

## เครื่องมือช่วยประเมินค่าใช้จ่ายของการพัฒนาซอฟต์แวร์ โดยเทคนิคโคโคโม 2

### A Software Tool For Estimating Software Development Cost Using Cocomo II

ศรินทร์ วัชรบุศราคำ ธาราทิพย์ สุวรรณศาสตร์ และ วิวัฒน์ วัฒนาวุฒิ

**Sarin Watcharabusaracum, Taratip Suwannasart and Wiwat Vatanawood**

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University

#### บทคัดย่อ

การประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์โดยวิธีของโคโคโม2 มีวิธีการหาค่าประมาณการโดยให้ผู้ใช้เป็นผู้กำหนดค่าสเกล(Rating Scale)ให้กับตัวขับเคลื่อนค่าใช้จ่าย(Cost Driver) ซึ่งมีทั้งสิ้น 17 ตัว ดังนั้นการกำหนดค่าสเกลให้กับตัวขับเคลื่อนค่าใช้จ่ายจึงเป็นสิ่งที่สำคัญมากในการใช้วิธีการประมาณค่าใช้จ่ายด้วยวิธีการโคโคโม2 เนื่องจากการประมาณค่าใช้จ่ายด้วยวิธีการโคโคโม2 จะใช้ค่าสเกลจากทุกๆ ตัวขับเคลื่อนค่าใช้จ่ายเป็นข้อมูลพื้นฐานในการประมาณการ ตัวขับเคลื่อนค่าใช้จ่ายทั้ง 17 ตัวมีความซับซ้อนและยากที่จะกำหนดค่าสเกล ดังนั้นในงานวิจัยนี้จึงเสนอเครื่องมือที่ช่วยประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์โดยใช้เทคนิคโคโคโม 2 ซึ่งเมื่อนำเครื่องมือไปทดสอบพบว่าค่าประมาณการที่ได้จากการใช้เครื่องมือมีความใกล้เคียง กับค่าประมาณการที่ได้จากโคโคโม 2 ด้วยความเชื่อมั่น 95 %

#### ABSTRACT

In order to estimate software development cost using COCOMO II, estimator has to determine cost drivers' rating scale. In COCOMO II, there are 17 cost drivers. All cost drivers are the important key to estimate the software development cost. COCOMO II has a guideline for determining rating scale but for inexperienced users this task is still complex. This research purposes a tool for estimating the software development cost using COCOMO II. The estimated software development cost using the method in this research is similar to the estimated cost from original COCOMO II with 95% confidence.

#### คำนำ

การประมาณค่าใช้จ่ายในการพัฒนาซอฟต์แวร์ที่ถูกพัฒนาขึ้นมีหลายวิธีดังตัวอย่าง เช่น เอ็กpert จัสมัณฑ์ (Expert Judgement) พากินสัน (Parkinson) ไพรส์ทูวิน (Price-to-win) อาณาโลจี (Analogy) ท็อปดาวน (Top-Down) และบ๊อตทอมอัป (Bottom-up) เป็นต้น ซึ่งวิธีการประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์โดยวิธีของโคโคโม(Constructive Cost Model) เป็นวิธีที่ได้รับการยอมรับอย่างกว้างขวาง เนื่องจากทีมผู้พัฒนาโคโคโมมีการติดตามผลการใช้งานและทำการปรับปรุงมาเป็นเวลากว่า 10 ปี และในปัจจุบันได้พัฒนามาเป็นโคโคโม2(COCOMO II) ซึ่งประกอบด้วย โมเดลย่อย 3 โมเดล คือ แอปพลิเคชันคอมโพสิชันโมเดล เออร์ดีไซ์โมเดล และ โปสคอมโพสิชันโมเดล แต่ยังคงวิธีการหาค่าประมาณการ

โดยให้ผู้ใช้เป็นผู้กำหนดค่าสเกล(Rating Scale)ให้กับปัจจัยที่ผลต่อค่าใช้จ่ายของการพัฒนาซอฟต์แวร์ หรือในที่นี้เรียกว่าตัวขับเคลื่อนค่าใช้จ่าย(Cost Driver) ดังนั้นการกำหนดค่าสเกลให้กับตัวขับเคลื่อนค่าใช้จ่ายจึงเป็นสิ่งที่สำคัญมากในการใช้วิธีการประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์ เนื่องจากการประมาณค่าใช้จ่ายด้วยวิธีการโคโคโม2 จะใช้ค่าสเกลจากทุกๆ ตัวขับเคลื่อนค่าใช้จ่ายเป็นข้อมูลพื้นฐานในการประมาณการ อย่างไรก็ตามตัวขับเคลื่อนค่าใช้จ่ายของโคโคโม2 มีความซับซ้อนและเข้าใจยาก ทำให้การกำหนดค่าสเกลให้กับตัวขับเคลื่อนค่าใช้จ่ายทำได้ยากสำหรับผู้ที่ไม่มีความรู้ด้านการประมาณค่าใช้จ่าย ดังนั้นในงานวิจัยนี้จึงเสนอเครื่องมือที่ช่วยประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์โดยใช้เทคนิคโคโคโม 2 โดยเครื่องมือที่พัฒนาขึ้นมา มีส่วนที่ต่างจาก โคโคโม 2 ก็คือ มีลักษณะเป็นชุดคำถาม ให้ผู้ใช้ตอบ จากนั้นเครื่องมือจะทำการวิเคราะห์ค่าสเกลและคำนวณค่าประมาณให้โดยอัตโนมัติ

### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

#### 1 การประมาณค่าใช้จ่ายโดยเทคนิคโคโคโม 2

โคโคโม2 เป็นโมเดลที่พัฒนาต่อจากโคโคโม ซึ่งโคโคโมได้พัฒนาครั้งแรกในปี คศ. 1981 [1]โดยการศึกษาโครงการซอฟต์แวร์ที่ได้รับการพัฒนาจนเสร็จสิ้นแล้วจำนวน 83 โครงการ แล้วนำมาวิเคราะห์ปัจจัยที่มีผลต่อค่าใช้จ่ายของการพัฒนาโครงการ ซึ่งมีทั้งหมด 15 ปัจจัยที่มีผลต่อการพัฒนาโครงการ โดยปัจจัยดังกล่าวนี้อาจเรียกว่าเป็นตัวขับเคลื่อนค่าใช้จ่าย อย่างไรก็ตามทีมงานผู้พัฒนาโคโคโมได้เริ่มทำการพัฒนาและปรับปรุงอย่างต่อเนื่องเป็นโคโคโม 2 ในปี คศ.1991[2,3,4] โคโคโม 2โมเดลประกอบด้วย 3 โมเดลย่อยคือ แอปพลิเคชันคอมโพสิชันโมเดล (Application Composition Model) เออร์ดีไซด์โมเดล (Early Design Model) และ โพสอาร์คิเทกเจอร์โมเดล (Post-Architecture Model) ในแต่ละโมเดลย่อยจะมีวิธีการประมาณค่าที่ต่างกันไปขึ้นอยู่กับว่าผู้ใช้สามารถให้ข้อมูลพื้นฐานได้ละเอียดมากน้อยเพียงใด ในงานวิจัยนี้ผู้วิจัยใช้วิธีการประมาณของโพสอาร์คิเทกเจอร์โมเดล เนื่องจากเป็นโมเดลย่อยที่ให้ค่าการประมาณที่ใกล้เคียงที่สุดเมื่อเทียบกับ 2 โมเดลย่อยที่เหลือ แต่ทั้งนี้ผู้ใช้จะต้องให้ข้อมูลพื้นฐานของซอฟต์แวร์ที่จะพัฒนาโดยละเอียดเช่นกัน โพสอาร์คิเทกเจอร์โมเดลประกอบด้วยตัวขับเคลื่อนค่าใช้จ่ายทั้งสิ้น 17 ตัว ซึ่งการกำหนดค่าสเกลให้กับตัวขับเคลื่อนค่าใช้จ่ายนี้สามารถกำหนดค่าได้ 6 ระดับได้แก่ สูงที่สุด(Extra High) สูงมาก(Very High) สูง(High) ปานกลาง(Nominal) ต่ำ (Low) และต่ำที่สุด(Very Low) เพื่อนำมาคำนวณค่าประมาณการ เช่น จำนวนคนที่ใช้ในการพัฒนาซอฟต์แวร์ต่อเดือน(Persons per Month) และระยะเวลาที่ใช้พัฒนามีหน่วยเป็นเดือน

#### 2 โปรแกรมการเรียนรู้เชิงอุปนัย

ในงานวิจัยนี้จะใช้โปรแกรมการเรียนรู้ C4.5 ในการสร้างต้นไม้การตัดสินใจ เพื่อช่วยในการวิเคราะห์หาค่าสเกลที่เหมาะสม โปรแกรมการเรียนรู้ C4.5[5,6]เป็นโปรแกรมการเรียนรู้เชิงอุปนัยที่พัฒนาต่อมาจาก ID3 (Induction Decision Tree) [6,7] ซึ่งเป็นโปรแกรมการเรียนรู้จากตัวอย่างที่อาศัยวิธีการจัดหมวดหมู่(Classification Model) จากตัวอย่างเฉพาะที่เรียกว่าข้อมูลสอน(Training Data) เพื่อนำมาสร้างเป็นต้นไม้ตัดสินใจ(Decision Tree) และ กฎการตัดสินใจ(Rule) ได้โดยอัตโนมัติ ข้อมูลสอนจะมีลักษณะ

คล้ายกับข้อมูลในฐานข้อมูลแบบสัมพันธ์(Relational Database) ที่ประกอบด้วยแถว(Record) หรือตัวอย่าง(Case) และสดมภ์(Column) หรือลักษณะ(Attribute) ซึ่งมี 2 ชนิดด้วยกันคือ

1. ลักษณะแบ่งกลุ่ม(Category Attribute) หรือ กลุ่ม(Class) เป็นการกำหนดว่าตัวอย่างนั้นๆ ถูกจัดไว้ในกลุ่มใด โดยจะมีได้เพียง 1 กลุ่มเท่านั้นในแต่ละตัวอย่าง ดังแสดงในตารางที่ 1

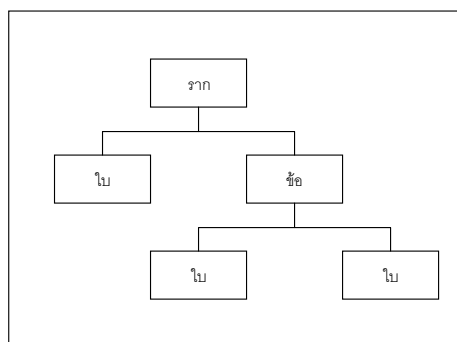
ลักษณะ	ค่าที่เป็นไปได้(กลุ่ม)
ค่าสเกลของตัวขับเคลื่อนค่าใช้จ่าย	ต่ำมาก , ต่ำ, ปานกลาง, สูง, สูงมาก, สูงที่สุด

ตารางที่ 1 ตัวอย่างลักษณะแบ่งพวก

2. ลักษณะไม่แบ่งกลุ่ม(Non-Category Attribute) เป็นลักษณะต่างๆ ของตัวอย่าง โดยอาจจะเก็บข้อมูลได้ทั้งชนิดค่าต่อเนื่องและ ค่าไม่ต่อเนื่อง ดังแสดงในตารางที่ 2 การสร้างต้นไม้การตัดสินใจจะเริ่มที่ราก(root)ก่อนแล้วจึงไล่ลงมาที่ข้อ(node) และใบ(leaf) ซึ่งข้อ คือข้อมูลลักษณะไม่แบ่งพวก ส่วนใบคือข้อมูลลักษณะแบ่งพวก ดังแสดงในรูปที่ 1 จะเห็นว่าการสร้างต้นไม้การตัดสินใจ จะเป็นการแบ่งตัวอย่างออกเป็นกลุ่มๆ ตามลักษณะไม่แบ่งกลุ่ม จนกระทั่งได้กลุ่มตัวอย่างที่เป็นกลุ่มเดียวกัน

ลักษณะ	ค่าที่เป็นไปได้
1) ซอฟต์แวร์ที่พัฒนาจะถูกนำไปใช้ในด้านที่เกี่ยวข้องกับชีวิตมนุษย์ หรือในด้านที่มีผลต่อชีวิตมนุษย์	ใช่ ไม่ใช่
2) ซอฟต์แวร์ที่จะพัฒนาสามารถทำให้เกิดความเสี่ยงต่อชีวิตและร่างกายของมนุษย์ได้	ใช่ ไม่ใช่
3) ซอฟต์แวร์ที่จะพัฒนาเป็นซอฟต์แวร์ที่เกี่ยวข้องกับรายรับหรือรายจ่ายขององค์กรเป็นอย่างมาก	ใช่ ไม่ใช่
4) ถ้าซอฟต์แวร์ไม่สามารถทำงานได้ตามปกติจะทำให้องค์กรเสียค่าใช้จ่ายหรือ เสียรายได้จำนวนมาก	ใช่ ไม่ใช่
5) ถ้าซอฟต์แวร์ไม่สามารถทำงานได้ตามปกติ ท่านจำเป็นต้องดำเนินการให้มีการแก้ไขทันที	ใช่ ไม่ใช่
6) การแก้ไขให้ซอฟต์แวร์สามารถกลับมาทำงานได้ตามปกติ ท่านคาดว่าจะทำให้เกิดการสูญเสียทางการเงินหรือทำให้เกิดความเสียหายกับข้อมูลโดยเฉลี่ยไม่มาก	ใช่ ไม่ใช่
7) ท่านสามารถดำเนินงานในลักษณะนั้นต่อได้ ถึงแม้ว่าซอฟต์แวร์จะไม่สามารถทำงานได้ก็ตาม	ใช่ ไม่ใช่

ตารางที่ 2 ตัวอย่างลักษณะไม่แบ่งพวก



รูปที่ 1 ลักษณะของต้นไม้

### ขั้นตอนการพัฒนา

ในการวิจัยนี้ ผู้วิจัยได้แบ่งขั้นตอนการพัฒนางานวิจัยออกเป็น 4 ขั้นตอนหลักๆ คือ การสร้างชุดคำถาม การกำหนดชุดคำตอบจากผู้เชี่ยวชาญ การสอนโปรแกรมการเรียนรู้C4.5 และ การทดสอบชุดคำถาม ซึ่งผลลัพธ์จากขั้นตอนดังกล่าวจะนำมาสร้างเป็นซอฟต์แวร์ที่ใช้ประมาณค่าใช้จ่ายที่เหมาะสมต่อไป

#### ขั้นที่ 1 กระบวนการสร้างชุดคำถาม

ผู้วิจัยได้ศึกษาและค้นคว้า ข้อมูลที่เกี่ยวข้องการวิธีการประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์โดยวิธีการของโคโคโม2 และงานวิจัยอื่นๆ ที่เกี่ยวข้อง[8,9] แล้วนำมาวิเคราะห์และสร้างเป็นคำถาม ซึ่งคำถามที่สร้างขึ้นในแต่ละตัวซับซ้อนค่าใช้จ่ายอาจมีจำนวนข้อไม่เท่ากัน เนื่องจากผู้วิจัยเน้นการสร้างคำถามที่ครอบคลุมคำจำกัดความ(Definitions) ที่ได้ถูกกำหนดไว้ใน[9]เป็นสำคัญ ซึ่งผู้วิจัยได้ใช้คำจำกัดความในแต่ละตัวซับซ้อนค่าใช้จ่ายเป็นพื้นฐานในการสร้างคำถามของตัวซับซ้อนค่าใช้จ่ายในประเภทนั้นๆ คำถามที่สร้างขึ้นถูกแบ่งออกเป็น 17 กลุ่มตามประเภทของตัวซับซ้อนค่าใช้จ่าย โดยมีคำถามรวมทั้งสิ้น 93 ข้อ คำถามที่สร้างขึ้นนี้ผู้วิจัยพัฒนาตามทฤษฎีของการสร้างคำถามที่ใช้ทดสอบ เก็บรวบรวมข้อมูล หรือรวบรวมความคิดเห็น[10,11] คำถามที่พัฒนาขึ้นในงานวิจัยนี้แบ่งออกเป็น 2 กลุ่มใหญ่ๆ ตามลักษณะของคำตอบ ได้แก่ คำถามแบบปลายปิด ซึ่งผู้ใช้สามารถเลือกคำตอบได้เพียง 2 ทางเลือกคือ ใช่หรือไม่ใช่ และคำถามแบบปลายเปิด ซึ่งผู้ใช้สามารถเติมคำตอบลงในช่องที่กำหนดไว้ รายละเอียดของชุดคำถามดูได้ที่[12]

#### ขั้นที่ 2 การกำหนดชุดคำตอบ

ผู้วิจัยได้นำชุดคำถามที่ได้จากขั้นตอนที่หนึ่ง มาสร้างเป็นแบบสอบถามเพื่อรวบรวมชุดคำตอบจากผู้เชี่ยวชาญ โดยที่ผู้เชี่ยวชาญหมายถึง ผู้ที่มีประสบการณ์ในการประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์ นักออกแบบระบบ หรือผู้บริหารโครงการ ในขั้นนี้ผู้เชี่ยวชาญกำหนดชุดของคำตอบในแต่ละตัวซับซ้อนค่าใช้จ่าย ซึ่งชุดคำตอบประกอบด้วย คำตอบในแต่ละข้อของคำถามและค่าสเกลที่เหมาะสมกับคำตอบนั้น ดังตัวอย่างใน ซึ่งจะนำมาเป็นข้อมูลสอนหรือตัวอย่าง ให้กับโปรแกรมการเรียนรู้ในขั้นถัดไป

### ขั้นที่ 3 การสอนโปรแกรมการเรียนรู้

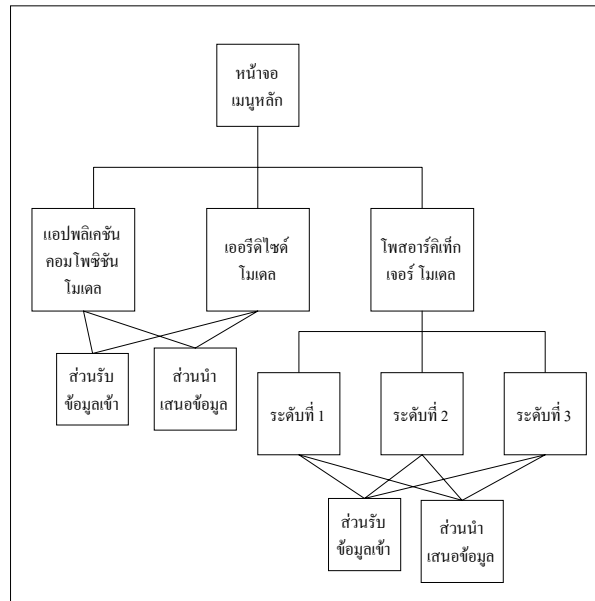
นำชุดคำตอบที่ได้จากขั้นที่ 2 มาเป็นข้อมูลสอน(Training Data) ให้กับโปรแกรมการเรียนรู้C4.5 โดยผู้วิจัยจะเลือกเฉพาะคำตอบของตัวขับเคลื่อนค่าใช้จ่าย 6 ตัวขับเคลื่อนค่าใช้จ่าย คือ ความเชื่อถือได้ของผลิตภัณฑ์ซอฟต์แวร์(RELY) ความซับซ้อนของผลิตภัณฑ์ซอฟต์แวร์(CPLX) ความต้องการที่จะนำผลิตภัณฑ์ซอฟต์แวร์กลับมาใช้ใหม่(RUSE) ความต้องการเอกสารที่ตรงตามวงจรชีวิต(DOCU) การใช้เครื่องมือพัฒนาซอฟต์แวร์(TOOL) และ สถานที่ที่พัฒนาซอฟต์แวร์(SITE) เป็นข้อมูลสอน เนื่องจากคำตอบของตัวขับเคลื่อนค่าใช้จ่ายทั้ง 6 ตัวนี้มีลักษณะเป็นปลายปิดซึ่งสามารถนำมาใช้ได้กับโปรแกรมการเรียนรู้ C4.5 เมื่อโปรแกรมการเรียนรู้ C4.5ได้รับข้อมูลสอนแล้วก็จะสร้าง(Generate) เป็นต้นไม้การตัดสินใจ(Decision Tree) ที่ดีที่สุดออกมาให้ 1 ต้น ซึ่งต้นไม้การตัดสินใจจะประกอบด้วย คำถามในแต่ละข้อซึ่งเป็นข้อ(node) ของต้นไม้ และ ระดับที่เลือกซึ่งเป็นใบ(leaf)

### ขั้นที่ 4 การทดสอบชุดคำถาม

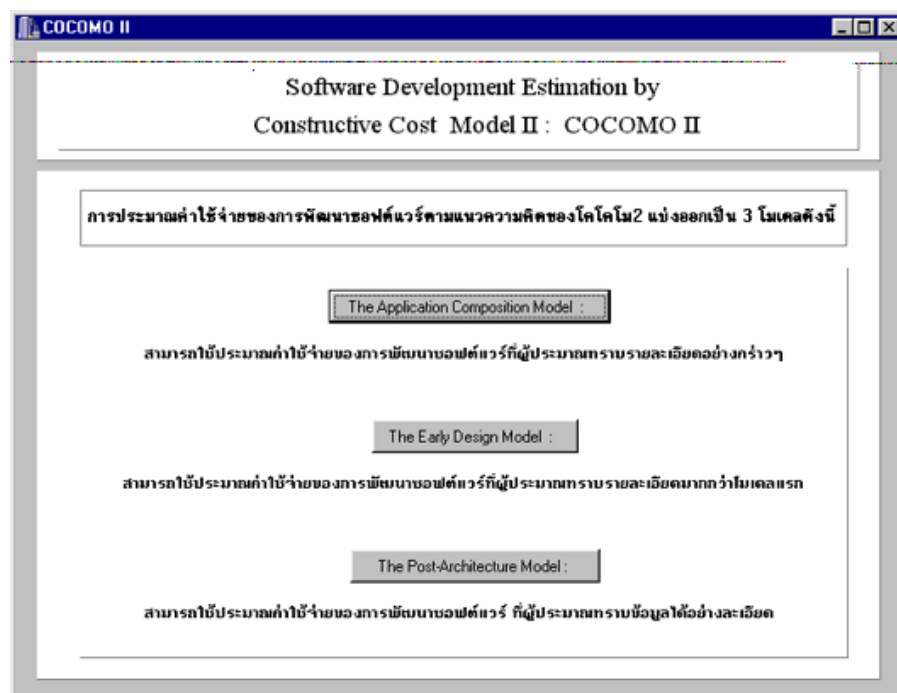
ผู้วิจัยได้ทำการทดสอบความเหมาะสมของชุดคำถามโดยสร้างแบบสอบถามขึ้นมา 2 ชุด โดยแบบสอบถามชุดที่ 1 มีลักษณะเช่นเดียวกับโคโคโม2 คือมีคำอธิบายทุกๆ ตัวขับเคลื่อนค่าใช้จ่ายซึ่งมีทั้งสิ้น 21 ข้อ ส่วนแบบสอบถามชุดที่ 2 มีลักษณะเดียวชุดคำถามที่ได้จากขั้นที่ 1 ซึ่งมีทั้งสิ้น 93 ข้อ ในที่นี้ผู้วิจัยได้กำหนดลักษณะของโครงการที่ต้องการจะประมาณค่าใช้จ่ายซึ่งประกอบด้วย ข้อจำกัด (Constraints) และ สิ่งที่ต้องการ (Requirements) จากนั้นนำแบบสอบถามไปให้ผู้ตอบ 2 กลุ่ม กลุ่มที่ 1 คือกลุ่มผู้เชี่ยวชาญจำนวน 10 ท่าน ซึ่งแต่ละท่านจะได้ทำแบบสอบถามทั้ง 2 ชุด จากนั้นนำค่าประมาณการที่ได้จากแบบสอบถามทั้ง 2 ชุดมาเปรียบเทียบกัน ส่วนในกลุ่มที่ 2 คือกลุ่มผู้ที่ไม่เคยมีประสบการณ์ในการประมาณค่าใช้จ่ายมาก่อนแต่มีประสบการณ์การพัฒนาซอฟต์แวร์อย่างน้อย 3 ปี จำนวนทั้งสิ้น 10 ท่าน จะทำแบบสอบถามชุดที่ 2 เพียงชุดเดียว จากนั้นผู้วิจัยได้นำค่าประมาณที่ได้มาเปรียบเทียบกับค่าประมาณการที่ได้จากแบบสอบถามชุดที่ 2 ของกลุ่มที่ 1 ซึ่งผลปรากฏว่าค่าประมาณที่ได้จากการใช้ชุดคำถามมีความใกล้เคียง กับค่าประมาณการที่ได้จากโคโคโม 2 ด้วยความเชื่อมั่น 95 %

### การพัฒนาเครื่องมือ

เครื่องมือนี้ใช้โปรแกรมภาษาบอร์แลนด์ซี(Borland C++) ในการพัฒนา ภายใต้ระบบปฏิบัติการวินโดวส์ และใช้ไมโครซอฟต์แอ็กเซสเป็นโปรแกรมจัดการฐานข้อมูล เครื่องคอมพิวเตอร์ที่ใช้เป็นเครื่องคอมพิวเตอร์ส่วนบุคคล รุ่นเพนเทียม 133 หน่วยความจำ 32 เมกะไบต์ โดยโครงสร้างของเครื่องมือนี้แบ่งออกเป็นส่วนๆ ดังรูปที่ 2 และส่วนของหน้าจอเมนูหลักดังรูปที่ 3 ซึ่งมีรายละเอียดอื่นๆดังนี้



รูปที่ 2 โมดูลของโปรแกรม



รูปที่ 3 หน้าจอเมนูหลัก

## 1 แอปพลิเคชันคอมโพสิชันโมเดล

การประมาณค่าใช้จ่ายโดยใช้แอปพลิเคชันคอมโพสิชันโมเดล เป็นวิธีการประมาณที่ผู้ใช้ไม่ทราบรายละเอียดของซอฟต์แวร์ที่จะพัฒนามากนัก แต่จะทราบส่วนที่ซอฟต์แวร์ติดต่อกับผู้ใช้ คือจำนวนรายการ จำนวนหน้าจอ และวิธีการประมาณนี้จะให้ค่าการประมาณที่หยาบมากที่สุดเมื่อเปรียบเทียบกับอีก 2 โมเดลที่เหลือ โดยการรับข้อมูลเข้าของแอปพลิเคชันคอมโพสิชันโมเดลนั้นเป็น การประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์ใหม่ ผู้ใช้ต้องป้อนชื่อโครงการและวันที่ทำการประมาณ แล้วโปรแกรมจะทำการตรวจสอบชื่อโครงการกับวันที่ทำการประมาณถ้ามีรายการซ้ำในแฟ้มข้อมูลก็จะให้ผู้ใช้ทำการแก้ไข แล้วจึงแสดงส่วนรับข้อมูลเข้า เพื่อให้ผู้ใช้ได้ป้อนข้อมูลที่จำเป็นสำหรับการประมาณการ เสร็จแล้วจึงทำการคำนวณค่าประมาณการ ส่วนการนำเสนอการนำเสนอในรูปของหน้าจอซึ่งมีรายละเอียดข้อมูลของซอฟต์แวร์ที่ได้ทำการประมาณการ เช่น ชื่อโครงการ วันที่ทำการประเมิน ผู้ประเมิน และ ค่าการประเมิน เป็นต้น ดังแสดงในรูปที่ 4

รูปที่ 4 ส่วนรับและนำเสนอข้อมูลเข้าของแอปพลิเคชันคอมโพสิชันโมเดล

รูปที่ 5 ส่วนรับข้อมูลเข้าของเออร์ลี่ไดไซน์โมเดล

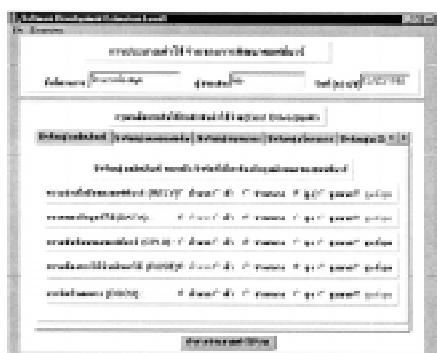
## 2 รีดิไซด์โมเดล

การประมาณค่าใช้จ่ายโดยใช้เออร์ลี่ไซด์โมเดล เป็นวิธีการประมาณที่ผู้ใช้ต้องทราบรายละเอียดของซอฟต์แวร์ที่จะพัฒนา เช่น ตัวขับเคลื่อนค่าใช้จ่ายทั้ง 7 ลักษณะ และขนาดของซอฟต์แวร์ วิธีการประมาณนี้จะให้ค่าการประมาณที่ใกล้เคียงมากกว่าโมเดลแรก โดยการรับข้อมูลเข้าของเออร์ลี่ไซด์โมเดลนั้นเป็นการประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์ใหม่ ผู้ใช้ต้องป้อนชื่อโครงการและวันที่ทำการประมาณ แล้วโปรแกรมจะทำการตรวจสอบชื่อโครงการกับวันที่ทำการประมาณถ้ามีรายการซ้ำในแฟ้มข้อมูลก็จะให้ผู้ใช้ทำการแก้ไข แล้วจึงแสดงส่วนรับข้อมูลเข้าดังรูปที่ 5 เพื่อให้ผู้ใช้ได้ป้อนข้อมูลที่จำเป็นสำหรับการประมาณการ คือ ตัวขับเคลื่อนค่าใช้จ่ายทั้ง 7 ตัว เสร็จแล้วจึงทำการคำนวณค่าประมาณการ แล้วนำเสนอในรูปของหน้าจอซึ่ง

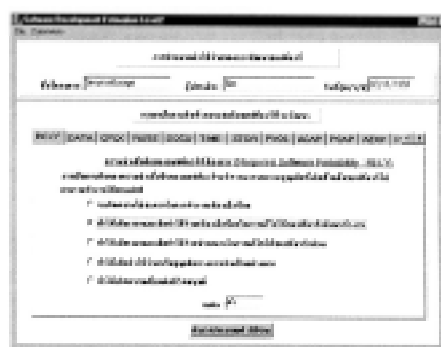
มีรายละเอียดข้อมูลของซอฟต์แวร์ที่ได้ทำการประมาณการ เช่น ชื่อโครงการ วันที่ทำการประเมิน ผู้ประเมิน ระดับของตัวขับเคลื่อนค่าใช้จ่ายทั้ง 7 ตัว และ ค่าการประเมิน เป็นต้น

### 3 โพสอาร์คิเทกเจอร์โมเดล

การประมาณค่าใช้จ่ายโดยใช้โพสอาร์คิเทกเจอร์โมเดลเป็นวิธีการประมาณที่ผู้ใช้งานต้องทราบรายละเอียดของซอฟต์แวร์ที่จะพัฒนา เช่น ตัวขับเคลื่อนค่าใช้จ่ายทั้ง 17 ลักษณะ และขนาดของซอฟต์แวร์ วิธีการประมาณนี้จะให้ค่าการประมาณที่ใกล้เคียงมากที่สุด และ ในโมเดลนี้จะมีเมนูย่อย ซึ่งเป็นเมนูของระดับของความสามารถที่ผู้ใช้งานสามารถเลือกให้เหมาะกับตนเอง ซึ่งมี 3 ระดับดังนี้ ระดับแรกสำหรับผู้ที่มีความชำนาญในการประมาณการค่าใช้จ่ายแบบโคโคโม2 ระดับที่สองสำหรับผู้ที่มีความรู้ด้านการประมาณการค่าใช้จ่ายแต่ไม่เคยใช้โคโคโม2 และ ระดับที่สามสำหรับผู้ที่ไม่มีความรู้ด้านการประมาณการค่าใช้จ่ายและไม่เคยใช้โคโคโม2 โดยการรับข้อมูลเข้าของโพสอาร์คิเทกเจอร์โมเดล คือการประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์ใหม่ ผู้ใช้ต้องป้อนชื่อโครงการและวันที่ทำการประมาณ แล้วโปรแกรมจะทำการตรวจสอบชื่อโครงการกับวันที่ทำการประมาณถ้ามีรายการซ้ำในแฟ้มข้อมูลก็จะให้ผู้ใช้ทำการแก้ไข แล้วจึงแสดงส่วนรับข้อมูลเข้า เพื่อให้ผู้ใช้ได้ป้อนข้อมูลที่จำเป็นสำหรับการประมาณการ คือ ตัวขับเคลื่อนค่าใช้จ่ายทั้ง 17 ตัว เสร็จแล้วจึงทำการคำนวณค่าประมาณการ ซึ่งมีระดับให้ผู้ใช้ได้เลือกใช้ ถึง 3 ระดับด้วยกัน ซึ่งก็จะมีหน้าจอที่แตกต่างกันไป ดังแสดงในรูปที่ 6 –8



รูปที่ 6 ส่วนรับข้อมูลเข้าของโพสอาร์คิเทกเจอร์โมเดลระดับที่ 1



รูปที่ 7 ส่วนรับข้อมูลเข้าของโพสอาร์คิเทกเจอร์โมเดลระดับที่ 2





รูปที่ 8 ส่วนรับข้อมูลเข้าของโพสอาร์คิเท็กเจอร์โมเดลระดับที่3

### สรุปการวิจัยและแนวทางการวิจัยต่อไป

จากผลการวิจัยจะเห็นว่า ค่าประมาณที่ได้โคโม 2 และค่าประมาณที่ได้จากชุดคำถามมีความใกล้เคียงกันมาก จึงทำให้สรุปได้ว่าชุดคำถามที่สร้างขึ้น และ ต้นไม้การตัดสินใจ มีความเหมาะสมสามารถนำมาใช้เพื่อช่วยหาสเกลสำหรับตัวซับซ้อนค่าใช้จ่ายในโคโม 2 และสามารถพัฒนาเป็นเครื่องมือที่ใช้ประมาณค่าใช้จ่ายของการพัฒนาซอฟต์แวร์ได้ แต่ทั้งนี้เนื่องจากกลุ่มตัวอย่างที่ใช้ทดสอบชุดคำถามยังมีจำนวนไม่มากเพียงพอ ดังนั้นจึงเห็นว่า ควรมีการทดสอบชุดคำถาม โดยใช้ผู้เชี่ยวชาญเพิ่มขึ้นอีกจำนวนหนึ่ง(30 ท่านเป็นอย่างน้อย)

### เอกสารอ้างอิง

1. Boehm, B., Software Engineering Economics, Prentice-Hall, Inc., Englewood Cliffs, N.J.,1981.
2. Clark, B.; Devnani-Chulani, S.; Boehm, B., "Calibrating the COCOMOII Post-Architecture Model," Software Engineering, 1998. Proceedings of the International Conference on , Page(s): 477 –480,1998
3. Clark, B.K., "Cost modeling process maturity-COCOMO 2.0," Aerospace Applications Conference, Proceedings., 1996 IEEE Volume: 3 , Page(s): 347 -369 ,1996.
4. Mukhopadhyay, T.; Kekre, S. "Software effort models for early estimation of process control applications," Software Engineering, IEEE Transactions Volume: 18 , Page(s): 915 –924.
5. กำพล ปัญญาเขษมา นิต,การประยุกต์การเรียนรู้กับการอนุมัตินิเชื่อ,วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2540.
6. Quinlan J.R , C4.5.Programs for Machine learning ,San Matco,CA: Morgan Kaufmann,1993.
7. Quinlan J.R ,Reading in Machine Learning ,CA: Morgan Kafmann,1991
8. Sommerville, I., Software Engineering, 5<sup>th</sup> Edition, Addison-Wesley, 1996

9. Center for Software Engineering, "COCOMOII Model definition Manual," Computer Science Department, University of Southern California, Los Angeles, <http://sunset.usc.edu/CocomoII/Cocomo.html>, 1997.
10. จุมพล สวัสดิยากร, การร่างแบบสอบถามและ ABC ในการวิจัย, โรงพิมพ์สุวรรณภูมิ, 2526.
11. Center for Software Engineering, "COCOMOII Cost Estimation Questionnaire," Computer Science Department, University of Southern California, Los Angeles, <http://sunset.usc.edu/CocomoII/Cocomo.html>, 1997
12. ศรีนทร์ วัชรบุศราคำ, การพัฒนาเครื่องมือช่วยประเมินค่าใช้จ่ายของการพัฒนาซอฟต์แวร์ โดยเทคนิคโคโคโม 2, วิทยานิพนธ์มหาบัณฑิตจุฬาลงกรณ์มหาวิทยาลัย, 2542
13. Ronald E. Walpole and Raymond H. Myers. Probability and Statistics for Engineers and Scientists, 5<sup>th</sup> ed. New Jersey: A Simon & Schuster Company, 1972.