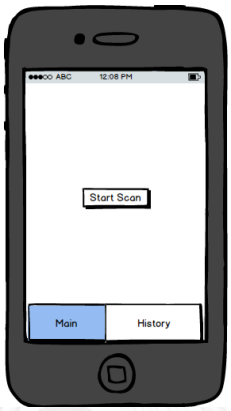
Effort (Man-Hour) = 15 * 2.2 = 33

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

การประมาณขนาดด้วยวิธีอ็อบเจกต์พอยต์

1. นับจำนวนและระบุความซับซ้อนของหน้าจอแต่ละหน้าจอ

หน้าจอหลัก



ประกอบด้วย 1 ส่วนหน้าจอ

และไม่มีการใช้แหล่งข้อมูล

หน้าจอกล้อง

ประกอบด้วย 1 ส่วนหน้าจอ

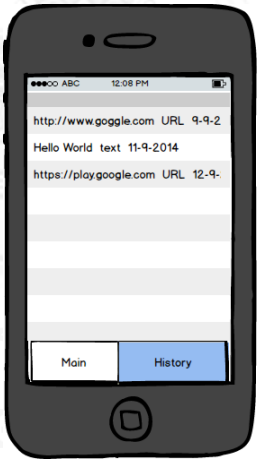
และไม่มีการใช้แหล่งข้อมูล

หน้าจอแสดงผลการสแกน

ประกอบด้วย 1 ส่วนหน้าจอ

มีการใช้แหล่งข้อมูลจากไฟล์ History

หน้าจอ History



ประกอบด้วย 1 ส่วนหน้าจอ

มีการใช้แหล่งข้อมูลจากไฟล์ History

2. นับจำนวนและระบุความซับซ้อนของหน้าจอแต่ละหน้าจอ

   แอพพลิเคชั่นนี้ไม่มีรายงาน

3. นับโมดูล

   มีทั้งหมด 3 โมดูล คือ Core Data, QR Code Reader, Table View

4. กรอกข้อมูลลงบนเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีอ็อบเจกต์พอยต์

เอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีอ๊อบเจกต์พอยต์

Application Name : แอพพลิเคชั่นตัวอย่าง

ประเภท	ระดับความซับซ้อน						รวม
	ต่ำ		ปานกลาง		สูง		
Screen	4	x1		x2		x3	4
Report		x2		x5		x8	
3GL Module					3	x10	30

Object Point: 34

% Reuse: 0

New Object Pont (NOP) = $\frac{(0)-100}{100} \times (_34_) =$ **34**

ประมาณค่าความพยายามด้วยการหารด้วยค่าความสามารถในการพัฒนาซอฟต์แวร์ ซึ่งพิจารณาจากปัจจัยความซับซ้อนทางสภาพแวดล้อม

Effort (Man-Month) = 34/25 = 1.36

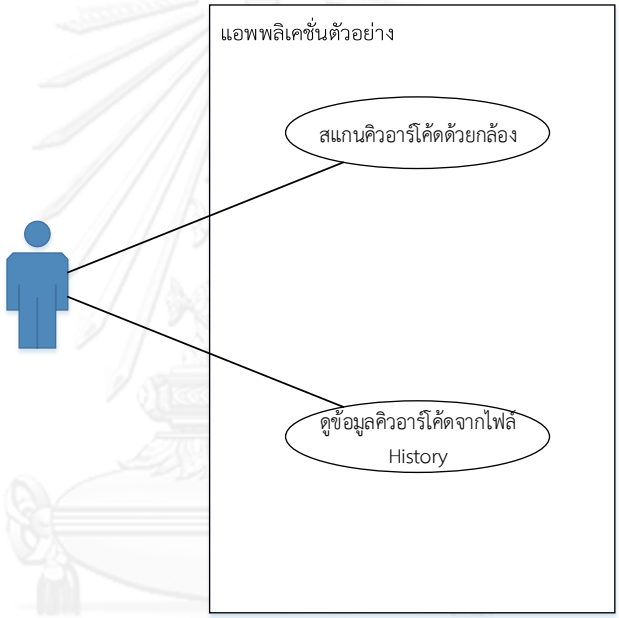
ทำการแปลงค่าให้เป็นหน่วย Man-Hour

Effort (Man-Hour) = 1.36 *(20*8) = 217.6

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

การประมาณขนาดด้วยวิธียูสเคสพอยต์

1. ระบุยูสเคส
   จากขั้นตอนการทำงานในแบบฟอร์มส่วนที่ 2 โมบายแอพพลิเคชั่นนี้มี 2 ยูสเคส
   - สแกนคิวอาร์โค้ดด้วยกล้อง
   - ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History
2. มีผู้กระทำการ 1 ผู้กระทำการคือ
   - User หรือผู้ใช้งาน
3. วาดแผนภาพยูสเคสได้ดังนี้



4. พิจารณาจำนวนรายการของแต่ละยูสเคสจากแบบฟอร์มส่วนที่ 2
   - สแกนคิวอาร์โค้ดด้วยกล้อง มีจำนวนรายการ 2 รายการ
   - ดูข้อมูลคิวอาร์โค้ดจากไฟล์ History มีจำนวนรายการ 2 รายการ
5. กรอกจำนวนยูสเคส และผู้กระทำการลงบนเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคส

CHULALONGKORN UNIVERSITY

เอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธียูสเคสพอยต์

Application Name: แอพพลิเคชั่นตัวอย่าง

Unadjusted Actor Weighting Table

Actor Type	Description	Weighting Factor	Number	Result
Simple	External System with well-defined API	1		
Average	External System using a protocol-based interface, e.g., HTTP, TCT/IP, or a database	2		
Complex	Human	3	1	3
			Unadjusted Actor Weight Total (UAW)	3

Unadjusted Use Case Weighting Table

Use Case Type	Description	Weighting Factor	Number	Result
Simple	1 – 3 transactions	5	2	10
Average	4 – 7 transactions	10		
Complex	> 7 transactions	15		
			Unadjusted Use Case Weight Total (UUCW)	10

Unadjusted Use Case Points (UUCP) = UAW + = _13_

Technical Complexity Factors

Factor Number	Description	Weight	Assigned Value	Weighted Value
T1	Distributed System	2	0	0
T2	Response adjectives	1	1	1
T3	End-user efficiency	1	1	1
T4	Complex processing	1	2	2
T5	Reusable code	1	0	0
T6	Easy to install	0.5	0	0
T7	Easy to use	0.5	0	0
T8	Portable	2	0	0
T9	Easy to change	1	1	1
T10	Concurrent	1	0	0
T11	Security features	1	0	0
T12	Access for third-parties	1	0	0
T13	Special training required	1	0	0
			Technical Factor Value (TFactor)	5

Technical Complexity Factor (TCF) = 0.6 + (0.01 * TFactor) = 0.6 +(0.01*_5_) = 0.65

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Environmental Factors

Factor Number	Description	Weight	Assigned Value	Weighted Value
E1	Familiar with development method used	1.5	3	4.5
E2	Application experience	0.5	3	1.5
E3	Object-Oriented experience	1	3	3
E4	Lead analyst capability	0.5	2	1
E5	Motivation	1	3	3
E6	Stable Requirements	2	5	10
E7	Part-time workers	-1	0	0
E8	Difficulty of programming language	-1	1	-1
			Environmental Factor Value (EFactor)	22

Environmental Factor (EF) = 1.4 + (-0.03 * EFactor) = 1.4+(-0.03*_22_) = 0.74

Adjusted Use Case Points (UCP) = UUCP * TCF * ECF = 13* 0.65 * 0.74 = 6.253

ประมาณค่าความพยายามโดยการนำค่ายูสเคสพอยต์ไปคูณกับค่าความสามารถในการพัฒนาซอฟต์แวร์ ซึ่งพิจารณาจากปัจจัยความซับซ้อนทางสภาพแวดล้อม

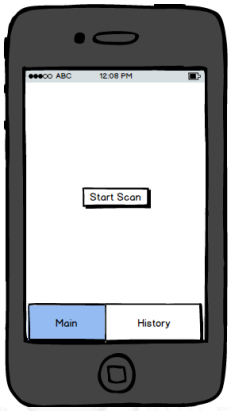
Effort (Man-Hour) = 6.253 * 20 = 125.06

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

การประมาณขนาดด้วยวิธีของ Nikunj Sakhrelia

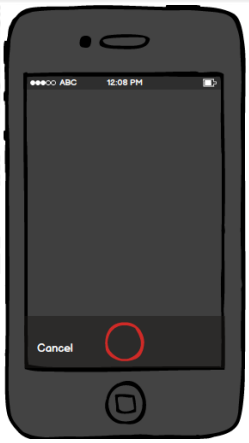
1. พิจารณาองค์ประกอบของหน้าจอและฟังก์ชั่นการทำงานบนหน้าจอแต่ละหน้าจอ

หน้าจอหลัก



- หน้าจอพร้อมปุ่มและไม่มีการใช้งานข้อมูลหรือฟังก์ชั่นใดๆ
- แท็บเมนูปรับแต่ง

หน้าจอกล้อง

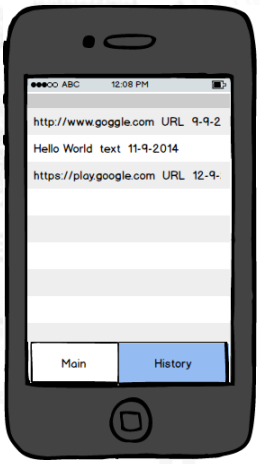


- การใช้งานกล้อง

หน้าจอแสดงผลการสแกน

- แสดงข้อมูลจากฐานข้อมูลภายใน

หน้าจอ History



- แสดงข้อมูลจากฐานข้อมูลภายใน
- แท็บเมนูปรับแต่ง

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2. เทียบองค์ประกอบกับตารางคำแนะนำของ Nikunj Sakhrelia

ระดับความซับซ้อน	รายละเอียดฟังก์ชั่นการทำงานและ องค์ประกอบบนหน้าจอของโมบายแอพพลิเคชั่น	ค่าน้ำหนัก (คน - ชั่วโมง)
ง่าย	การแสดงข้อมูลจากแหล่งข้อมูลภายใน หน้าจอแสดงผลพร้อมปุ่มหรือเมนูอย่างง่าย การใช้งานเอพีไอที่ไม่ต้องปรับแต่ง เข้าสู่ระบบ ลืมรหัสผ่าน การแสดงข้อมูลจากเว็บไซต์โดยใช้ Rssfeed การแสดงข้อมูลทวิตเตอร์(Twitter)ผ่าน Twitter feed การแสดงข้อมูลวิดีโอจากเว็บไซต์ยูทูป(Youtube) ผ่าน Youtube video feed แท็บเมนูที่มีการปรับแต่ง	4
ปานกลาง	มีการเรียกใช้เซอร์เวอร์เอพีไอ(Server API)	8
ซับซ้อน	ตารางแสดงข้อมูลแบบปรับแต่ง การใช้งานกล้อง การใช้งานเซนเซอร์ต่างๆของอุปกรณ์ บันทึกเสียง หรือวิดีโอ การใช้งานเอพีไอที่ต้องปรับแต่ง	16

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3. กรอกข้อมูลลงบนเอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

เอกสารการประมาณขนาดโมบายแอพพลิเคชั่นด้วยวิธีของ Nikunj Sakhrelia

Application Name : แอพพลิเคชั่นตัวอย่าง

หน้าจอ	จำนวนองค์ประกอบ			รวม
	ต่ำ(4)	ปานกลาง(8)	สูง(16)	
หน้าจอหลัก	2			8
หน้าจอกล้อง			1	16
หน้าจอผลการสแกน	1			4
หน้าจอ history	2			8
			Effort:	36

วิธีการนี้จะมีผลลัพธ์เป็นค่าความพยายามซึ่งเท่ากับ 36 คน-ชั่วโมง

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ประวัติผู้เขียนวิทยานิพนธ์

นายวชิรา สุขสวัสดิ์ เกิดวันที่ 29 มิถุนายน พ.ศ. 2531 สำเร็จการศึกษาวิทยาศาสตรบัณฑิต สาขาวิชาการพัฒนาซอฟต์แวร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปี พ.ศ. 2554 และได้เข้าศึกษาต่อชั้นบัณฑิตศึกษา วิทยาศาสตรมหาบัณฑิต สาขาการพัฒนาซอฟต์แวร์ด้านธุรกิจ ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย ปัจจุบันทำงานในตำแหน่งเจ้าหน้าที่ฝ่ายกำกับงานปฏิบัติการ ที่ธนาคารกรุงศรีอยุธยา จำกัด มหาชน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University