แบบจำลองการประเมินความเสี่ยงของอัตราความเสียหายของซอฟต์แวร์จากการมีอคติทางการรู้คิด

นายนพธนิษฐ์ โชติสาร

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2555

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

COGNITIVE BIAS RISK ASSESSMENT MODEL OF SOFTWARE DAMAGE RATE

Mr. Noptanit Chotisarn

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Software Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2012

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	แบบจำลองการประเมินความเสี่ยงของอัตราความเสียหายของซอฟต์แวร์จากการมีอคติทางการรู้คิด
โดย	นายนพธนิษฐ์ โชติสาร
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

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

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล)

..... กรรมการ
(รองศาสตราจารย์ ดร.ทวิตีย์ เสนีวงศ์ ณ อยุธยา)

..... กรรมการภายนอกมหาวิทยาลัย
(รองศาสตราจารย์ ดร.วีระ บุญจริง)

นพธนิษฐ์ โชติสาร : แบบจำลองการประเมินความเสี่ยงของอัตราความเสียหายของซอฟต์แวร์จากการมีอคติทางการรู้คิด (COGNITIVE BIAS RISK ASSESSMENT MODEL OF SOFTWARE DAMAGE RATE) อ.ที่ปรึกษาวิทยานิพนธ์หลัก : ผศ.นครทิพย์ พร้อมพูล, 175 หน้า.

ปัจจุบันอุตสาหกรรมซอฟต์แวร์มีการนำหลักวิศวกรรมซอฟต์แวร์มาใช้มากขึ้น แต่ยังมีคำถามเกิดขึ้นว่าหลักการวิศวกรรมซอฟต์แวร์ที่มีอยู่ในปัจจุบันนั้นเพียงพอต่อการพัฒนาซอฟต์แวร์ได้อย่างมีคุณภาพหรือไม่ และจากการที่มนุษย์เป็นทรัพยากรที่สำคัญที่สุดในโครงการซอฟต์แวร์นั้น มนุษย์ก็น่าจะเป็นปัจจัยสำคัญหนึ่งต่อความสำเร็จของโครงการซอฟต์แวร์ในเชิงคุณภาพ โดยเรียกปัจจัยจากมนุษย์นี้ว่า "มนุษยปัจจัย" ซึ่งงานวิจัยนี้สนใจศึกษามนุษยปัจจัย ทางด้านจิตวิทยาของการมีอคติทางการรู้คิด โดยมองว่าการมีอคตินี้ น่าจะส่งผลต่อโครงการพัฒนาซอฟต์แวร์ ที่จะนำมาซึ่งความเสียหายกับซอฟต์แวร์ได้ โดยศึกษาอคติเหล่านี้ในขั้นตอนแรก ๆ ของวิศวกรรมซอฟต์แวร์ นั่นคือ ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ จึงน่าจะมีการประเมินปัจจัยความเสี่ยงในด้านนี้ด้วยในช่วงต้นของโครงการพัฒนาซอฟต์แวร์ เพื่อช่วยให้ผู้มีส่วนเกี่ยวข้องในโครงการ ตระหนักถึงความเสี่ยงที่อาจจะเกิดขึ้นมาได้ในโครงการพัฒนาซอฟต์แวร์ งานวิจัยนี้จึงมีวัตถุประสงค์ในการนำเสนอวิธีการเพื่อสร้างแบบจำลองการประเมินความเสี่ยงต่อความเสียหายของซอฟต์แวร์ ในด้านการมีอคติทางการรู้คิดของมนุษย์ในทั้งสองกระบวนการ

ผลการทดลองพบว่าสามารถพยากรณ์ความเสียหายได้ในระดับที่ต่ำถึงค่อนข้างต่ำ ซึ่งทำให้ในการคำนวณค่าความเสี่ยง ก็จะสามารถหาค่าความเสี่ยงซึ่งก็มีความสามารถในการพยากรณ์ต่ำด้วย ถึงแม้ว่าจะไม่สามารถพยากรณ์ออกมาได้แม่นยำมาก แต่ก็ใช่ว่าจะไม่มีความเสียหายที่เกิดจากประเด็นทางด้านการมีอคติทางการรู้คิดเลย และพบว่ามีความสมเหตุสมผลในระดับหนึ่งกับลักษณะของการมีอคติในเชิงจิตวิทยา ที่อคติไม่ได้ส่งผลในแง่ลบเพียงแง่เดียว หรือส่งผลบวกเพียงแง่เดียวเช่นกันกับการมีอคติกับการทำงานในงานวิศวกรรมซอฟต์แวร์ ทั้งในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ดังที่ได้ทำการวิจัย ในบางครั้งก็พบว่าการมีอคติมากเกินไป หรือน้อยเกินไปก็อาจจะส่งผลให้เกิดความเสียหายได้มากกว่าการมีอคติในระดับปานกลางได้เช่นกัน ดังนั้นในการบริหารจัดการความเสี่ยง ควรที่จะให้ความสำคัญกับประเด็นนี้เช่นกัน เพราะอาจจะเป็นสาเหตุหนึ่งของความล้มเหลวหรือความสำเร็จของโครงการได้เช่นกัน

ภาควิชา ...วิศวกรรมคอมพิวเตอร์.... ลายมือชื่อนิสิต .....

สาขาวิชา ...วิศวกรรมซอฟต์แวร์.... ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก .....

ปีการศึกษา ...2555.....

5470951021 : MAJOR SOFTWARE ENGINEERING

KEYWORDS : HUMAN FACTOR/ SOFTWARE DESIGN/ COGNITIVE BIAS/ RISK ASSESSMENT/ SOFTWARE REQUIREMENTS

NOPTANIT CHOTISARN : COGNITIVE BIAS RISK ASSESSMENT MODEL OF SOFTWARE DAMAGE RATE. ADVISOR : ASST. PROF. NAKORNTHIP PROMPOON, 175 pp.

Nowadays software engineering principles have increasingly been used in software industry. There is the question arising whether the existing software engineering principles are sufficient for the development of quality software. Humans are the most important resources in a software project. It seems to be one of the major factors for the success of a software project in terms of quality. The factor from human is called "Human factor". This research is interested in the study of behavioral perspective of human factor in term of psychology, especially cognitive bias that is likely to cause the damage of software development project. This may create a potential risk to software projects. The potential biases studied in this research are in the early stages of software engineering composed of the software requirements gathering and specification process and the software design process. This research aims to present a method to create a model for risk assessment in terms of the human cognitive bias that affects the process of requirements gathering and definition and software design process.

The results showed that the prediction of damage levels were relatively low. This allows risk calculation will be able to determine the risks levels were relatively low too. Although the prediction was unable to forecast accurately, there was likely a cause of software damage from the issue of the cognitive bias and a reasonable extent on the nature of the psychological. The bias does not only result in a negative aspect but also results in a positive negative aspect to the work in software engineering in both the software requirements gathering and specification process and the software design process. It is sometimes found that the bias occurrence level as very low or very high has an effect to the software damage more than the occurrence level as medium. Therefore, in the risk management should focus on this issue as well since it may be the cause of the failure or success of the project as well.

Department: ...Computer Engineering.... Student's Signature .....

Field of Study: ..Software Engineering.... Advisor's Signature .....

Academic Year: ...2012.....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ได้รับคำแนะนำ ข้อคิดเห็น และแนวทางในการทำวิจัยที่ดีต่าง ๆ จากอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล ที่คอยช่วยเหลือให้ความรู้ทางด้านที่จำเป็นต่อวิทยานิพนธ์ และคำปรึกษาที่สามารถแก้ไขปัญหาต่าง ๆ ที่เกิดขึ้นได้ นอกจากนั้นยังถ่ายทอดประสบการณ์ต่าง ๆ ให้เป็นข้อคิด และแนวทางที่ดีต่อการทำงานต่อไป ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

กราบขอบพระคุณคณะกรรมการสอบวิทยานิพนธ์ รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์ รองศาสตราจารย์ ดร.ทวิตีย์ เสนีวงศ์ ณ อยุธยา และ รองศาสตราจารย์ ดร.วีระ บุญจริง ที่สละเวลามาให้ข้อเสนอแนะ และแนวคิดต่างๆ ที่เป็นประโยชน์ต่อการพัฒนาวิทยานิพนธ์ฉบับนี้

ขอขอบพระคุณหัวหน้างานและเพื่อนร่วมงาน บริษัท ทรู อินฟอร์เมชั่น เทคโนโลยี จำกัด ที่ให้การสนับสนุนในการศึกษาต่อมาโดยตลอด ขอบคุณเพื่อน ๆ มหาวิทยาลัยธรรมศาสตร์ และสมาชิกห้องปฏิบัติการวิศวกรรมซอฟต์แวร์ จุฬาลงกรณ์มหาวิทยาลัย ทั้งเพื่อน ๆ พี่ ๆ น้อง ๆ โดยเฉพาะอย่างยิ่ง นางสาวพิชชากร เอกวรานุกูลศิริ ที่ให้ความอนุเคราะห์ในการทำแบบสอบถาม ให้คำแนะนำและความช่วยเหลืออย่างดีเสมอมา

สุดท้ายขอกราบขอบพระคุณ คุณยาย คุณแม่ คุณลุง พี่สาว พี่ชาย และขอบคุณน้องสาว น้องชาย และญาติ ๆ ที่คอยสนับสนุนให้ผู้ทำวิจัยมีโอกาสได้ศึกษา ทำในสิ่งที่ดี และคอยเป็นกำลังใจให้กันและกันตลอดมา

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง
บทคัดย่อภาษาอังกฤษ ..... จ
กิตติกรรมประกาศ ..... ฉ
สารบัญ ..... ช
สารบัญตาราง ..... ฎ
สารบัญภาพ ..... ต
บทที่ 1 บทนำ 1
1.1 ที่มาและความสำคัญของปัญหา 1
1.2 วัตถุประสงค์ของการวิจัย 2
1.3 ขอบเขตของงานวิจัย 2
1.4 ขั้นตอนและวิธีดำเนินการวิจัย 3
1.5 ประโยชน์ที่คาดว่าจะได้รับ 3
1.6 ผลงานที่ได้รับการตีพิมพ์ 4
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 5
2.1 ทฤษฎีที่เกี่ยวข้อง 5
2.1.1 อคติทางการรู้คิด (Cognitive Bias) 5
2.1.1.1 การใช้เหตุผลและการตัดสินใจ 5
2.1.1.2 การประมาณความถี่ (Frequency Estimates) 6
2.1.1.3 ประเภทของอคติที่ใช้ในการแก้ปัญหา 7
2.1.1.4 ความสัมพันธ์ระหว่างอคติ 7
2.1.2 การจัดการความเสี่ยง (Risk Management) 10
2.1.3 ความน่าจะเป็น (Probability) 11
2.1.3.1 นิยามความน่าจะเป็น 12
2.1.3.2 การแจกแจงความน่าจะเป็นแบบปัวซอง (Poisson Probability Distribution) 12
2.1.4 แบบจำลองข้อมูลสองขั้นตอน (Two Stage Information Model) 14
2.1.5 การวิเคราะห์ความถดถอย (Regression Analysis) 14

หน้า

2.1.5.1 การวิเคราะห์ความถดถอยและสหสัมพันธ์อย่างง่าย (Simple Regression Analysis and Correlation) 14
2.1.5.2 การวิเคราะห์ความถดถอยและสหสัมพันธ์เชิงซ้อน (Multiple Regression and Correlation Analysis) 16
2.1.6 ตัวแปรหุ่น (Dummy Variable) 17
2.2 งานวิจัยที่เกี่ยวข้อง 18
2.2.1 งานวิจัยเรื่อง “Why Do We Need Personality Diversity in Software Engineering” 19
2.2.2 งานวิจัยเรื่อง “Empirical Analyses of the Factors Affecting Confirmation Bias and the Effects of Confirmation Bias on Software Developer/Tester Performance” 22
2.2.3 งานวิจัยเรื่อง “Software Designers, Are You Biased?” 23
2.2.4 งานวิจัยเรื่อง “The effects of optimistic and pessimistic biasing on software project status reporting” 24
2.2.5 งานวิจัยเรื่อง “Using Traceability to Mitigate Cognitive Biases in Software Development” 26
2.2.6 งานวิจัยเรื่อง “Cross-Cultural Risk Assessment Model” 28
2.2.7 งานวิจัยเรื่อง “Framework for Developing a Software Cost Estimation Model for Software Modification Based on a Relational Matrix of Project Profile and Software Cost Using and Analogy Estimation Method” 29
บทที่ 3 แบบจำลองการประเมินความเสี่ยงจากการมีอคติทางการรู้คิด 31
3.1 แบบจำลองการประเมินความเสี่ยง 31
3.1.1 วิเคราะห์ปัจจัยทางด้านอคติทางการรู้คิด 33
3.1.1.1 อคติ Anchoring and adjustment 33
3.1.1.2 อคติ Availability 35
3.1.1.3 อคติ Confirmation 35
3.1.2 วิเคราะห์กระบวนการวิศวกรรมซอฟต์แวร์ที่สนใจ 36
3.1.3 วิเคราะห์หาวิธีการคำนวณค่าความเสี่ยง 37

หน้า

3.1.4 กำหนดปัจจัยการมีอคติให้เป็นค่าระดับ....37
3.1.5 การกำหนดค่าระดับของปัจจัยการมีอคติให้เป็นตัวแปรหุ่น....39
3.1.6 การกำหนดค่าอัตราความเสียหาย....42
3.1.7 การวิเคราะห์ความถดถอยเพื่อสร้างสมการพยากรณ์อัตราความเสียหายโดยประมาณ...44
3.1.8 คำนวณค่าอัตราความเสียหายประมาณจากสมการพยากรณ์....45
3.1.9 คำนวณค่าความน่าจะเป็น....48
3.1.10 การคำนวณค่าความเสี่ยง....50
3.1.11 คำนวณมูลค่าความเสียหายจากการมีความเสี่ยง....51
3.2 การสร้างแบบสอบถาม....52
3.2.1 ส่วนอธิบายแบบสอบถาม....53
3.2.2 แบบสอบถามข้อมูลส่วนบุคคล....53
3.2.3 แบบสอบถามสำหรับการพบอคติในทั้งสองกระบวนการ....53
บทที่ 4 การทดลองและการวิเคราะห์ผล....62
4.1 ทดลองกับกลุ่มตัวอย่าง....63
4.2 วิเคราะห์ข้อมูลที่ได้จากกลุ่มตัวอย่างสำหรับการวิเคราะห์ความเสียหาย....63
4.3 วิเคราะห์ความถดถอย....87
4.4 วิเคราะห์ผลจากการวิเคราะห์ความถดถอย....87
4.4.1 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์จากอคติประเภท Anchoring and adjustment....88
4.4.2 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์จากอคติประเภท Availability....90
4.4.3 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์จากอคติประเภท Confirmation....92
4.4.4 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการออกแบบซอฟต์แวร์จากอคติประเภท Anchoring and adjustment....95
4.4.5 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการออกแบบซอฟต์แวร์จากอคติประเภท Availability....97
4.4.6 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการออกแบบซอฟต์แวร์จากอคติประเภท Confirmation....99
4.5 วิเคราะห์ผลของการประมาณความเสียหาย.... 101

4.5.1 ระดับความสัมพันธ์และแนวโน้มของการมีอคติกับผลของความเสียหายของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ 102
4.5.2 ระดับความสัมพันธ์และแนวโน้มของการมีอคติกับผลของความเสียหายของกระบวนการออกแบบซอฟต์แวร์ 104
4.6 จำแนกความเสียหายออกเป็นระดับความรุนแรง 106
4.7 วิเคราะห์ข้อมูลที่ได้จากกลุ่มตัวอย่างสำหรับการวิเคราะห์ความน่าจะเป็น 106
4.8 คำนวณและจัดลำดับค่าความเสี่ยงและสร้างตารางความเสี่ยง 110
4.8.1 ค่าความเสี่ยงและตารางความเสี่ยงของกระบวนการรวมรวบและกำหนดความต้องการซอฟต์แวร์ 110
4.8.2 ค่าความเสี่ยงและตารางความเสี่ยงของกระบวนการออกแบบซอฟต์แวร์ 115
4.9 อภิปรายผล 118
4.9.1 การมีอคติกับผลของความเสียหายของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ 118
4.9.2 การมีอคติกับผลของความเสียหายของกระบวนการออกแบบซอฟต์แวร์ 120
บทที่ 5 เครื่องมือสนับสนุนแบบจำลองและการทดสอบเครื่องมือ 123
5.1 แผนภาพยูสเคส 124
5.2 แผนภาพคลาส 124
5.3 สมการในการคำนวณ 126
5.4 เครื่องมือที่ใช้ในการพัฒนา 130
5.5 ส่วนต่อประสานกับผู้ใช้งาน 130
5.6 การทดสอบเครื่องมือ 138
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ 141
6.1 สรุปผลการวิจัย 141
6.2 ข้อจำกัด 142
6.3 ข้อเสนอแนะ 144
รายการอ้างอิง 146
ภาคผนวก 149
ภาคผนวก ก แบบสอบถามที่ใช้ในการเก็บข้อมูล 150
ภาคผนวก ข ผลการทดลองจากกลุ่มตัวอย่างก่อนที่จะนำมาสร้างแบบสอบถาม 167

หน้า

ข.1 ผลการประมาณค่าความเสียหายของซอฟต์แวร์จากการมีอคติในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ 167
ข.2 ผลการประมาณค่าความเสียหายของซอฟต์แวร์จากการมีอคติในกระบวนการออกแบบซอฟต์แวร์ 171
ประวัติผู้เขียนวิทยานิพนธ์ 175

สารบัญตาราง

หน้า

ตารางที่ 3.1 ค่าระดับของจำนวนร้อยละของเหตุการณ์ที่เกิดขึ้น....38
ตารางที่ 3.2 ตัวอย่างการคิดค่าเฉลี่ยของการมีอคติในแต่ละประเภท....38
ตารางที่ 3.3 ตัวอย่างการคิดค่าเฉลี่ยของการมีอคติในแต่ละประเภท....39
ตารางที่ 3.4 ตัวอย่างตัวแปรหุ่นของตัวแปร $DReX_3$ ที่มี g=5 กลุ่ม....42
ตารางที่ 3.5 ตัวอย่างตัวแปรต่าง ๆ ของ Confirmation bais ที่จะนำเข้าวิเคราะห์ความถดถอย...44
ตารางที่ 3.6 ตัวอย่างค่า m ที่จะใช้ในสมการปัวซอง....50
ตารางที่ 3.7 ตัวอย่างลักษณะคำสำคัญที่ใช้แทนอคติแต่ละประเภท....55
ตารางที่ 3.8 ลำดับความสัมพันธ์ของอคติในกระบวนการรวบรวมและกำหนดความต้องการ....56
ตารางที่ 3.9 ลำดับความสัมพันธ์ของอคติในกระบวนการออกแบบ....56
ตารางที่ 3.10 ลักษณะเหตุการณ์ของอคติในกระบวนการรวบรวมและกำหนดความต้องการ....57
ตารางที่ 3.10 ลักษณะเหตุการณ์ของอคติในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)....58
ตารางที่ 3.11 ลักษณะเหตุการณ์ของอคติแต่ละประเภทในกระบวนการออกแบบ....58
ตารางที่ 3.11 ลักษณะเหตุการณ์ของอคติในกระบวนการออกแบบ (ต่อ)....59
ตารางที่ 3.12 ตัวอย่างลักษณะแบบสอบถามจากกระบวนการรวบรวมและกำหนดความต้องการ....60
ตารางที่ 4.1 ค่าเฉลี่ยและค่าที่ปัดเศษแล้วของการมีอคติแต่ละประเภทในกระบวนการรวบรวมและกำหนดความต้องการ....64
ตารางที่ 4.1 ค่าเฉลี่ยและค่าที่ปัดเศษแล้วของการมีอคติแต่ละประเภทในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)....65
ตารางที่ 4.2 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการ....65
ตารางที่ 4.2 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)....66
ตารางที่ 4.3 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Availability ในกระบวนการรวบรวมและกำหนดความต้องการ....67

หน้า

ตารางที่ 4.3 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Availability ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)...68
ตารางที่ 4.4 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Confirmation ในกระบวนการรวบรวมและกำหนดความต้องการ 68
ตารางที่ 4.4 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Confirmation ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ).....69
ตารางที่ 4.5 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน โดยจำแนกออกเป็นแต่ละระดับความรุนแรง ในกระบวนการรวบรวมและกำหนดความต้องการ 70
ตารางที่ 4.5 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน โดยจำแนกออกเป็นแต่ละระดับความรุนแรง ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ).....71
ตารางที่ 4.6 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการ 71
ตารางที่ 4.6 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ).....72
ตารางที่ 4.7 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Availability ในกระบวนการรวบรวมและกำหนดความต้องการ 73
ตารางที่ 4.7 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Availability ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ).....74
ตารางที่ 4.8 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Confirmation ในกระบวนการรวบรวมและกำหนดความต้องการ.....74
ตารางที่ 4.8 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Confirmation ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ).....75
ตารางที่ 4.9 ค่าเฉลี่ยและค่าที่ปัดเศษแล้วของการมีอคติแต่ละประเภทในกระบวนการออกแบบ....76
ตารางที่ 4.9 ค่าเฉลี่ยและค่าที่ปัดเศษแล้วของการมีอคติแต่ละประเภทในกระบวนการออกแบบ (ต่อ).....77
ตารางที่ 4.10 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Anchoring and adjustment ในกระบวนการออกแบบ 77

หน้า

ตารางที่ 4.10 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Anchoring and adjustment ในกระบวนการออกแบบ (ต่อ).....78
ตารางที่ 4.11 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Availability ในกระบวนการออกแบบ.....79
ตารางที่ 4.11 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Availability ในกระบวนการออกแบบ (ต่อ).....80
ตารางที่ 4.12 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Confirmation ในกระบวนการออกแบบ.....80
ตารางที่ 4.12 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Confirmation ในกระบวนการออกแบบ (ต่อ).....81
ตารางที่ 4.13 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน โดยจำแนกออกเป็นแต่ละระดับความรุนแรง ในกระบวนการออกแบบ.....82
ตารางที่ 4.13 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน โดยจำแนกออกเป็นแต่ละระดับความรุนแรง ในกระบวนการออกแบบ (ต่อ).....83
ตารางที่ 4.14 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Anchoring and adjustment ในกระบวนการออกแบบ.....83
ตารางที่ 4.14 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Anchoring and adjustment ในกระบวนการออกแบบ (ต่อ).....84
ตารางที่ 4.15 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Availability ในกระบวนการออกแบบ.....85
ตารางที่ 4.15 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Availability ในกระบวนการออกแบบ (ต่อ).....86
ตารางที่ 4.16 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Confirmation ในกระบวนการออกแบบ.....86
ตารางที่ 4.16 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Confirmation ในกระบวนการออกแบบ (ต่อ).....87
ตารางที่ 4.17 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย.....88
ตารางที่ 4.18 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย.....88
ตารางที่ 4.19 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย.....88
ตารางที่ 4.20 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย.....89

หน้า

ตารางที่ 4.21 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์ 89
ตารางที่ 4.22 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย 90
ตารางที่ 4.23 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย 90
ตารางที่ 4.24 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย 91
ตารางที่ 4.25 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย 91
ตารางที่ 4.26 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์ 91
ตารางที่ 4.27 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย 92
ตารางที่ 4.28 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย 93
ตารางที่ 4.29 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย 93
ตารางที่ 4.30 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย 93
ตารางที่ 4.31 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์ 94
ตารางที่ 4.32 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย 95
ตารางที่ 4.33 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย 95
ตารางที่ 4.34 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย 95
ตารางที่ 4.35 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย 96
ตารางที่ 4.36 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์ 96
ตารางที่ 4.37 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย 97
ตารางที่ 4.38 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย 97
ตารางที่ 4.39 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย 98
ตารางที่ 4.40 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย 98
ตารางที่ 4.41 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์ 98
ตารางที่ 4.42 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย 99
ตารางที่ 4.43 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย 100
ตารางที่ 4.44 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย 100
ตารางที่ 4.45 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย 100
ตารางที่ 4.46 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์ 101
ตารางที่ 4.47 ค่าความเสียหายในแต่ละระดับและค่า m ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ 107
ตารางที่ 4.47 ค่าความเสียหายในแต่ละระดับและค่า m ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ (ต่อ) 108

หน้า

ตารางที่ 4.48 ค่าความเสียหายในแต่ละระดับและค่า m ในกระบวนการออกแบบซอฟต์แวร์..... 108
ตารางที่ 4.48 ค่าความเสียหายในแต่ละระดับและค่า m ในกระบวนการออกแบบซอฟต์แวร์ (ต่อ)..... 109
ตารางที่ 4.49 ค่าความน่าจะเป็นด้วยวิธีปัวซองของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์..... 109
ตารางที่ 4.50 ค่าความน่าจะเป็นด้วยวิธีปัวซองของกระบวนการออกแบบซอฟต์แวร์..... 110
ตารางที่ 4.51 เหตุการณ์ที่เกิดการมีอคติในการทำงานในทุกรูปแบบ..... 111
ตารางที่ 4.52 ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ..... 112
ตารางที่ 4.52 ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ (ต่อ)..... 113
ตารางที่ 4.53 ตารางจัดลำดับความเสี่ยง..... 114
ตารางที่ 4.54 ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ..... 115
ตารางที่ 4.54 ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ (ต่อ)..... 116
ตารางที่ 4.55 ตารางจัดลำดับความเสี่ยง..... 117
ตารางที่ 5.1 สมการทางคณิตศาสตร์ทั้งหมดที่ใช้ในการสร้างเครื่องมือสนับสนุนแบบจำลอง..... 126
ตารางที่ 5.1 สมการทางคณิตศาสตร์ทั้งหมดที่ใช้ในการสร้างเครื่องมือสนับสนุนแบบจำลอง (ต่อ)..... 127
ตารางที่ 5.1 สมการทางคณิตศาสตร์ทั้งหมดที่ใช้ในการสร้างเครื่องมือสนับสนุนแบบจำลอง (ต่อ)..... 128
ตารางที่ 5.1 สมการทางคณิตศาสตร์ทั้งหมดที่ใช้ในการสร้างเครื่องมือสนับสนุนแบบจำลอง (ต่อ)..... 129
ตารางที่ 5.2 การทดสอบการแสดงผลและการทำงานของเครื่องมือ..... 138
ตารางที่ 5.2 การทดสอบการแสดงผลและการทำงานของเครื่องมือ (ต่อ)..... 139
ตารางที่ 5.2 การทดสอบการแสดงผลและการทำงานของเครื่องมือ (ต่อ)..... 139
ตารางที่ ข.1 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย..... 167
ตารางที่ ข.2 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย..... 168
ตารางที่ ข.3 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย..... 168
ตารางที่ ข.4 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย..... 169
ตารางที่ ข.5 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์..... 169
ตารางที่ ข.6 ผลลัพธ์ Correlations ของอคติแต่ละประเภท..... 170

หน้า

ตารางที่ ข.7 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหายของซอฟต์แวร์ 171
ตารางที่ ข.8 ผลลัพธ์ Model Summary ของการประมาณความเสียหายของซอฟต์แวร์ 171
ตารางที่ ข.9 ผลลัพธ์ ANOVA ของการประมาณความเสียหายของซอฟต์แวร์ 172
ตารางที่ ข.10 ผลลัพธ์ Coefficients ของการประมาณความเสียหายของซอฟต์แวร์ 172
ตารางที่ ข.11 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์ 172
ตารางที่ ข.12 ผลลัพธ์ Correlations ของอคติแต่ละประเภท 173

สารบัญภาพ

หน้า

ภาพที่ 2.1 แบบจำลองแนวคิดรูปแบบการตัดสินของโฮการ์ธ 8
ภาพที่ 2.2 Causal Framework ในส่วนของอคติทางการรู้คิด 9
ภาพที่ 2.6 Causal Framework แสดงความสัมพันธ์ระหว่าง Traceability, Task complexity, Cognitive biases และ Maintenance performance 28
ภาพที่ 3.1 แผนภาพกิจกรรมอธิบายภาพรวมของการสร้างแบบจำลองการประเมินความเสี่ยง 32
ภาพที่ 3.2 อธิบายลักษณะของอคติ Anchoring and adjustment 34
ภาพที่ 3.3 อธิบายลักษณะของอคติ Confirmation 36
ภาพที่ 3.4 แบบจำลองจากการประยุกต์ใช้แบบจำลองข้อมูลสองขั้นตอน 36
ภาพที่ 3.5 แผนภาพแสดงภาพรวมของการสร้างแบบสอบถาม 54
ภาพที่ 4.1 แผนภาพกิจกรรมอธิบายภาพรวมของการทดลองและวิเคราะห์ผล 62
ภาพที่ 4.3 กราฟแสดงระดับความสัมพันธ์และแนวโน้มของการมีอคติกับผลของความเสียหายของกระบวนการออกแบบซอฟต์แวร์ 104
ภาพที่ 5.1 แผนภาพยูสเคสของเครื่องมือสำหรับการประเมินความเสี่ยง 124
ภาพที่ 5.2 แผนภาพคลาสของเครื่องมือสำหรับการประเมินความเสี่ยง 125
ภาพที่ 5.3 หน้าจอเริ่มต้นของโปรแกรม Biskuit 130
ภาพที่ 5.4 แสดงผลของการกดปุ่ม Execute ในครั้งแรกภายหลังเปิดโปรแกรมขึ้นมา 133
ภาพที่ 5.5 แสดงการเลือกกระบวนการที่ต้องการคำนวณใน Combo Box 133
ภาพที่ 5.6 แสดงการเลือกระดับของการมีอคติในการทำงานที่ต้องการคำนวณใน Combo Box 134
ภาพที่ 5.7 แสดงการเลือกรูปแบบการคำนวณโดยคิดมูลค่าความเสียหายที่เป็นตัวเงิน 134
ภาพที่ 5.8 การกรอกมูลค่าความเสียหายที่เป็นตัวเงินเพื่อที่จะนำใช้ในการคำนวณ 135
ภาพที่ 5.9 แสดงผลการกดปุ่ม Execute ภายหลังจากเลือกการคำนวณมูลค่าความเสียหายเป็นตัวเงิน 135
ภาพที่ 5.10 แสดงผลการกดปุ่ม Export เพื่อนำผลที่ได้จากการคำนวณมาอยู่ในรูปของ .txt และ .xls 136
ภาพที่ 5.11 แสดงผลการ Export เพื่อนำผลที่ได้จากการคำนวณมาอยู่ในรูปของ .txt 136
ภาพที่ 5.12 แสดงผลการ Export เพื่อนำผลที่ได้จากการคำนวณมาอยู่ในรูปของ .xls 137
ภาพที่ 5.13 แสดงผลการกดปุ่ม Reset 137

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ในปัจจุบันได้มีการนำเอาหลักการของวิศวกรรมซอฟต์แวร์มาใช้ในอุตสาหกรรมซอฟต์แวร์กันอย่างแพร่หลาย โดยเฉพาะในส่วนของวงจรชีวิตของการพัฒนาซอฟต์แวร์ หรือ Software Development Life Cycle (SDLC) ที่เริ่มตั้งแต่ขั้นตอนของการรวบรวมและกำหนดความต้องการ การออกแบบ การพัฒนา การทดสอบ และการบำรุงรักษา ซึ่งมีคำถามเกิดขึ้นว่าวิธีการทางวิศวกรรมซอฟต์แวร์ที่มีอยู่ในแต่ละกระบวนการเหล่านี้เพียงพอที่จะช่วยให้นักวิเคราะห์ระบบ นักออกแบบ หรือผู้ที่มีหน้าที่ในขั้นตอนนี้ตัดสินใจออกแบบเพื่อการพัฒนาซอฟต์แวร์ที่เหมาะสมหรือไม่ [1] ซึ่งจากงานวิจัยในหัวข้อที่ 2.2.1 ถึง 2.2.6 ในงานวิจัยที่เกี่ยวข้อง ได้มีการกล่าวไว้ถึงมนุษยปัจจัย (Human Factor) ที่มีผลต่อกระบวนการพัฒนาซอฟต์แวร์ ซึ่งในโครงการพัฒนาซอฟต์แวร์นั้นมีมนุษย์ซึ่งเป็นปัจจัยสำคัญของความสำเร็จของโครงการ ยกตัวอย่างมนุษยปัจจัยย่อยดังกล่าว เช่น ด้านสังคมและวัฒนธรรม ด้านกายภาพ ด้านพฤติกรรม เป็นต้น สำหรับงานวิจัยนี้มุ่งเน้นเพื่อที่จะศึกษาผลกระทบของมนุษยปัจจัย ในมุมมองของเชิงพฤติกรรม ด้านจิตวิทยาของการมีอคติทางการรู้คิด (Cognitive Bias) โดยในหลายงานวิจัยพบว่ามีคำถามที่น่าสนใจหลายประเด็นเกี่ยวกับอคติทางการรู้คิด เช่น อะไรคืออคติหรือความโน้มเอียง ทำไมถึงมีอคติ หรือความโน้มเอียงเกิดขึ้นมาได้ที่ใดในกระบวนการทางวิศวกรรมซอฟต์แวร์เป็นจุดที่อคติจะเกิดขึ้นมากที่สุด [1] หรือยังไม่มีโมเดลในการอธิบายพฤติกรรมนั้น [2] เป็นต้น และมักจะเน้นไปในเชิงที่ว่า ถ้ามีปัจจัยดังนี้เกิดขึ้นแล้ว จะมีผลต่อโครงการซอฟต์แวร์อย่างไร ซึ่งก็อาจจะมีผลต่อความสำเร็จของโครงการพัฒนาซอฟต์แวร์ได้

ผู้วิจัยจึงได้นำเอาแนวคิดทางด้านอคติทางการรู้คิด นั่นคือ รูปแบบของความโน้มเอียงในการตัดสินใจกับเหตุการณ์ที่เกิดขึ้น ที่นำไปสู่การบิดเบือนการรับรู้และการตัดสินใจที่ไม่ถูกต้อง อันมีผลต่อการตัดสิน (Judgment) เพื่อที่จะปฏิบัติต่อกิจกรรมอย่างใดอย่างหนึ่งในกระบวนการทางวิศวกรรมซอฟต์แวร์ ยกตัวอย่างเช่น การมีอคติในการยืนยันความเชื่อของตน (Confirmation Bias) ที่มีในกระบวนการทดสอบซอฟต์แวร์ โดยผู้พัฒนาซอฟต์แวร์ มักจะทำการทดสอบซอฟต์แวร์ที่ตนเองสร้างขึ้นโดยเชื่อว่าซอฟต์แวร์ที่ตนเองสร้างขึ้นมานั้นมีความถูกต้อง และทำการทดสอบโดยที่พยายามทดสอบให้ซอฟต์แวร์นั้นสามารถทำงานได้มากกว่าที่จะทดสอบเพื่อให้ซอฟต์แวร์ทำงานผิดพลาด ซึ่งอาจจะทำให้เป็นการเพิ่มจำนวนข้อบกพร่องของซอฟต์แวร์ที่ไม่สามารถตรวจหาข้อบกพร่องได้

สำหรับกระบวนการทางวิศวกรรมซอฟต์แวร์ที่สนใจสำหรับงานวิจัยนี้ได้สนใจในกระบวนการของการได้มาซึ่งข้อกำหนดความต้องการ (Requirements Specification) และกระบวนการของการได้มาซึ่งข้อกำหนดการออกแบบ (Design Specification) นั่นคือ กระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ ตามลำดับ ซึ่งเป็นขั้นตอนแรก ๆ ที่มีความสำคัญของกระบวนการพัฒนาซอฟต์แวร์ โดยผู้วิจัยเห็นว่า ขั้นตอนก่อนที่จะให้ได้มาซึ่งผลิตภัณฑ์ของทั้งสองกระบวนการนั้น น่าจะมีอคติทางการรู้คิดเกิดขึ้นได้จากทฤษฎีข้อมูลสองขั้นตอน [3], [4] ที่อธิบายว่า กระบวนการใด ๆ จะต้องผ่านขั้นตอนที่มีข้อผิดพลาด (Error) และขั้นตอนที่มีอคติ (Bias) ซึ่งกระบวนการดังกล่าวเป็นกระบวนการต้นทางของกระบวนการพัฒนาซอฟต์แวร์โดยทั่วไป ที่เป็นตัวกำหนดทิศทางในการพัฒนาซอฟต์แวร์ ซึ่งอาจจะส่งผลต่อคุณภาพ หรือความสำเร็จของโครงการซอฟต์แวร์ในขั้นตอนต่อไปได้ ซึ่งนับว่าปัจจัยดังกล่าวน่าจะเป็นความเสี่ยงที่อาจจะเกิดขึ้นภายในโครงการได้ จึงน่าจะมีการประเมินปัจจัยความเสี่ยงในด้านนี้ด้วยในช่วงต้นของโครงการพัฒนาซอฟต์แวร์ เพื่อช่วยให้ผู้มีส่วนเกี่ยวของในโครงการ หรือผู้มีอำนาจในการตัดสินใจ ตระหนักถึงความเสี่ยงที่อาจจะเกิดขึ้นมาได้ในโครงการพัฒนาซอฟต์แวร์

งานวิจัยนี้จึงมีจุดประสงค์เพื่อสร้างแบบจำลองการประเมินความเสี่ยงในด้านอคติทางการรู้คิดอันมีผลต่อกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ เพื่อให้ได้มาซึ่งข้อกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ เพื่อสร้างข้อกำหนดการออกแบบซอฟต์แวร์ โดยสร้างเครื่องมือเพื่อสนับสนุนแบบจำลองการประเมินความเสี่ยงนั้น เพื่อใช้เป็นข้อมูลประกอบการประเมินความเสี่ยงของโครงการซอฟต์แวร์ โดยสามารถนำผลของการประเมินไปบริหารจัดการความเสี่ยงเพื่อที่จะลดความเสี่ยงจากปัจจัยดังกล่าวที่อาจมีผลกระทบต่อความล้มเหลว หรือคุณภาพของโครงการซอฟต์แวร์ได้

1.2 วัตถุประสงค์ของการวิจัย

งานวิจัยนี้มีวัตถุประสงค์เพื่อสร้างแบบจำลองการประเมินความเสี่ยงของอัตราความเสียหายของซอฟต์แวร์จากการมีอคติทางการรู้คิดในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์และการออกแบบซอฟต์แวร์ และสร้างเครื่องมือการประเมินความเสี่ยงเพื่อสนับสนุนแบบจำลองการประเมินความเสี่ยงดังกล่าว

1.3 ขอบเขตของงานวิจัย

1) สร้างแบบจำลองการประเมินความเสี่ยงอันมีปัจจัยความเสี่ยงมาจากอคติทางการรู้คิดทั้งสามประเภท คือ Anchoring and adjustment, Availability, และ Confirmation

2) ครอบคลุมในกระบวนการทางวิศวกรรมซอฟต์แวร์ตั้งแต่ การรวบรวมและกำหนดความต้องการซอฟต์แวร์ เพื่อจัดทำข้อกำหนดความต้องการซอฟต์แวร์ และการออกแบบซอฟต์แวร์เพื่อจัดทำข้อกำหนดการออกแบบซอฟต์แวร์
3) แบบจำลองนี้แสดงแค่ความเสี่ยงที่อาจจะเกิดขึ้นจากปัจจัยความเสี่ยงจากอคติทางการรู้คิดทั้งสามประเภทกับการพัฒนาซอฟต์แวร์
4) เครื่องมือต้นแบบสำหรับการประเมินความเสี่ยงจะรองรับเพียงแค่แบบจำลองการประเมินความเสี่ยงอันมีปัจจัยความเสี่ยงมาจากอคติทางการรู้คิดทั้งสามประเภท ที่ได้สร้างขึ้นมาเท่านั้น

1.4 ขั้นตอนและวิธีดำเนินการวิจัย

1) ศึกษาและทำความเข้าใจในเรื่องของอคติทางการรู้คิด ปัจจัยทางด้านมนุษย์ และความเสี่ยง
2) ศึกษาและวิเคราะห์ประเด็นปัญหาที่เกิดขึ้นในงานวิศวกรรมซอฟต์แวร์ ที่เกี่ยวข้องกับปัจจัยทางด้านมนุษย์และงานวิจัยที่เกี่ยวข้อง
3) ศึกษาและทำความเข้าใจวิธีการหาค่าความเสี่ยง ความน่าจะเป็น ผลกระทบของความเสี่ยง รวมถึงวิธีในการพัฒนาเครื่องมือให้สามารถรองรับแบบจำลองในการประเมินความเสี่ยงได้
4) ออกแบบและกำหนดลักษณะข้อมูลนำเข้าในการประเมินความเสี่ยง โดยทำการเก็บรวบรวมข้อมูลด้วยการใช้แบบสอบถาม
5) วิเคราะห์ข้อมูลที่ได้มาเพื่อนำไปใช้กับแบบจำลองการประเมินความเสี่ยง
6) เลือกเครื่องมือและพัฒนาเครื่องมือตามที่ได้ออกแบบไว้
7) ทดสอบและประเมินผลวิธีวิจัย และทำการสร้างเครื่องมือที่พัฒนาขึ้นเพื่อสนับสนุนแนวคิดในงานวิจัยนี้
8) จัดทำบทความวิชาการและเผยแพร่บทความวิชาการ
9) สรุปผลแนวทางการวิจัยและข้อเสนอแนะ และจัดทำเล่มโครงงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1) ได้แบบจำลองการประเมินความเสี่ยงของอัตราความเสียหายของการพัฒนาซอฟต์แวร์ ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ที่มีปัจจัยความเสี่ยงมาจากการมีอคติทางการรู้คิด เพื่อให้ผู้จัดการโครงการ นักวิเคราะห์ระบบ นักพัฒนาซอฟต์แวร์ หรือผู้มีส่วนเกี่ยวข้องกับการวางแผนโครงการในการพัฒนาซอฟต์แวร์ได้ใช้แบบจำลองนี้สำหรับการประมาณความเสียหายที่อาจจะเกิดขึ้นกับซอฟต์แวร์

2) ได้เครื่องมือต้นแบบที่สนับสนุนแบบจำลองการประเมินความเสี่ยงของอัตราความเสียหายของการพัฒนาซอฟต์แวร์ ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ที่มีปัจจัยความเสี่ยงมาจากการมีอคติทางการรู้คิด เพื่อให้ผู้จัดการโครงการ นักวิเคราะห์ระบบ นักพัฒนาซอฟต์แวร์ หรือผู้มีส่วนเกี่ยวข้องกับการวางแผนโครงการในการพัฒนาซอฟต์แวร์ได้ใช้เครื่องมือต้นแบบนี้สำหรับการประมาณความเสียหายที่อาจจะเกิดขึ้นกับซอฟต์แวร์
3) แนวคิดในการประเมินความเสี่ยงของอัตราความเสียหายของการพัฒนาซอฟต์แวร์ อันเกิดจากอคติทางการรู้คิดนี้สามารถนำไปประยุกต์ใช้กับกระบวนการอื่น ๆ ของการพัฒนาซอฟต์แวร์ได้

1.6 ผลงานที่ได้รับการตีพิมพ์

ส่วนหนึ่งของงานวิจัยนี้ได้รับการคัดเลือกให้นำเสนอในงานประชุมวิชาการดังนี้

1) หัวข้องานวิจัย "Predicting Software Damage Rate from Cognitive Bias in Software Design Process" ในการประชุมวิชาการระดับนานาชาติ "2013 International Conference on Information, Business and Education Technology (ICIBET 2013)" ซึ่งจัดขึ้น ณ กรุงปักกิ่ง ประเทศจีน ระหว่างวันที่ 14-15 มีนาคม พุทธศักราช 2556
2) หัวข้องานวิจัย "Forecasting Software Damage Rate from Cognitive Bias in Software Requirements Gathering and Specification Process" ในการประชุมวิชาการระดับนานาชาติ "Third IEEE International Conference on Information Science and Technology (ICIST 2013)" ซึ่งจัดขึ้น ณ เมืองหยางโจว มณฑลเจียงซู ประเทศจีน ระหว่างวันที่ 23-25 มีนาคม พุทธศักราช 2556

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องกับงานวิจัยนี้ประกอบไปด้วยเรื่อง อคติทางการรู้คิด การจัดการความเสี่ยง ความน่าจะเป็น แบบจำลองข้อมูลสองขั้นตอน การวิเคราะห์ความถดถอย และตัวแปรหุ่น ซึ่งมีรายละเอียดดังนี้

2.1.1 อคติทางการรู้คิด (Cognitive Bias)

อคติทางการรู้คิด คือ รูปแบบของความโน้มเอียงในการตัดสินใจกับเหตุการณ์ที่เกิดขึ้น ที่นำไปสู่การบิดเบือนการรับรู้การตัดสินใจที่ไม่ถูกต้อง และการตีความที่ไร้เหตุผล หรือสิ่งที่เรียกว่า ความไม่สมเหตุสมผล หรือความไม่มีเหตุผล [5], [6] ซึ่งเรื่องนี้เป็นเรื่องหนึ่งของการคิดและการตัดสินใจ

การคิดมีความหมายที่ครอบคลุมการทำงานของสมองและจิตใจ ในทางจิตวิทยาการคิดมีความหมายคล้ายคลึงกับการใช้เหตุผล หรือการพิจารณาไตร่ตรอง นักจิตวิทยาที่ศึกษาเรื่องของการคิดส่วนใหญ่สนใจศึกษาการคิดในความหมายดังกล่าว โดยสนใจการคิดที่เกิดขึ้นเวลาที่พยายามแก้ไขปัญหา ตัดสินข้อเท็จจริง พิจารณาข้อได้เปรียบและข้อเสียเปรียบประกอบการตัดสินใจ

2.1.1.1 การใช้เหตุผลและการตัดสินใจ

การใช้เหตุผลเป็นการหาข้อสรุปจากข้ออ้างบางอย่าง ซึ่งบรรดานักจิตวิทยาเชื่อว่า คนเรามีกระบวนการการใช้เหตุผลตามหลักตรรกะ ด้วยเหตุนี้นักคณิตศาสตร์ในศตวรรษที่ 19 ชื่อ George Boole จึงได้เขียนสรุปกฎเกณฑ์ทางตรรกะขึ้นไว้ ซึ่งหลักตรรกะเหล่านี้ได้กลายเป็นพื้นฐานของขั้นตอนการทำงานของคอมพิวเตอร์ในเวลาต่อมา อย่างไรก็ตาม มีการวิพากษ์วิจารณ์กันว่า มนุษย์จะไม่ได้คิดเป็นเหตุเป็นผลตามหลักตรรกะเสมอไป อาจจะเป็นไปได้ว่า กฎเกณฑ์เชิงตรรกะบอกว่าคนเราควรคิดอย่างไร แต่ไม่ได้สอดคล้องกับวิธีการคิดจริง ๆ ของมนุษย์เสมอไป สำหรับในการแก้ปัญหา มนุษย์มีกระบวนการการใช้เหตุผล 2 ลักษณะ คือ การใช้เหตุผลเชิงนิรนัย (Deductive Reasoning) และการใช้เหตุผลเชิงอุปนัย (Inductive Reasoning) [7]

1) การใช้เหตุผลเชิงนิรนัย เป็นการที่ผู้ใช้เหตุผลพยายามพิจารณาว่า จะสามารถหาข้อสรุปอะไรได้บ้างจากข้ออ้าง (Premises) ที่มีอยู่ โดยความถูกต้องของข้อสรุปขึ้นอยู่

กับว่า ข้อสรุปนั้นนำมาจากข้ออ้างที่ให้ไว้ตามหลักตรรกวิทยา หรือไม่ ส่วนการที่ข้ออ้างนั้นจะเป็นความจริงหรือไม่ หรือข้อสรุปนั้นจะมีความเป็นไปได้หรือไม่ จะไม่เกี่ยวกับการสรุปตามข้ออ้างตามหลักตรรกะหรือไม่ ดังนั้นแม้บางครั้ง อาจจะได้ข้อสรุปแปลก ๆ แต่ถ้าเป็นไปตามหลักตรรกะก็ถือว่าถูกต้อง วิธีตามหลักตรรกวิทยานี้เรียกว่าซิลโลจิซึมส์ (Syllogisms) จากการทดลอง ผู้เข้าร่วมการทดลองมีการประเมินผิดพลาดประมาณร้อยละ 70 ถึง 80 โดยความผิดพลาดดังกล่าวเกิดจากมนุษย์มักจะตัดสินข้อสรุปที่มีข้ออ้างที่ให้ไว้ ตามความรู้เดิม หรือตามความรู้สึกของตนมากกว่าการสรุปตามหลักตรรกะ ที่จะต้องอาศัยเพียงข้ออ้างที่มีอยู่ ด้วยเหตุนี้ ข้อสรุปที่แม้จะผิดหลักตรรกวิทยา แต่ฟังดูคุ้นเคย หรือมีความเป็นไปได้ จึงมักจะถูกตัดสินว่าถูกต้อง ไม่ว่าข้อสรุปนั้นจะมาจากข้ออ้างที่ถูกต้องตามที่ได้ให้ไว้หรือไม่

2) การใช้เหตุผลเชิงอุปนัย คือการที่ใช้เหตุผลจากบางสิ่งบางอย่างแล้วสรุปเป็นหลักการทั่วไป การคิดแบบนี้เป็นหลักการหัวใจของวิทยาศาสตร์ นั่นคือ การหาหลักการทั่วไปที่จะสามารถอธิบายเหตุการณ์ต่าง ๆ ได้ การทำเช่นนี้นักวิทยาศาสตร์ต้องตั้งสมมติฐาน และทำการทดสอบสมมติฐานนั้น ในการตั้งคำถามว่า จะมีเหตุการณ์นั้น ๆ เกิดขึ้นมากน้อยแค่ไหน เพื่อพิจารณาว่า เหตุการณ์นั้น ๆ จะมีแนวโน้มที่จะเกิดขึ้นอีกในอนาคตได้ ในลักษณะนี้การตั้งสมมติฐานของมนุษย์ทั้งในเชิงของวิทยาศาสตร์และสิ่งที่เกิดขึ้นในชีวิตประจำวัน ล้วนขึ้นอยู่กับ การประมาณความถี่ (Frequency Estimates)

2.1.1.2 การประมาณความถี่ (Frequency Estimates)

มนุษย์เรามักจะประมาณความถี่โดยวิธีง่าย ๆ คือ พยายามที่คิดถึงกรณีเฉพาะเจาะจงที่เกี่ยวข้องกับเรื่องที่เราตัดสินใจ เช่น ตัวอย่างซอฟต์แวร์ที่ต้องการความมั่นคงสูง หากตัวอย่างดังกล่าวผุดขึ้นมาในจิตใจได้อย่างง่าย ๆ เราจะสรุปว่า สถานการณ์นั้น ๆ เป็นเรื่องปกติ แต่ถ้าตัวอย่างผุดขึ้นมาในจิตใจได้อย่างช้า ๆ หรือต้องใช้ความพยายามในการนึก เราจะสรุปว่า สถานการณ์นั้น ๆ เป็นเรื่องผิดปกติ วิธีการคิดแบบนี้เรียกว่า Availability heuristic

ความผิดพลาดในหลายสถานการณ์ การตัดสินด้วย Availability heuristic อาจนำไปสู่ความผิดพลาดที่ร้ายแรงได้ เช่น โอกาสที่เราจะละเลยในการเข้ารหัสข้อมูลในจุดที่มีความล่อแหลมมีมากน้อยแค่ไหน ถ้าเราตัดสินใจโดยดูแต่ความน่าจะเป็น (Probability) เราจะคิดว่าในจุดนี้น่าจะมีการเข้ารหัสหรือการเข้ารหัสเพียงแค่จุดนี้ก็น่าจะเพียงพอ การประมาณการแบบนี้ถ้ามีใช้ Availability heuristic และการตัดสินใจของนักออกแบบระบบอาจคิดแค่ว่าให้ใส่การเข้ารหัสข้อมูลในการเข้าถึงระบบครั้งแรก

ก็เพียงพอแล้ว อาจจะทำให้นักออกแบบประมาณสถานการณ์ผิดไป จึงไม่ได้ระมัดระวังในจุดอื่น ๆ ให้ครบถ้วน

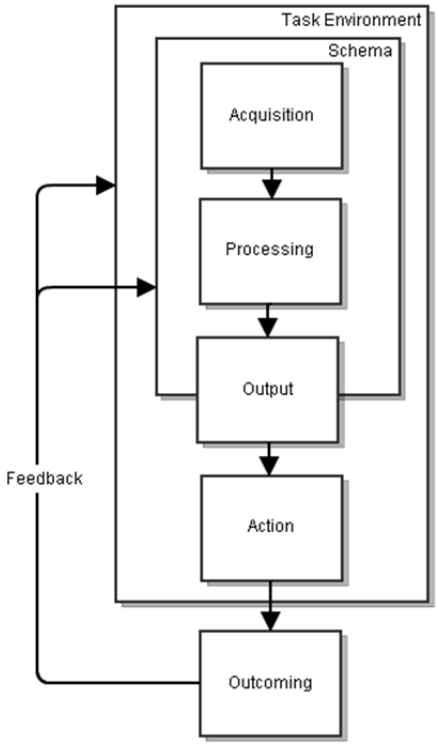
2.1.1.3 ประเภทของอคติที่ใช้ในการแก้ปัญหา

สำหรับอคติทางการรู้คิดที่พบในระหว่างการแก้ปัญหาต่าง ๆ นั้น มากที่สุด ได้แก่ Anchoring and adjustment, Availability และ Confirmation [8] ซึ่งมีคำจำกัดความดังนี้

1) อคติการยึดติดและการปรับเปลี่ยน (Anchoring and Adjustment Bias) – อคตินี้มาจากการประมาณการเบื้องต้นของมนุษย์เกี่ยวกับปัญหาต่าง ๆ โดยอ้างจากประสบการณ์ของตนและการปรับเปลี่ยนการประมาณการเบื้องต้นเหล่านั้น ก่อนที่จะได้มาเป็นขั้นสุดท้ายของการแก้ปัญหาที่เหมาะสมกว่า [9], [10] เช่น การประมาณการความต้องการของระบบว่าควรจะต้องมีความสามารถอย่างไรบ้างในเบื้องต้น และปรับแต่งการประมาณนั้นให้เหมาะสมกับระบบที่จะพัฒนา เป็นต้น
2) อคติการหามาได้ง่าย (Availability Bias) – อคตินี้แสดงให้เห็นว่ามนุษย์ได้รับอิทธิพลจากข้อมูลที่ง่ายต่อการระลึกถึงและมีอคติจากข้อมูลที่อยู่ในการเผยแพร่ของสื่อ หรือข้อมูลที่เพิ่งเกิดขึ้นมาเมื่อไม่นานนี้ [9] เช่น ในการกำหนดความสามารถของซอฟต์แวร์ อาจจะคำนึงถึงสถาปัตยกรรมระบบที่กำลังเป็นกระแสอยู่ในปัจจุบัน เป็นต้น
3) อคติการยืนยันความเชื่อของตน (Confirmation Bias) – อคตินี้แสดงให้เห็นว่ามนุษย์มักจะมุ่งเน้นที่ข้อมูลที่สอดคล้องกับแนวคิดที่ไตร่ตรองไว้ก่อนแล้วของพวกเขา ขณะที่พวกเขาไม่สนใจข้อมูลที่ไม่สอดคล้องกับแนวความคิดที่ผ่านมาของพวกเขา [11] เช่น ในการเลือกวิธีการแก้ปัญหาที่จะใช้ในการออกแบบซอฟต์แวร์ อาจจะเลือกมาจากข้อมูลที่ตนเองเชื่อ หรือข้อมูลที่ตนเองมีอยู่ เป็นต้น

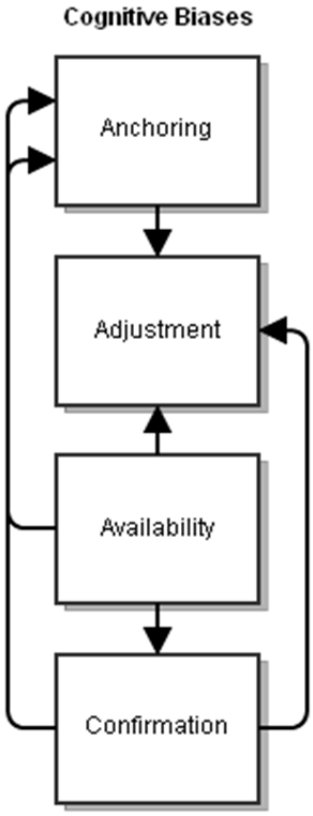
2.1.1.4 ความสัมพันธ์ระหว่างอคติ

โดยจากแบบจำลองแนวคิดรูปแบบการตัดสินของโฮการ์ธ (Hogarth’s Concept Model of Judgment) ได้อธิบายแสดงความสัมพันธ์ของอคติทางการรู้คิดทั้งสามประเภท แต่ไม่รวม Confirmation ไว้ในแผนภาพนี้ ดังภาพที่ 2.1



ภาพที่ 2.1 แบบจำลองแนวคิดรูปแบบการตัดสินของโฮการ์ธ [12]

อคติเกี่ยวกับการตัดสิน (Judgmental Bias) จัดเป็นอคติทางการรู้คิด ซึ่งภายใน Judgmental bias แบ่งออกตามแบบจำลองดังกล่าว จะบอกถึงลำดับขั้นตอนของกรอบแนวคิดของการตัดสินใจว่ามีลำดับขั้นตอนอย่างไร โดยจะมีอคติที่เราสนใจปรากฏอยู่ในขั้นตอนของแบบจำลองโดย อคติ Availability อยู่ในขั้นตอนของ Acquisition biases และอคติ Anchoring and adjustment อยู่ในขั้นตอนของ Information processing biases [12] สำหรับอคติ Confirmation ได้มีงานวิจัยของ Kannan Mohan และ Radhika Jain ในปี 2008 กล่าวไว้ โดยนำเสนอ Causal Framework ของอคติทางการรู้คิด [8] ซึ่งเป็นกล่าวถึงความสัมพันธ์ของอคติทางการรู้คิดแต่ละตัวที่เราสนใจ ดังภาพที่ 2.2



ภาพที่ 2.2 Causal Framework ในส่วนของอคติทางการรู้คิด [8]

จากแผนภาพ Causal Framework ในส่วนของอคติทางการรู้คิดดังกล่าวจะชี้ให้เห็นถึงความสัมพันธ์โดย จะเห็นได้ว่า Availability กับ Confirmation จะถูกใช้เมื่อมี Anchoring เกิดขึ้น นั่นคือเมื่อมีการประมาณการในเบื้องต้น (Anchoring) เกิดขึ้น โดยอาจประมาณจากความคุ้นเคย หรือจากจุดที่เป็นที่รู้จัก หรือจุดอ้างอิง จะต้องมีการระลึกถึงข้อมูลที่นึกออกได้ง่ายหรือการหามาได้ง่าย (Availability) หรือเหตุการณ์ที่เพิ่งจะเกิดขึ้น กับการยืนยันถึงข้อมูล หรือแนวความคิดที่ระลึกมานั้นว่าถูกต้อง (Confirmation) ซึ่งการประมาณการเบื้องต้นนี้จะเป็นตัวตั้งต้นของการปรับเปลี่ยน (Adjustment) [13], [14] ซึ่งการปรับเปลี่ยน ทำเพื่อให้ได้การแก้ปัญหา หรือตอบโจทย์นั้น ๆ ซึ่งในขั้นตอนนี้อาจจะมีการยืนยัน (Confirmation) อีกครั้งว่าการปรับเปลี่ยนของเรานั้นถูกต้อง เหมาะสมหรือไม่

จากทฤษฎีนี้ได้กล่าวถึงกระบวนการคิดและการตัดสินใจของมนุษย์ ในการแก้ปัญหาใด ๆ โดยในการตัดสินใจในการแก้ปัญหาต่าง ๆ ของมนุษย์ในแต่ละครั้ง ก็อาจจะมีอคติเกิดขึ้นได้ เรียกว่าเป็นอคติที่พบในการแก้ปัญหา และยังได้กล่าวถึงความสัมพันธ์ของอคติแต่ละตัว ว่ามีความสัมพันธ์กัน

อย่างไร ส่วนปัญหาใด ๆ นั้น ก็รวมไปถึงปัญหาทางด้านวิศวกรรมซอฟต์แวร์ด้วย ดังนั้นจึงนำเอาอคติที่พบในการแก้ปัญหานี้มาเพื่อศึกษาว่า มีความเกี่ยวข้องในด้านใดกับปัญหาทางด้านวิศวกรรมซอฟต์แวร์บ้าง

2.1.2 การจัดการความเสี่ยง (Risk Management)

ความเสี่ยง (Risk) คือผลกระทบของความไม่แน่นอนตามเป้าหมาย หรือวัตถุประสงค์ โดยผลกระทบ (Effects) นั้นคือความเบี่ยงเบนจากสิ่งที่คาดหวังไว้ ทั้งในแง่ลบและ/หรือแง่บวก โดยเป้าหมายหรือวัตถุประสงค์ (Objectives) สามารถมีลักษณะที่แตกต่างกัน เช่น เป้าหมายทางการเงิน เป้าหมายทางสุขภาพ เป้าหมายทางความปลอดภัย และเป้าหมายทางสิ่งแวดล้อม อีกทั้งยังสามารถนำไปใช้ในระดับที่แตกต่างกัน เช่น ระดับกลยุทธ์ ระดับองค์กร ระดับโครงการ ระดับของผลิตภัณฑ์ และระดับกระบวนการ

ความเสี่ยงเป็นลักษณะที่มักจะอ้างอิงถึงเหตุการณ์ (Events) ที่อาจเกิดขึ้น และผลกระทบ (Consequences) หรือผลลัพธ์ ที่เกิดขึ้น ซึ่งไม่ได้กำหนดว่าจะให้มีหน่วยของผลกระทบของเหตุการณ์เป็นอะไร ซึ่งก็อาจจะเป็นได้ทั้งจำนวนครั้ง จำนวนครั้งต่อหนึ่งหน่วยเวลา หรืออื่น ๆ ซึ่งผลกระทบนี้รวมไปถึงการเปลี่ยนแปลงของสถานการณ์ ซึ่งก็อาจจะมีหน่วยเป็น การเปลี่ยนแปลง 1 ครั้งในรอบทุก 1 ปี ซึ่งก็เรียกได้ว่าเป็นอัตราของการเปลี่ยนแปลง โดยความเสี่ยงมักจะแสดงในแง่ของการคูณกันของผลกระทบของเหตุการณ์ รวมถึงการเปลี่ยนแปลงของสถานการณ์ กับความเป็นไปได้ (Likelihood) ที่สัมพันธ์กันของเหตุการณ์ที่เกิดขึ้น (Occurrence) และสำหรับความไม่แน่นอน (Uncertainty) เป็นสถานะของความขาดตกบกพร่องของข้อมูลที่เกี่ยวข้องกับความรู้ความเข้าใจ การเกิดขึ้น หรือความน่าจะเป็นของเหตุการณ์ [15]

จากข้อความข้างต้น สามารถนิยามได้ดังนี้ [16]

ความเสี่ยง = ความเสียหายจากการเกิดเหตุการณ์ × ความน่าจะเป็นของการเกิดเหตุการณ์ (2.1)

ความเสี่ยงในงานวิจัยนี้จะใช้เป็นค่าความคาดหวัง (Expected Value) ที่คาดว่าจะมีผลต่อโครงการ จำเป็นที่จะต้องมีการดำเนินการกับความเสี่ยงอันอาจจะเกิดขึ้น ทั้งในแง่บวกและแง่ลบ โดยเฉพาะอย่างยิ่งในแง่ลบ อันอาจจะก่อให้เกิดความเสียหาย การดำเนินการกับความเสี่ยงดังกล่าวอาจจะกระทำได้โดย การหลีกเลี่ยงไม่ทำกิจกรรมที่เป็นสาเหตุที่จะก่อให้เกิดความเสียหาย หรือลด

ปัจจัยอันเป็นสาเหตุของความเสี่ยงนั้น หรือเพิ่มความระมัดระวัง เพิ่มความตระหนักในการกระทำกิจกรรมดังกล่าว นั่นคือจะต้องมีการบริหารจัดการความเสี่ยง และการที่จะบริหารจัดการความเสี่ยงได้นั้น จะต้องมีความรู้ความเข้าใจเกี่ยวกับความเสี่ยงนั้น ๆ จากการประเมินความเสี่ยง

การประเมินความเสี่ยง (Risk Assessment) เป็นการรวมทุกกิจกรรมของ การระบุความเสี่ยง (Risk Identification) การวิเคราะห์ความเสี่ยง (Risk Analysis) และการประเมินค่าความเสี่ยง (Risk Evaluation) ตามลำดับ [15]

1) การระบุความเสี่ยง (Risk Identification) – คือกระบวนการในการหา การแสดงให้เห็นถึงความสำคัญ และการอธิบายความเสี่ยง โดยที่จะเกี่ยวข้องกับการระบุแหล่งที่มาของความเสี่ยง (Risk Sources) เหตุการณ์ สาเหตุ และผลกระทบที่อาจเกิดขึ้นของความเสี่ยง แล้วยังเกี่ยวข้องประวัติข้อมูล ทฤษฎีวิเคราะห์ ความคิดเห็นของผู้เชี่ยวชาญ และความต้องการของผู้มีส่วนร่วม
2) การวิเคราะห์ความเสี่ยง (Risk Analysis) – คือกระบวนการเพื่อที่จะเข้าใจธรรมชาติของความเสี่ยง และเพื่อระบุระดับของความเสี่ยง (Level of Risk) โดยมีพื้นฐานสำหรับการประเมินค่าความเสี่ยงและการตัดสินใจเกี่ยวกับการปฏิบัติต่อความเสี่ยง (Risk Treatment) นอกจากนี้ยังรวมไปถึงการประเมินความเสี่ยง
3) การประเมินค่าความเสี่ยง (Risk Evaluation) – คือกระบวนการของการเทียบผลของการวิเคราะห์ความเสี่ยงและเงื่อนไขความเสี่ยง (Risk Criteria) เพื่อตรวจสอบว่ามีความเสี่ยงและ/หรือขนาดของความเสี่ยงว่าเป็นที่ยอมรับได้หรือไม่ โดยช่วยในการตัดสินใจในการเลือกปฏิบัติของความเสี่ยง

2.1.3 ความน่าจะเป็น (Probability)

ในส่วนนี้จะกล่าวถึงเรื่องของความน่าจะเป็น ซึ่งจะใช้เป็นตัวแปรหนึ่งในสมการความเสี่ยง โดยจะกล่าวถึงส่วนของนิยามความน่าจะเป็น และการเลือกใช้วิธีการแจกแจงความน่าจะเป็นแบบไม่ต่อเนื่อง หรือตัวแปรสุ่มวิยุต ที่เป็นการแจกแจงความน่าจะเป็นแบบของปัวซอง

2.1.3.1 นิยามความน่าจะเป็น

ความน่าจะเป็น (Probability) หรือในบางครั้งในภาษาอังกฤษก็มักจะใช้คำว่า “Likelihood” คือ โอกาสของการเกิดขึ้นของบางสิ่ง ในศัพท์เฉพาะทางของการจัดการความเสี่ยง ความน่าจะเป็นถูกใช้ในการอ้างอิงถึงโอกาสของการเกิดขึ้นของบางสิ่ง ไม่ว่าจะเป็นการกำหนด การวัด หรือการระบุเชิงปรนัย หรือเชิงอัตนัย เชิงคุณภาพ หรือเชิงปริมาณ ซึ่งอธิบายโดยทั่วไปและในเชิงคณิตศาสตร์ เช่น ความน่าจะเป็น หรือความถี่ในช่วงเวลาที่กำหนด เป็นต้น โดยคำว่า “Probability” มักจะถูกใช้ไปในการตีความหมายเชิงคณิตศาสตร์ ส่วนคำว่า “Likelihood” ถูกใช้ในการตีความหมายในเชิงที่กว้างกว่าคำว่า “Probability” [15] ซึ่งมีความสัมพันธ์กับความเสี่ยง ที่ความเสี่ยงมักจะแสดงไว้ในแง่ของการรวมกันของผลกระทบของเหตุการณ์ รวมถึงการเปลี่ยนแปลงของสถานการณ์ และความเป็นไปได้ที่สัมพันธ์กันของการเกิดขึ้นของเหตุการณ์

ความน่าจะเป็นในความหมายเชิงคณิตศาสตร์ เกี่ยวกับความเป็นไปได้ โดยศึกษาเรื่องที่อาจจะเกิดขึ้นหรือไม่เกิดขึ้น โดยทฤษฎีความน่าจะเป็น ได้จำกัดความน่าจะเป็นว่าอยู่ตั้งแต่ 0 ถึง 1 โดยที่ 1 หมายถึงเหตุการณ์นั้นจะเกิดขึ้นแน่นอน และโดยที่ 0 หมายถึงเหตุการณ์นั้นไม่มีโอกาสที่จะเกิดขึ้น [17]

การแจกแจงความน่าจะเป็น แสดงค่าความน่าจะเป็นของตัวแปรสุ่มวิยุต หรือเป็นฟังก์ชันที่กำหนดค่าความน่าจะเป็น เป็นส่วนพื้นที่ใต้เส้นโค้ง เรียกว่า การแจกแจงความน่าจะเป็นแบบไม่ต่อเนื่อง (Discrete Probability Distribution) โดยที่ ตัวแปรสุ่มวิยุต (Discrete Random Variables) เป็นตัวแปรสุ่มที่มีค่าไม่ต่อเนื่อง กำหนดค่าเป็น x1, x2, x3, ... เช่น จำนวนเหรียญที่ออกหัวในการโยนเหรียญ 5 เหรียญ จำนวนครั้งที่ทดลองจนได้ผลสำเร็จ หรือเหตุการณ์ที่ต้องการ เป็นต้น [18]

2.1.3.2 การแจกแจงความน่าจะเป็นแบบปัวซอง (Poisson Probability Distribution)

การแจกแจงความน่าจะเป็นแบบปัวซอง เป็นการแจกแจงความน่าจะเป็นตัวแปรสุ่มแบบไม่ต่อเนื่องชนิดหนึ่ง โดยเป็นการแจกแจงที่อธิบายถึงจำนวนครั้งของเหตุการณ์หรือจำนวนสิ่งที่สนใจที่เกิดขึ้นในช่วงเวลาที่กำหนดหรือในพื้นที่ที่กำหนด โดยในการทดลองเชิงสุ่มในช่วงเวลาหนึ่ง มีสมบัติว่าโอกาสที่จะเกิดเหตุการณ์ 1 ครั้งในช่วงเวลาสั้น ๆ แต่ทราบค่าเฉลี่ยของจำนวนครั้งที่เหตุการณ์นั้นเกิดขึ้นในช่วงเวลาที่กำหนด [19]

ถ้าให้ X เป็นจำนวนเหตุการณ์หรือจำนวนสิ่งที่สนใจที่เกิดขึ้นในช่วงเวลาหนึ่ง หรือในพื้นที่หนึ่ง X จะเป็นตัวแปรสุ่มแบบปัวซอง ที่มีพารามิเตอร์ μ โดยมีฟังก์ชันความน่าจะเป็นของ X เป็นดังนี้

$$P(x;\mu) = P(X = x) = \frac{e^{-\mu}\mu^{x}}{X!}; x = 0,1,2,\dots,n \qquad (2.2)$$

โดยที่ e ≈ 2.71828...

μ = จำนวนสิ่งที่สนใจที่เกิดขึ้นโดยเฉลี่ยในช่วงเวลาหรือในพื้นที่หรือขอบเขตที่กำหนดค่าเฉลี่ยและค่าความแปรปรวนของตัวแปรสุ่มแบบปัวซอง

ค่าเฉลี่ย $E(X) = \mu$

ค่าแปรปรวน $V(X) = \sigma^2 = \mu$

สำหรับการคำนวณหาความน่าจะเป็นแบบปัวซอง ยังสามารถหาได้จากตารางแจกแจงความน่าจะเป็นแบบปัวซองได้อีกด้วย การแจกแจงปัวซอง ตั้งชื่อตามนักคณิตศาสตร์ชาวฝรั่งเศสชื่อ Simeon Denis Poisson โดยการแจกแจงปัวซองมีลักษณะดังนี้ [19]

1) เหตุการณ์ที่เกิดขึ้นเป็นอิสระต่อกัน โดยที่เหตุการณ์ที่เกิดขึ้นในช่วงใดช่วงหนึ่ง หรือเวลาใดเวลาหนึ่ง จะไม่มีผลต่อความน่าจะเป็นของการเกิดเหตุการณ์ในช่วงเวลาอื่น ๆ
2) ในช่วงใดช่วงหนึ่งอาจมีจำนวนเหตุการณ์ที่เกิดขึ้นไม่จำกัด
3) ความน่าจะเป็นของการเกิดเหตุการณ์ในช่วงใด ๆ เป็นสัดส่วนกับความยาวของช่วงทั้งหมด เช่น โดยเฉลี่ยในครึ่งชั่วโมง จะมีโทรศัพท์มา 4 ครั้ง ดังนั้นใน 10 นาทีจะมีโทรศัพท์มาเฉลี่ย 4/3 ครั้ง
4) ในช่วงเวลาสั้น ๆ ความน่าจะเป็นที่จะเกิดเหตุการณ์ 1 ครั้ง มีค่าน้อย ความน่าจะเป็นที่จะเกิดเหตุการณ์มากกว่า 1 ครั้งมีค่าเป็น 0 เช่น ในช่วงเวลา 1 วินาที โอกาสที่จะมีโทรศัพท์มา 1 ครั้ง ในช่วงนี้มีค่าน้อย และโอกาสที่จะมีโทรศัพท์มาเกินกว่า 1 ครั้ง มีค่าเป็น 0 (แต่โดยเฉลี่ยในครึ่งชั่วโมงจะมีโทรศัพท์มา 4 ครั้ง)

2.1.4 แบบจำลองข้อมูลสองขั้นตอน (Two Stage Information Model)

เป็นแบบจำลองที่การกล่าวถึงเรื่องการเปลี่ยนแปลงไปของข้อมูลในแต่ละขั้นตอน ซึ่งแบบจำลองดังกล่าวถูกใช้อยู่ในงานวิจัยของ Andrew P. Snow และ Mark Keil หลายงาน [2], [3], [4] โดยประยุกต์เอาเรื่องแบบจำลองข้อมูลสองขั้นตอนไปใช้ในงานการจัดทำรายงานสถานะของโครงการซอฟต์แวร์ ซึ่งสามารถใช้เป็นแบบจำลองเพื่อตามรอยรายงานสถานะของโครงการได้ว่าก่อนที่จะได้รายงานสถานะของโครงการมานั้น ต้องผ่านขั้นตอนใดบ้าง มีข้อมูลนำเข้าเป็นอะไร ข้อมูลส่งออกเป็นอะไร มีข้อผิดพลาด และมีอคติเกิดขึ้นในขั้นตอนใดได้บ้าง ซึ่งเป็นแบบจำลองที่อธิบายการเกิดขึ้นของอคติ และในงานวิจัยนี้สนใจเฉพาะการเกิดอคติเท่านั้น ดังที่จะกล่าวต่อไปในงานวิจัยที่เกี่ยวข้องหัวข้อที่ 3.4

2.1.5 การวิเคราะห์ความถดถอย (Regression Analysis)

เป็นการศึกษาถึงความสัมพันธ์ของตัวแปรตั้งแต่ 2 ตัวขึ้นไป โดยมีวัตถุประสงค์ที่จะประมาณหรือพยากรณ์ค่าของตัวแปรตัวหนึ่งจากตัวแปรอื่น ๆ ที่มีความสัมพันธ์กับตัวแปรที่ต้องการพยากรณ์ โดยจะต้องมีการกำหนดหรือทราบค่าตัวแปรอื่น ล่วงหน้า เช่น ถ้าทราบความสัมพันธ์ระหว่างยอดขายกับค่าโฆษณาแล้ว จะทำให้สามารถประมาณหรือพยากรณ์ยอดขายเมื่อกำหนดหรือทราบงบประมาณในการโฆษณา และจะศึกษาถึงการเปลี่ยนแปลงของยอดขายเมื่องบประมาณในการโฆษณาเปลี่ยนแปลงไป โดยอาศัยหลักการของการวิเคราะห์ความถดถอย [19] ซึ่งแบ่งออกเป็น 2 ประเภท คือ

2.1.5.1 การวิเคราะห์ความถดถอยและสหสัมพันธ์อย่างง่าย (Simple Regression Analysis and Correlation)

เป็นการศึกษาถึงความสัมพันธ์ระหว่างตัวแปร 2 ตัว หรือลักษณะที่สนใจศึกษา 2 ลักษณะ โดยที่ต้องทราบค่าของตัวแปรตัวหนึ่งหรือต้องกำหนดค่าของตัวแปรหนึ่งไว้ล่วงหน้า โดยตัวแปรที่ต้องกำหนดค่าไว้ล่วงหน้าเรียกว่า ตัวแปรอิสระ (Independent Variable) จะแทนด้วยสัญลักษณ์ X ส่วนที่ต้องขึ้นกับตัวแปรอิสระจะเรียกว่า ตัวแปรตาม (Dependent Variable) จะแทนด้วยสัญลักษณ์ Y

วัตถุประสงค์ของการวิเคราะห์ความถดถอยและสหสัมพันธ์

1) เพื่อศึกษาความสัมพันธ์ระหว่างตัวแปรว่ามีความสัมพันธ์กันมากน้อยเพียงใด ถ้า X และ Y มีความสัมพันธ์กันมาก แสดงว่า ถ้า X มีค่าเปลี่ยนแปลงไป จะมีผลกระทบต่อค่าของ Y เป็นอย่างมาก
2) ใช้ความสัมพันธ์ที่วิเคราะห์ได้มาประมาณค่าหรือพยากรณ์ค่า Y ในอนาคต เมื่อกำหนดค่า X

สำหรับการหารูปแบบความสัมพันธ์ระหว่างตัวแปร Y และ X นั้น ในขั้นแรกจะนำเอาข้อมูลของตัวแปรทั้งสองมาเขียนกราฟแสดงความสัมพันธ์ ซึ่งจะเรียกกราฟนี้ว่า แผนภาพการกระจาย (Scatter Diagram) หลังจากนั้นทำการพิจารณาจากแผนภาพการกระจายว่าความสัมพันธ์ของตัวแปรทั้งสองจะอยู่ในรูปแบบใด เช่น เส้นโค้ง เส้นตรง พาราโบลา เป็นต้น โดยที่จะต้องเขียนความสัมพันธ์ให้อยู่ในรูปแบบทางคณิตศาสตร์ได้

ในงานวิทยานิพนธ์นี้จะศึกษาเฉพาะความสัมพันธ์ของตัวแปร X และ Y ในรูปเชิงเส้นหรือเส้นตรงเท่านั้น จึงเรียกเป็นการวิเคราะห์ความถดถอยอย่างง่ายที่ความสัมพันธ์อย่างง่ายที่ความสัมพันธ์ของตัวแปรอยู่ในรูปเชิงเส้นว่า การวิเคราะห์ความถดถอยเชิงเส้นอย่างง่าย (Simple Linear Regression Analysis) สามารถแสดงความสัมพันธ์ในรูปสมการเชิงเส้น ดังนี้

$$Y_i = \beta_0 + \beta_1 X_i + e_i \; ; i = 1,2, \dots, N \qquad (2.3)$$

โดยที่ Y = ตัวแปรตาม เนื่องจากค่าของ Y ขึ้นอยู่กับค่าของ X

X = ตัวแปรอิสระ

β_0 = ส่วนตัดแกน Y หรือ คือค่าของ Y เมื่อ X มีค่าเป็นศูนย์

e = ความคลาดเคลื่อนอย่างสุ่ม (Random Error)

β_1 = ความชัน (Slope) ของเส้นตรง ซึ่งเป็นค่าที่แสดงถึงอัตราการเปลี่ยนแปลงของ Y เมื่อ X เปลี่ยนไป 1 หน่วย และจะเรียก β_1 ว่าสัมประสิทธิ์ความถดถอย (Regression Coefficient) โดยค่าของ β_1 อาจจะเป็นได้ดังนี้

1) $\beta_1 > 0$ แสดงว่า X และ Y มีความสัมพันธ์ในทางเดียวกันคือ ถ้า X เพิ่ม Y จะเพิ่มด้วย แต่ถ้า X ลดลง Y ก็จะลดลงด้วย
2) $\beta_1 < 0$ แสดงว่า X และ Y มีความสัมพันธ์ในทางตรงข้ามคือ ถ้า X เพิ่มขึ้น Y ลดลง แต่ถ้า X ลดลง Y จะเพิ่มขึ้น
3) β_1 มีค่าเข้าใกล้ศูนย์ แสดงว่าค่า X และ Y มีความสัมพันธ์กันน้อย
4) $\beta_1 = 0$ แสดงว่า X และ Y ไม่มีความสัมพันธ์กัน

2.1.5.2 การวิเคราะห์ความถดถอยและสหสัมพันธ์เชิงซ้อน (Multiple Regression and Correlation Analysis)

จากการวิเคราะห์ความถดถอยและสหสัมพันธ์อย่างง่าย เป็นเพียงการศึกษาถึงความสัมพันธ์ของตัวแปร Y กับตัวแปรอิสระ X เพียงตัวเดียว นั่นคือสนใจการศึกษาปัจจัยที่มีอิทธิพลหรือผลต่อตัวแปรตาม Y เพียงปัจจัยเดียว โดยทั่วไปแล้ว ปัจจัยที่มีอิทธิพลต่อ Y จะมีหลายปัจจัยหรือกล่าวได้ว่ามีตัวแปรอิสระหลายตัวที่มีอิทธิพลต่อ Y

ถ้ามีตัวแปรอิสระ k ตัว (X_1, X_2, ..., X_k) ที่มีความสัมพันธ์กับตัวแปรตาม Y โดยความสัมพันธ์อยู่ในรูปเชิงเส้น จะได้สมการความถดถอยเชิงซ้อน ซึ่งแสดงความสัมพันธ์ระหว่าง Y และ X_1, X_2, ..., X_k ดังนี้

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + e \qquad (2.4)$$

กำหนดความหมายของตัวแปรดังนี้

ตัวแปร Y แทนด้วยตัวแปรตาม

ตัวแปร X_1, X_2, X_3, ..., X_k แทนตัวแปรอิสระ

ตัวแปร e แทนค่าความคลาดเคลื่อนอย่างสุ่ม

ตัวแปร β_0 แทนส่วนตัดแกน Y เมื่อ $X_1 = X_2 = \ldots X_k = 0$

ตัวแปร $\beta_0, \ldots, \beta_k$ แทนสัมประสิทธิ์ความถดถอยเชิงส่วน (Partial Regression Coefficient) ตัวแปร Y และ X_1, X_2, X_3, ..., X_k มีความสัมพันธ์ในรูปแบบเชิงเส้น

$\beta_1, \beta_2, \ldots, \beta_k$ เป็นสัมประสิทธิ์ความถดถอยเชิงส่วน (Partial Regression Coefficient) โดยที่ค่า β_i เป็นค่าที่แสดงถึงการเปลี่ยนแปลงของตัวแปรตาม Y เมื่อตัวแปรอิสระ X_i เปลี่ยนไป 1 หน่วย โดยที่ตัวแปรอิสระ X ตัวอื่นๆ มีค่าคงที่ เช่น ถ้า X_1 เปลี่ยนไป 1 หน่วย ค่า Y จะเปลี่ยนไป β_1 หน่วย โดยที่ X_2,X_3, ..., X_k มีค่าคงที่

และค่าประมาณของ Y คือ

$$\hat{Y} = a + b_1X_1 + b_2X_2 + \cdots + b_kX_k \tag{2.5}$$

2.1.6 ตัวแปรหุ่น (Dummy Variable)

ในการเขียนสมการพยากรณ์เพื่อพยากรณ์ตัวแปรตาม นอกจากจะให้ความสำคัญกับตัวแปรเชิงปริมาณแล้วตัวแปรเชิงคุณภาพก็มีความสำคัญไม่น้อยไปกว่ากัน แต่ในการวิเคราะห์การถดถอยโดยทั่วไปจะเน้นศึกษาเฉพาะตัวแปรเชิงปริมาณ เพราะสามารถวัดค่าออกมาได้ ทำให้ง่ายต่อการนำไปวิเคราะห์ และมองข้ามตัวแปรเชิงคุณภาพซึ่งก็มีอิทธิพลต่อตัวแปรตามเช่นกัน ทำให้ขาดความรู้ความเข้าใจเกี่ยวกับการนำตัวแปรเชิงคุณภาพไปใช้ในการวิเคราะห์การถดถอย แต่ต้องทำความเข้าใจในเบื้องต้นก่อนว่าตัวแปรเชิงคุณภาพที่ใช้ในการวิเคราะห์การถดถอย จะนำไปใช้เฉพาะตัวแปรพยากรณ์เท่านั้น ส่วนตัวแปรตามต้องเป็นตัวแปรที่วัดได้ในเชิงปริมาณ ซึ่งถ้ามีความจำเป็นในการนำตัวแปรเหล่านี้มาใช้ในสมการพยากรณ์ สามารถกระทำได้โดยทำตัวแปรดังกล่าวให้เป็นตัวแปรจัดประเภท (Categorical Variable) หรือตัวแปรหุ่น (Dummy Variable) เพื่อนำไปวิเคราะห์การถดถอย [20], [21]

สำหรับตัวแปรเชิงคุณภาพที่จะนำมาใช้กับวิธีนี้มีสองลักษณะ คือ

1) ตัวแปรทวิภาค (Dichotomous Variable) เป็นตัวแปรที่เป็นไปได้เพียง 2 ลักษณะ (g=2) เช่น เพศ (ชาย, หญิง) การตอบข้อสอบ (ถูก, ผิด) เป็นต้น

2) ตัวแปรพหุวิภาค (Polytomous Variable) เป็นตัวแปรที่มีจำนวนรายการมากกว่า 2 ลักษณะ เช่น ศาสนา ภูมิลำเนา กลุ่มการทดลอง ประเภทของสินค้า เป็นต้น

สำหรับในงานวิจัยนี้ตัวแปรอคติที่สนใจสามประเภท โดยในแต่ละประเภทถูกแบ่งลักษณะของการมีอคติออกเป็น 5 ระดับดังนั้นจึงนับเป็นตัวแปรแบบพหุวิภาค ที่มีจำนวนรายการเท่ากับ 5 (g=5) ซึ่งสามารถสร้างตัวแปรอิสระที่แตกต่างกันขึ้นมาได้ g-1 ตัวแปร หรือเท่ากับ 5-1 หรือ 4 ตัวแปรหุ่นด้วยแนวคิดของ Dummy Coding กำหนดให้มีการเลือกตัวแปรหรือกลุ่มใดกลุ่มหนึ่งเป็นตัวแปรหรือกลุ่มอ้างอิง สำหรับใช้เป็นกลุ่มหรือตัวแปรเปรียบเทียบ โดยตัวแปรหรือกลุ่มอ้างอิงจะถูกกำหนดค่าให้เป็น 0 สำหรับตัวแปรหรือกลุ่มอื่น ๆ ที่ต้องการนำมาเปรียบเทียบกำหนดค่าให้เป็น 1 ใช้เป็นตัวแทนของตัวแปรอคติในแต่ละประเภทที่ได้สารสนเทศอย่างสมบูรณ์ โดยที่ไม่ต้องใช้ตัวแปร 5 ตัว ซึ่งจะทำให้เกิดความซ้ำซ้อนและไม่ได้สารสนเทศอะไรเพิ่มเติมและทำให้การคำนวณมีปัญหาและผลการวิเคราะห์ไม่สามารถให้ค่าเฉพาะที่คงที่ได้ ดังนั้นตัวแปรที่ 5 ในกรณีนี้จึงไม่จำเป็น ซึ่งรายละเอียดและตัวอย่างของการประยุกต์ใช้กับงานวิจัยนี้ด้วยวิธี Dummy Coding จะแสดงต่อไปในหัวข้อ 3.1.7

2.2 งานวิจัยที่เกี่ยวข้อง

หัวข้อนี้กล่าวถึงภาพรวมของงานวิจัยที่มีมาว่ามี แนวคิด ทฤษฎี คำนิยาม ความหมาย ความสัมพันธ์ และส่วนที่นำมาใช้กับงานวิจัยนี้ โดยในงานวิจัยแรก [22] จะชี้ให้เห็นถึงภาพรวมความสำคัญ ในงานวิจัยในเชิงมนุษยปัจจัย ที่มีผลกับงานวิศวกรรมซอฟต์แวร์ ในงานวิจัยที่สอง [23] เป็นการชี้ให้เห็นถึงผลกระทบต่อกระบวนการวิศวกรรมซอฟต์แวร์กับการมีอคติ โดยจะกล่าวถึงอคติในการยืนยันความเชื่อของตนที่มีผลต่อจำนวนข้อบกพร่องในซอฟต์แวร์ ซึ่งอคติในการยืนยันความเชื่อของตนนี้ก็เป็นเรื่องหนึ่งในเรื่องมนุษยปัจจัย ในงานวิจัยที่สาม [1] จะกล่าวถึงมนุษยปัจจัยพื้นฐานอันก่อให้เกิดความล้มเหลวของการใช้เหตุผลในการออกแบบซอฟต์แวร์

ในงานวิจัยที่สี่ [2] ชี้ให้เหตุถึงการเปลี่ยนแปลงของข้อมูลในแต่ละขั้นตอนที่สนใจศึกษา โดยมุ่งไปในกระบวนการทางวิศวกรรมซอฟต์แวร์ โดยจะกล่าวถึงทฤษฎีข้อมูลสองขั้นตอน ซึ่งเป็นแบบจำลองแสดงการเปลี่ยนแปลงของข้อมูลในแต่ละขั้นตอน ซึ่งสามารถมีอคติที่เกิดขึ้นในระหว่างการเปลี่ยนแปลงในแต่ละขั้นตอนนั้น ๆ ในงานวิจัยที่ห้า [8] จะกล่าวถึงอคติประเภทต่าง ๆ ที่พบบ่อยในการใช้แก้ไขปัญหาต่าง ๆ รวมถึงปัญหาในเชิงวิศวกรรมซอฟต์แวร์ และความสัมพันธ์ระหว่างอคตินั้น ๆ

ในงานวิจัยที่หก [18] ชี้ให้เห็นถึงการสร้างแบบจำลองขึ้นมาเพื่อประเมินความเสี่ยงในด้านมนุษยปัจจัย โดยจะกล่าวถึงการสร้างแบบประเมินความเสี่ยงต่อความสำเร็จของโครงการพัฒนาซอฟต์แวร์ที่เกิดขึ้นจากความแตกต่างทางวัฒนธรรม และในงานวิจัยที่เจ็ด [24] ได้นำเอาวิธีการวิเคราะห์ตัวแปรต้นกับตัวแปรตาม ด้วยวิธีวิเคราะห์ความถดถอยเชิงซ้อน มาประยุกต์ใช้ในการหาปัจจัยความเสี่ยง และผลกระทบที่สัมพันธ์กันกับปัจจัยความเสี่ยง

2.2.1 งานวิจัยเรื่อง "Why Do We Need Personality Diversity in Software Engineering"

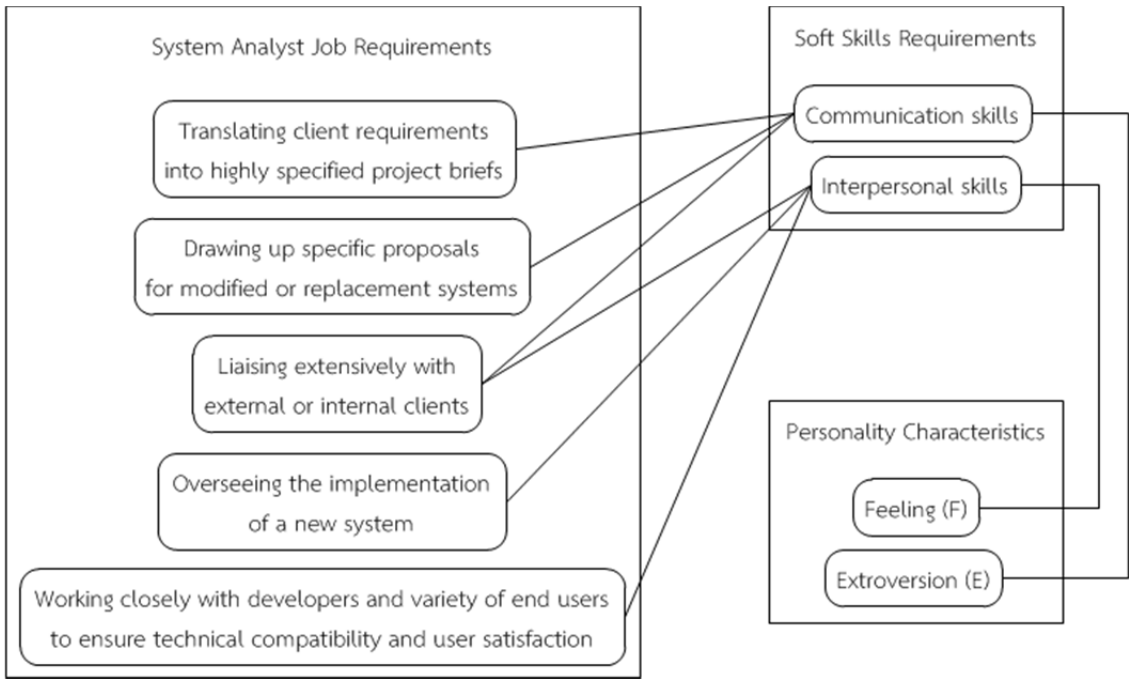
งานวิจัยนี้ [22] ได้กล่าวถึงความสำคัญของความหลากหลายทางบุคลิกภาพของบุคคลในงานวิศวกรรมซอฟต์แวร์ โดยได้กล่าวถึงที่มาของงานวิจัยนี้ว่า ในการแก้ปัญหา มนุษย์มีการมองการแก้ปัญหาไปในหลายมุมมอง ซึ่งรวมไปถึงการแก้ปัญหาในทางด้านวิศวกรรมซอฟต์แวร์ด้วย ซึ่งใช้หลักจิตวิทยามาจำแนกประเภทของบุคคลที่จะทำงานต่าง ๆ ในกระบวนการวิศวกรรมซอฟต์แวร์ให้เหมาะสมในแต่ละบทบาทหน้าที่ โดยวัตถุประสงค์ของงานวิจัยนี้คือ

1) เพื่อกระตุ้นความตระหนักในมนุษยปัจจัย (Human Factor) ในหมู่วิศวกรซอฟต์แวร์
2) เพื่อศึกษาวิธีการด้านปัจจัยทางจิตวิทยาที่สามารถนำไปสู่ความมีประสิทธิภาพในการทำงาน
3) เพื่อกระตุ้นให้วิศวกรซอฟต์แวร์มีความเข้าใจในเชิงลึกและเชิงกว้าง ในการนำมนุษยปัจจัยไปใช้ในแง่ของกิจกรรมที่เกี่ยวกับวิศวกรรมซอฟต์แวร์
4) เพื่อเน้นความสำคัญของความหลากหลายของทักษะในด้านวิศวกรรมซอฟต์แวร์

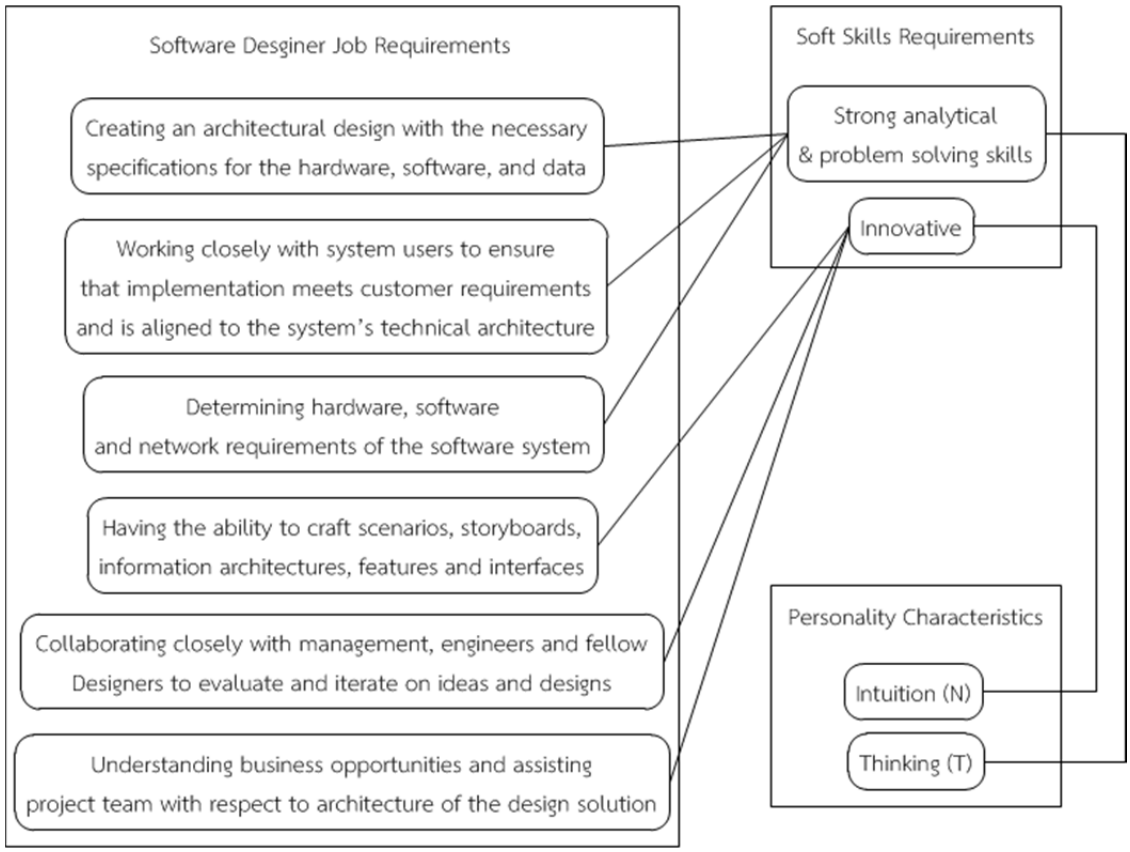
โดยได้ทำการจับคู่ MBTI หรือ Myers-Briggs Type Indicators ซึ่งเป็นเครื่องมือในการวัดและการทำความเข้าใจเกี่ยวกับบุคลิกภาพของปัจเจกบุคคล กับงานที่ต้องทำและทักษะที่ต้องใช้ในแต่ละบทบาทของแต่ละกระบวนการวงจรชีวิตของซอฟต์แวร์ (Software Life Cycle) ว่าในแต่ละงานจะต้องมีบุคคลที่มีบุคลิกภาพอย่างไรจึงจะเหมาะสมกับงานนั้น ๆ ซึ่งสามารถช่วยในการเลือกใช้คนให้เหมาะสมกับงาน เพื่อให้ได้ประสิทธิภาพในการทำงานให้ดีขึ้น และนำเสนอมาเป็นกรอบความรู้ หรือกรอบความคิด ผลวิจัย หรือรายงานที่แสดงถึงสิ่งที่เป็นความจริงและความเชื่อของสมมติฐานจากวงการวิศวกรรมซอฟต์แวร์

จากงานวิจัยที่เกี่ยวข้องนี้ ได้ทราบถึงแนวทางการทำวิจัยด้านจิตวิทยาที่มีความสัมพันธ์เกี่ยวข้องกับงานด้านวิศวกรรมซอฟต์แวร์ว่า มนุษยปัจจัยและการศึกษาความรู้ในศาสตร์อื่น ๆ เช่น ความรู้เชิงจิตวิทยา ควรที่จะเป็นสิ่งที่พึงตระหนักถึงในวงการวิศวกรรมซอฟต์แวร์ด้วยเช่นกัน

และจากงานวิจัยที่เกี่ยวข้องนี้ยังได้พบว่าในงานทางด้านการรวบรวมและกำหนดความต้องการและวิเคราะห์ระบบ (Requirements & System Analysis) และการออกแบบซอฟต์แวร์ (Software Design) มีบุคลิกภาพตามที่ได้จับคู่กับวิธี MBTI แล้วดังภาพที่ 2.3 และ 2.4



ภาพที่ 2.3 แสดงการจับคู่ของนักวิเคราะห์ระบบและทักษะไปยังลักษณะบุคลิกภาพ [22]



ภาพที่ 2.4 แสดงการจับคู่ของนักออกแบบซอฟต์แวร์และทักษะไปยังลักษณะบุคลิกภาพ [22]

นักวิเคราะห์ระบบ (System Analyst) และนักออกแบบซอฟต์แวร์ (Software Designer) นักวิเคราะห์ระบบ มีลักษณะเป็น Extroversion นั่นคือ พูดคุยกันเพื่อจะนำมาคิด ดังนั้นจึงมีลักษณะเป็นคนช่างพูด คนเริ่มต้นบทสนทนา และ Feeling นั่นคือ ในกระบวนการตัดสินใจ นักวิเคราะห์ระบบจะใช้อารมณ์ความรู้สึกในการตัดสินใจ ส่วนนักออกแบบซอฟต์แวร์ มีลักษณะเป็น Intuition นั่นคือเป็นพวกที่การหยั่งรู้โดยสัญชาตญาณ หรือการหยั่งรู้อันเกิดขึ้นภายในใจ และสามรถใช้ข้อมูลที่เป็นนามธรรม หรือแค่เพียงแนวคิดได้ และ Thinking นั่นคือในกระบวนการการตัดสินใจ จะใช้วิธีทางตรรกวิทยา ในการตัดสินใจ

จะเห็นได้ว่าในลักษณะบุคลิกภาพของบุคคลที่ทำงานในงานวิศวกรรมซอฟต์แวร์ มีคุณลักษณะหรือความสามารถที่เปลี่ยนแปลงได้ ตามแต่ลักษณะ และความคิดของบุคคล หรือเป็นนามธรรม (Subjective) เช่น Feeling และ Intuition เป็นต้น จึงได้แนวคิดเหล่านี้มาเป็นส่วนในการศึกษาวิจัยในงานที่เกี่ยวกับการวิเคราะห์และออกแบบซอฟต์แวร์ หรือระบบ

2.2.2 งานวิจัยเรื่อง “Empirical Analyses of the Factors Affecting Confirmation Bias and the Effects of Confirmation Bias on Software Developer/Tester Performance”

งานวิจัยนี้ [23] กล่าวถึงเรื่องของอคติ Confirmation อันเกิดขึ้นในกระบวนการขั้นตอนของการทดสอบซอฟต์แวร์ซึ่งเกี่ยวข้องกับปรากฏการณ์ที่ นักพัฒนาซอฟต์แวร์และนักทดสอบซอฟต์แวร์มักจะเลือกการทดสอบที่เป็นยืนยันผลเป็นบวกมากกว่าการทดสอบเพื่อยืนยันผลเป็นลบ โดยมีทฤษฎีที่เกี่ยวข้องกับผลที่เป็นไปได้ของอคติ Confirmation ในกระบวนการพัฒนาและทดสอบซอฟต์แวร์ เนื่องจากแนวโน้มของการทดสอบซอฟต์แวร์เพื่อยืนยันว่าผลเป็นบวก ทำให้มีข้อบกพร่องของซอฟต์แวร์ที่ตรวจไม่พบ นำไปสู่การเพิ่มมากขึ้นของความหนาแน่นของข้อบกพร่องของซอฟต์แวร์ (Software Defect Density)

ด้วยศึกษาวิเคราะห์ว่ามีปัจจัยที่มีผลต่อการมีอคติ Confirmation เพื่อที่จะหาวิธีในการหลีกเลี่ยงการมีอคติ Confirmation นั้น โดยหาปัจจัยจากประสบการณ์ในการทดสอบและพัฒนาซอฟต์แวร์ ทักษะการให้เหตุผลที่ได้รับมาจากการศึกษา ในมุมของประสิทธิภาพของนักพัฒนาและนักทดสอบซอฟต์แวร์ โดยใช้การทำแบบทดสอบที่อ้างถึงการทดลองทางจิตวิทยาของ Wason นั่นคือ Wason’s Rule Discovery Experiment และ Wason’s Selection Task จากกลุ่มตัวอย่างทั้งที่เป็นพนักงานบริษัทในบริษัทด้านโทรคมนาคม และกลุ่มบัณฑิตศึกษาทางด้านวิศวกรรมคอมพิวเตอร์

ผลจากงานวิจัยที่เกี่ยวข้องนี้วิเคราะห์พบว่ามีความเกี่ยวพันกันโดยตรง ระหว่างความสัมพันธ์ของความหนาแน่นของข้อผิดพลาดของซอฟต์แวร์กับระดับของการมีอคติ Confirmation ของผู้พัฒนาและผู้ทดสอบ และยังได้กล่าวถึงงานวิจัยในอนาคตว่า จะมีการใช้อคติ Confirmation เป็นตัววัดในการคาดการณ์ล่วงหน้าในการหาข้อบกพร่องของซอฟต์แวร์

จากงานวิจัยที่เกี่ยวข้องนี้แสดงให้เห็นถึงความสัมพันธ์ของการมีอคติ Confirmation อันเป็นอคติตัวหนึ่งในอคติทางการรู้คิด ซึ่งเป็นเรื่องหนึ่งที่เกี่ยวกับมนุษยปัจจัย กับเรื่องการของพัฒนาและทดสอบซอฟต์แวร์ อันเป็นผลให้มีการเพิ่มขึ้นอย่างมีนัยสำคัญของข้อบกพร่องของซอฟต์แวร์ ซึ่งผู้วิจัยมองว่าเป็นความเสี่ยงหนึ่งต่อความน่าเชื่อถือ และความสำเร็จของโครงการพัฒนาซอฟต์แวร์ ซึ่งผู้วิจัย

ได้นำแนวความคิดจากตอนท้ายของงานวิจัยในการคาดการณ์ถึงข้อบกพร่องที่อาจจะเกิดในซอฟต์แวร์ เป็นที่มาในการหาปัจจัยความเสี่ยงอันอาจจะเกิดจากการมีอคติทางการรู้คิดเหล่านี้

2.2.3 งานวิจัยเรื่อง “Software Designers, Are You Biased?”

งานวิจัยนี้ [1] มีการตั้งคำถามเกี่ยวกับเรื่องของการออกแบบซอฟต์แวร์ไว้ว่า กระบวนการทางวิศวกรรมซอฟต์แวร์ที่มีอยู่นั้น เพียงพอต่อการออกแบบซอฟต์แวร์ให้เหมาะสมหรือไม่ โดยในงานวิจัยนี้ได้กล่าวอ้างว่า โดยทั่วไปพบว่ามนุษย์มักจะมีการตัดสินใจที่มีความเอียง หรือมีอคติในการตัดสินใจ นั่นรวมถึงนักออกแบบซอฟต์แวร์เอง หรือผู้เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ ก็อาจจะตกอยู่ภายใต้อคติเหล่านี้เช่นกัน

โดยจุดมุ่งหมายของงานวิจัยนี้ต้องการที่จะให้เห็นถึงภาพรวมของอคติทางการรู้คิดและความล้มเหลวในการใช้เหตุผลอันอาจจะนำไปสู่การตัดสินใจในการออกแบบที่ไม่น่าเชื่อถือ ซึ่งคาดว่าจะช่วยในการปรับปรุงคุณภาพโดยรวมของการออกแบบซอฟต์แวร์ที่ดีได้ โดยได้ให้สามสาเหตุพื้นฐานอันก่อให้เกิดความล้มเหลวของการใช้เหตุผลในการออกแบบ ดังนี้

1) Cognitive bias – ความผิดเพี้ยนไปของการตัดสินใจในสถานการณ์เฉพาะอย่าง อันเนื่องมาจากผลทางจิตวิทยาและการพิจารณาไตร่ตรองที่ไม่เพียงพอ ของสิ่งที่น่าจะเป็นไปได้ หรือความเป็นไปได้
2) Illogical reasoning – เหตุผลเชิงตรรกวิทยานั้นต้องการข้ออ้างพื้นฐานของการตัดสินใจ โดยสำหรับกรณีเกี่ยวกับการออกแบบและความต้องการนั้น ข้ออ้างพื้นฐานควรจะเป็นข้อเท็จจริงและเป็นจริง รวมถึงข้อโต้แย้งและการอนุมานถึงข้อสรุปในการออกแบบ ควรจะน่าเชื่อถือได้
3) Low quality premises – ในโครงการซอฟต์แวร์ขนาดใหญ่ซึ่งใช้นักออกแบบหลายคนในการตัดสินใจ มักจะมีความล้มเหลวในการอธิบายถึงข้ออ้างอย่างถูกต้อง เช่น ข้ออ้างที่มีความไม่ถูกต้อง หรือข้ออ้างที่มีไม่เพียงพอ สามารถเป็นสาเหตุของการตัดสินใจที่ไม่ถูกต้อง

โดยได้อธิบายถึงสาเหตุทั้งสามด้วยว่ามีรายละเอียดอย่างไรและมีการทำงานอย่างไร นอกจากนี้งานวิจัยนี้ยังได้ให้วิธีปฏิบัติในการปรับปรุงเรื่องการใช้เหตุผลในการออกแบบไว้อีกด้วยโดยรวบรวมมาจากหลาย ๆ งานวิจัย เพื่อหลีกเลี่ยงการที่นักออกแบบจะมีการใช้ความมีอคติ หนึ่งในนั้นมีประเด็นเรื่อง

การวิเคราะห์ความเสี่ยง (Risk Analysis) โดยเน้นไปในเรื่องของการจัดลำดับความสำคัญและผลกระทบในการออกแบบซอฟต์แวร์ เพื่อเตือนนักออกแบบซอฟต์แวร์ให้มีการตระหนักถึงเรื่องนี้ด้วย

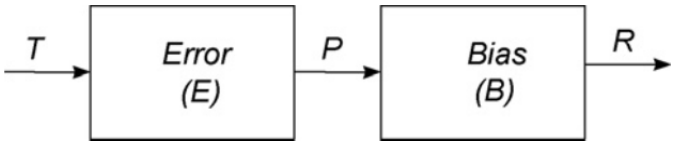
และยังได้กล่าวถึงงานวิจัยในอนาคต โดยได้แนะนำว่าน่าจะมีเรื่องใดที่จะสามารถดำเนินการศึกษาต่อไปได้ ตัวอย่างเช่น จะมีความอคติใดเกิดขึ้น และทำไมจึงเกิดขึ้น แล้วเราจะหลีกเลี่ยงมันได้อย่างไร โดยจะมีกระบวนการใดในวิศวกรรมซอฟต์แวร์ที่จะสามารถมีอคติเกิดขึ้นได้อีกบ้าง หรืออาจจะเป็นการศึกษาวิธีการทางเหตุผลในการออกแบบเพื่อปรับปรุงคุณภาพของการออกแบบ หรือนำไปประยุกต์ใช้ในงานอื่น ๆ เช่น สถาปัตยกรรมซอฟต์แวร์ เป็นต้น

จากงานวิจัยที่เกี่ยวข้องนี้แสดงให้ทราบว่ามีการทำการวิจัยที่เกี่ยวกับอคติ การใช้เหตุผล การตัดสินใจ การหาข้อสรุปจากข้ออ้างอิง ในกระบวนการอื่น ๆ ของวิศวกรรมซอฟต์แวร์ โดยในที่นี้คือกระบวนการออกแบบซอฟต์แวร์ และได้ให้รายละเอียดและคำจำกัดความของประเด็นดังกล่าวไว้ พร้อมทั้งแนวทางการวิจัยในอนาคตที่สามารถนำมาใช้ในการค้นหางานวิจัยเพื่อมาต่อยอดจากคำถามที่งานวิจัยนี้ได้ตั้งไว้ เช่น จะมีความอคติใดเกิดขึ้นบ้าง ที่กระบวนการใดบ้าง และส่งผลอย่างไร ซึ่งผู้วิจัยได้นำมาเป็นแนวทางในการออกแบบงานวิจัยนี้ว่าควรจะให้ความสนใจในอคติประเภทใดบ้าง และอคติเหล่านั้นน่าจะเกิดที่กระบวนการใด และจะส่งผลอย่างไร โดยผู้วิจัยจะประมาณความเสี่ยงที่อาจจะเกิดขึ้นจากผลกระทบและความถี่ในการเกิดขึ้นได้

2.2.4 งานวิจัยเรื่อง “The effects of optimistic and pessimistic biasing on software project status reporting”

งานวิจัยนี้ [2] ได้กล่าวถึงเอกสารรายงานสถานะของโครงการซอฟต์แวร์ (Software Project Status Report) ที่มีการใช้อคติอยู่ในการจัดทำรายงาน โดยพบว่ารายงานดังกล่าวมีอคติอยู่ถึง 60% โดยอคติที่งานวิจัยนี้สนใจคือการที่ทำรายงานสถานะออกไปในแง่บวก (Optimistic) และแง่ลบ (Pessimistic) โดยงานวิจัยนี้ได้ให้เหตุผลของการทำรายงานสถานะออกมาในสองแง่ ว่ามีเหตุผลจากอะไร จากการทำแบบสำรวจเพื่อสอบถามความคิดเห็นของกลุ่มผู้จัดการโครงการ (Project Manager) ตัวอย่างเช่น Optimistic เกิดจากการที่ผู้จัดการโครงการเกรงที่รายงานข่าวที่ไม่ดีให้ผู้บริหารระดับสูงและต้องการที่จะให้ตัวเองดูดี เป็นต้น สำหรับ Pessimistic เกิดจากผู้จัดการโครงการต้องการเวลาเพิ่มในการทำงาน เป็นต้น

โดยสนใจศึกษาความถี่ที่จะเกิดอคติและความเข้มข้นของการนำอคติไปใช้ โดยได้ใช้แบบจำลองทฤษฎีข้อมูล (Information-Theoretic Model) แบบ Two-stage probabilistic model เพื่อศึกษาผลกระทบของรายงานที่มีอคติ และรายงานที่มีข้อผิดพลาด ซึ่งแสดงได้ดังภาพที่ 2.5



ภาพที่ 2.5 Two-stage status model [2], [4]

จากภาพที่ 2.5 อักษรภาษาอังกฤษแทนความหมายได้ดังนี้

T คือ True status หรือสถานะที่เป็นจริง

P คือ Perceived status หรือสถานะที่ได้รับ

R คือ Report status หรือตัวเอกสารายงานสถานะ

E คือ Error หรือความผิดพลาดที่เกิดขึ้น

B คือ Bias หรืออคติที่เกิดขึ้น

โดยที่สถานะที่เป็นจริง กับสถานะที่ได้รับจะแตกต่างกัน ถ้าผู้จัดการโครงการมีข้อผิดพลาด ในการประเมินสถานะปัจจุบันกับตัวรายงานสถานะจะแตกต่างกันถ้าผู้จัดการโครงการมีอคติในการรับรู้ของผู้จัดการโครงการก่อนที่จะรายงานสถานะดังกล่าว สำหรับลักษณะของโครงการ กำหนดให้อยู่ในโครงการที่มีความเสี่ยงสูงและโครงการที่มีความเสี่ยงต่ำ โดยจะได้ความน่าจะเป็นของการเกิดหรือไม่เกิดอคติในโครงการแต่ละลักษณะ รวมถึงความน่าจะเป็นที่เกิดอคติประเภทแง่ดี หรือแง่ร้ายในโครงการแต่ละลักษณะด้วย และจุดมุ่งหมายของงานวิจัยนี้เพื่อให้เข้าใจถึงการจัดทำแบบรายงานสถานะโครงการซอฟต์แวร์ให้ดียิ่งขึ้น

จากงานวิจัยที่เกี่ยวข้องนี้ได้นำแนวคิดของ Two-stage status model มาใช้อ้างอิงเพื่อเป็นข้อมูลพื้นฐานในการสร้างแบบจำลอง โดยผู้วิจัยคาดว่าน่าจะมีอคติเกิดขึ้น โดยไม่สนใจขั้นตอนที่อาจจะเกิดความผิดพลาด (Error) ในกระบวนการการรวบรวมความต้องการ (Requirements Gathering) ซึ่งจะได้ข้อมูลมาชุดหนึ่ง เป็นข้อกำหนดความต้องการซอฟต์แวร์ (Software Requirements

Specification) และกระบวนการสร้างข้อกำหนดการออกแบบ (Design Specification) ซึ่งก็จัดว่าเป็นลำดับการเปลี่ยนแปลงไปของข้อมูลตามแบบจำลองทฤษฎีข้อมูล โดยจากข้อมูลความต้องการไปเป็นข้อกำหนดความต้องการ และจากข้อกำหนดความต้องการไปเป็นข้อกำหนดการออกแบบ นอกจากนี้ยังได้เอาวิธีการจัดทำแบบสอบถามเพื่อรวบรวมข้อมูลของงานวิจัยนี้มาใช้ โดยงานวิจัยนี้ได้กล่าวถึงการจัดทำแบบสอบถามโดยใช้ Four Research Question Strategy คือเป็นการออกแบบแบบสอบถามเพื่อช่วยให้ผู้ตอบแบบสอบถามสามารถตอบคำถามให้ได้ตรงจุดที่ต้องการ

2.2.5 งานวิจัยเรื่อง “Using Traceability to Mitigate Cognitive Biases in Software Development”

งานวิจัยนี้ [8] ได้กล่าวถึงการตามตรวจสอบย้อนกลับ (Traceability) เพื่อลดอคติทางการรู้คิดในการพัฒนาซอฟต์แวร์ โดยก็มีการตั้งคำถามคล้าย ๆ กับงานวิจัยก่อนหน้าว่า มีนักพัฒนาซอฟต์แวร์ที่ได้รับผลกระทบจากอคติทางการรู้คิดในขณะที่ทำการออกแบบซอฟต์แวร์บ้างหรือไม่ ถ้ามีแล้ว จะสามารถลดผลกระทบของอคติแต่ละตัวต่อประสิทธิภาพผู้พัฒนาได้อย่างไร และยังกล่าวว่า อย่างไรก็ตามคำถามเหล่านี้ ก็ยังไม่ได้รับการแสวงหาคำตอบในบริบทของการพัฒนาซอฟต์แวร์เท่าใดนัก เมื่อมีการเปลี่ยนแปลงส่วนต่าง ๆ ของซอฟต์แวร์ นักพัฒนามักจะใช้วิทยาการศึกษาสำนึก (Heuristics) ซึ่งหมายถึงวิธีการแก้ไขปัญหาที่เหมือนจะไม่มีแนวทาง หรือกฎเกณฑ์ที่ตายตัว คงมีแตรการทดลอง หรือการคาดเดาในหลาย ๆ ทางเลือกในการตัดสินใจที่คิดว่าจะได้ผลดีที่สุด ซึ่งมักจะใช้ผลจากประสบการณ์ที่ผ่านมา ที่ได้รับอิทธิพลจากอคติทางการรู้คิด โดยงานวิจัยนี้มีเป้าหมายเพื่อจะทดสอบความมีอิทธิพลและการพยายามหาการลดความมีอคติด้วยการใช้เครื่องช่วย เช่น การตรวจสอบย้อนกลับ เป็นต้น

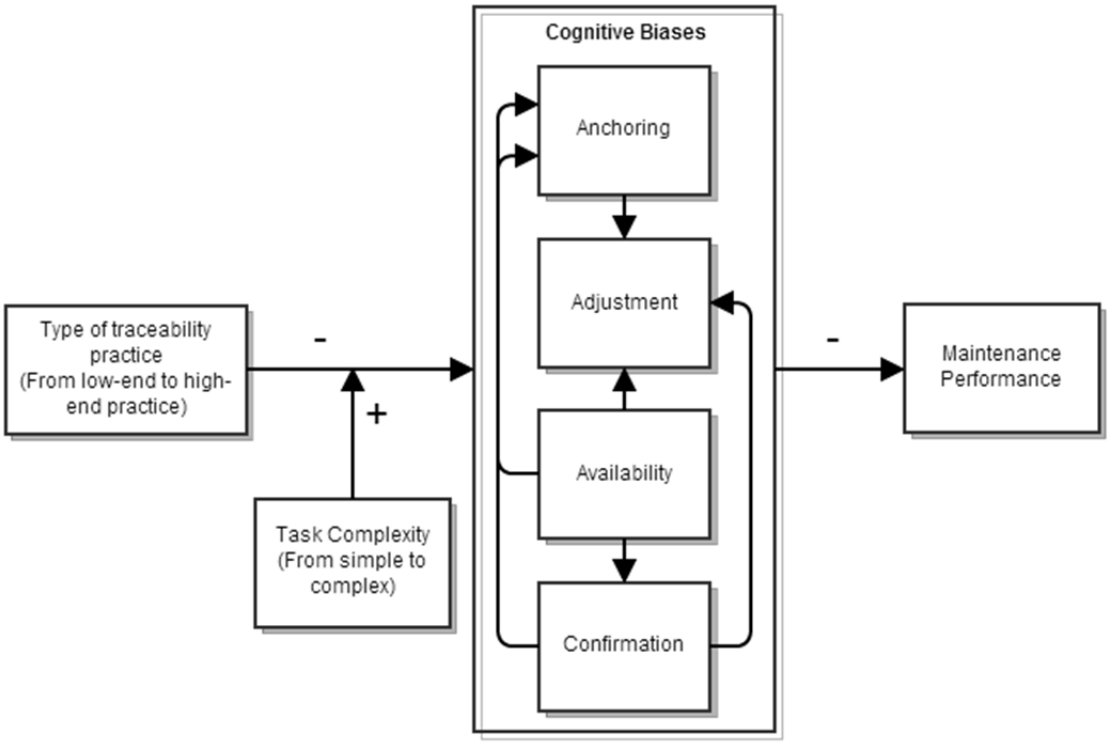
ในงานวิจัยนี้ได้อธิบายถึงอคติทางการรู้คิดที่เป็นที่พบมากที่สุดในการแก้ปัญหาต่าง ๆ คำจำกัดความ และตัวอย่างของอคติทางการรู้คิดในการแก้ปัญหาไว้ ดังนี้

1) Anchoring and adjustment bias – อคตินี้มาจากการประมาณการเบื้องต้นของมนุษย์เกี่ยวกับปัญหาและการปรับเปลี่ยนการประมาณการเบื้องต้นเหล่านั้น ก่อนที่จะได้มาเป็นขั้นสุดท้ายของการแก้ปัญหาที่เหมาะสมกว่า

2) Availability bias – อคตินี้แสดงให้เห็นว่ามนุษย์ได้รับอิทธิพลจากข้อมูลที่ง่ายต่อการระลึกถึงและมีอคติจากข้อมูลที่อยู่ในการเผยแพร่ของสื่อ หรือข้อมูลที่เพิ่งเกิดขึ้นมาเมื่อไม่นานนี้
3) Confirmation bias – อคตินี้แสดงให้เห็นว่ามนุษย์มักจะมุ่งเน้นที่ข้อมูลที่สอดคล้องกับแนวคิดที่ไตร่ตรองไว้ก่อนแล้วของพวกเขา ขณะที่พวกเขาไม่สนใจข้อมูลที่ไม่สอดคล้องกับแนวความคิดที่ผ่านมาของพวกเขา

นอกจากนี้ งานวิจัยนี้ยังได้บอกถึงการที่อคติทางการรู้คิดมีผลต่อการพัฒนาซอฟต์แวร์ได้อย่างไร ซึ่งนำเสนอโดยการอธิบายถึงการทำงานของอคติทางการรู้คิดว่ามีการทำงานอย่างไร โดยระบุว่าอคติ Anchoring and adjustment เกิดขึ้นในหลายขั้นตอนตลอดกระบวนการที่มีการเปลี่ยนแปลงการออกแบบระบบ หรือซอฟต์แวร์ ผู้พัฒนาจะใช้อคติ Anchoring and adjustment ตามวัตถุประสงค์ในการพัฒนาในตอนต้นของกระบวนการเพื่อที่จะทำความเข้าใจกับความต้องการของระบบ การออกแบบระบบ เป็นต้น หลังจากนั้น จะมีการใช้อคติ Confirmation และอคติ Availability ซึ่งพบว่ากระทบกับกระบวนการของอคติ Anchoring and adjustment ตลอดจนวิธีกาโดยธรรมชาติในการหาความรู้ของนักพัฒนา

โดยได้นำเสนอกรอบงานอันแสดงถึงความสัมพันธ์ของเหตุผลระหว่าง การตรวจสอบย้อนกลับ ความซับซ้อนของงาน อคติทางการรู้คิด และประสิทธิภาพในการบำรุงรักษาไว้ดังภาพที่ 2.6



ภาพที่ 2.6 Causal Framework แสดงความสัมพันธ์ระหว่าง Traceability, Task complexity, Cognitive biases และ Maintenance performance [8]

จากงานวิจัยที่เกี่ยวข้องนี้ได้นำเอาคำจำกัดความมาใช้สำหรับอคติแต่ละตัวที่สนใจ โดยอคติที่สนใจนั้นเป็นอคติตัวเดียวกันกับที่งานวิจัยนี้ได้นำเสนอไว้ ทั้งนี้ยังนำเอาความสัมพันธ์จาก Causal Framework เฉพาะในส่วนของ อคติทางการรู้คิด (Cognitive Biases) มาเป็นส่วนในการพิจารณาเพื่อเขียนสูตรแสดงความสัมพันธ์ของอคติทางการรู้คิดว่ามีความสัมพันธ์กันอย่างไร มีการทำงานอย่างไร และเพื่อจะดูว่าจะส่งผลกระทบต่อการเกิดความเสี่ยงได้อย่างไร โดยจะนำเอามาเป็นปัจจัยที่น่าจะทำให้เกิดความเสี่ยง

2.2.6 งานวิจัยเรื่อง “Cross-Cultural Risk Assessment Model”

งานวิจัยนี้ [18] เป็นงานวิจัยหนึ่งที่กล่าวถึงเรื่องของมนุษยปัจจัยเช่นกัน เพียงแต่เป็นในด้านของความแตกต่างทางวัฒนธรรม ที่มุ่งศึกษาในแง่ของการประเมินความเสี่ยง (Risk Assessment) ที่หาค่าความเสี่ยงอันเนื่องมาจากความแตกต่างทางวัฒนธรรมของบุคคลากรในทีมพัฒนาซอฟต์แวร์ โดยแบ่งแยกปัจจัยความเสี่ยงตามความแตกต่างด้านภาษา ลักษณะการติดต่อสื่อสาร ทัศนคติเกี่ยวกับเวลา และวิธีการพัฒนา

แบบประเมินความเสี่ยงที่งานวิจัยนี้สร้างขึ้น ประกอบด้วยตัวแบบเชิงคณิตศาสตร์และแบบสอบถามที่เป็นเครื่องมือในการเก็บข้อมูลเพื่อใช้ในการคำนวณค่าความเสี่ยง โดยตัวแบบเชิงคณิตศาสตร์สร้างจากการนิยามของความเสี่ยง คือ ความเสี่ยงมีค่าเท่ากับ ความน่าจะเป็นที่จะเกิดเหตุการณ์ที่จะมีความเสียหายคูณกับมูลค่าของความเสียหายเมื่อเกิดเหตุการณ์นั้น ๆ สำหรับความน่าจะเป็นหาได้จากค่าเฉลี่ยของจำนวนความขัดแย้งที่เกิดขึ้นภายในทีมพัฒนาซอฟต์แวร์ และแปลงค่าความน่าจะเป็นโดยสูตรการแจกแจงปัวซง และมูลค่าความเสียหายได้มาจากการประเมินค่าตามระดับความเสียหาย โดยแบบประเมินค่าความเสี่ยงนี้สามารถนำไปใช้กับบริษัท หรือทีมพัฒนาซอฟต์แวร์ที่มีบุคคลากรต่างวัฒนธรรมมาทำงานร่วมกัน ผลการประเมินสามารถใช้ในการบริหารความเสี่ยงในขณะดำเนินการโครงการ เพื่อผลสำเร็จของการพัฒนาซอฟต์แวร์ในที่สุด

จากงานวิจัยที่เกี่ยวข้องนี้ได้นำเอาวิธีการสร้างแบบจำลองในการประเมินความเสี่ยงจากทฤษฎีความเสี่ยงของ Barry W. Boehm มาเป็นต้นแบบ โดยนำมาปรับเข้าให้ใช้ได้กับมนุษยปัจจัยในด้านของอคติทางการรู้คิด ที่ได้ระบุปัจจัยขึ้นมาจากอคติทั้งสามประเภท นั่นคืออคติ Anchoring and adjustment, Availability และ Confirmation

นอกจากนี้ยังมีการทำแบบสอบถามเพื่อรวบรวมข้อมูล ซึ่งก็ได้นำเอาแนวคิดในการทำแบบสอบถามนี้ร่วมด้วยกับแนวคิด Four Research Question Strategy อีกด้วย เพื่อให้ได้มาซึ่งความน่าจะเป็นอันอาจจะเกิดขึ้นของปัจจัยต่าง ๆ ผลกระทบ โดยรวบรวมมาจากแบบสอบถาม อันจะนำมาเป็นข้อมูลนำเข้าของแบบประเมินความเสี่ยงที่สร้างขึ้นต่อไป

2.2.7 งานวิจัยเรื่อง “Framework for Developing a Software Cost Estimation Model for Software Modification Based on a Relational Matrix of Project Profile and Software Cost Using and Analogy Estimation Method”

งานวิจัยนี้ [24] ทำการศึกษาและการวิเคราะห์โพรไฟล์ (Profile) โครงการจากปัจจัยหลักที่มีผลต่อต้นทุน เพื่อนำเสนอแบบจำลองการประมาณต้นทุนซอฟต์แวร์ส่วนปรับปรุงเพิ่มเติมจากเมตริกซ์ความสัมพันธ์ของรายละเอียดโครงการและปัจจัยต้นทุน โดยใช้วิธีการเปรียบเทียบ (Analogy) ในการประมาณจำนวนรายการ (Modification List) และแรงงาน (Effort) ที่ใช้ในการพัฒนาซอฟต์แวร์ส่วนปรับปรุงเพิ่มเติมจากซอฟต์แวร์ที่มีอยู่ให้สอดคล้องกับความต้องการของลูกค้า

โดยงานวิจัยนี้ศึกษาการทำให้วิธีการประมาณต้นทุนซอฟต์แวร์ในแบบที่ไม่ใช่อัลกอริทึม ให้เป็นวิธีการอัลกอริทึม ซึ่งพัฒนาวิธีการอัลกอริทึมมาจากวิธีเปรียบเทียบ และสนใจในส่วนซอฟต์แวร์ส่วนปรับปรุงเพิ่มเติม โดยศึกษาปัจจัยในส่วนของปัจจัยทางด้านลูกค้า ปัจจัยทางด้านซอฟต์แวร์ และปัจจัยของทีมงาน และนำมาหาความสัมพันธ์ด้วยวิธีทางสถิติ โดยใช้วิธีการวิเคราะห์ความถดถอยเชิงสหสัมพันธ์ เพื่อศึกษาว่า ปัจจัยใดมีผลต่อรายการการปรับปรุงเปลี่ยนแปลงและแรงงานบ้าง และเพื่อให้ได้ค่าประมาณไปใช้ในการประมาณต้นทุนซอฟต์แวร์ต่อไป

จากงานวิจัยที่เกี่ยวข้องนี้ ได้นำเอาแนวคิดในการนำวิธีการทางสถิติมาใช้ในการหาความสัมพันธ์ และหาค่าประมาณ เพื่อนำไปศึกษาว่าปัจจัยเสี่ยงทางด้านอคติทางการรู้คิดแต่ละประเภท มีความสัมพันธ์กับอัตราความเสียหายของซอฟต์แวร์อย่างไร และเพื่อให้ได้สมการทางคณิตศาสตร์ จากวิธีการวิเคราะห์ความถดถอยเชิงสหสัมพันธ์ ในการนำมาหาค่าความเสียหายประมาณของซอฟต์แวร์จากปัจจัยความเสี่ยงเหล่านั้น เพื่อนำไปเป็นค่าอัตราความเสียหายในการประเมินความเสี่ยงในสมการหาค่าความเสี่ยงต่อไป

บทที่ 3

แบบจำลองการประเมินความเสี่ยงจากการมีอคติทางการรู้คิด

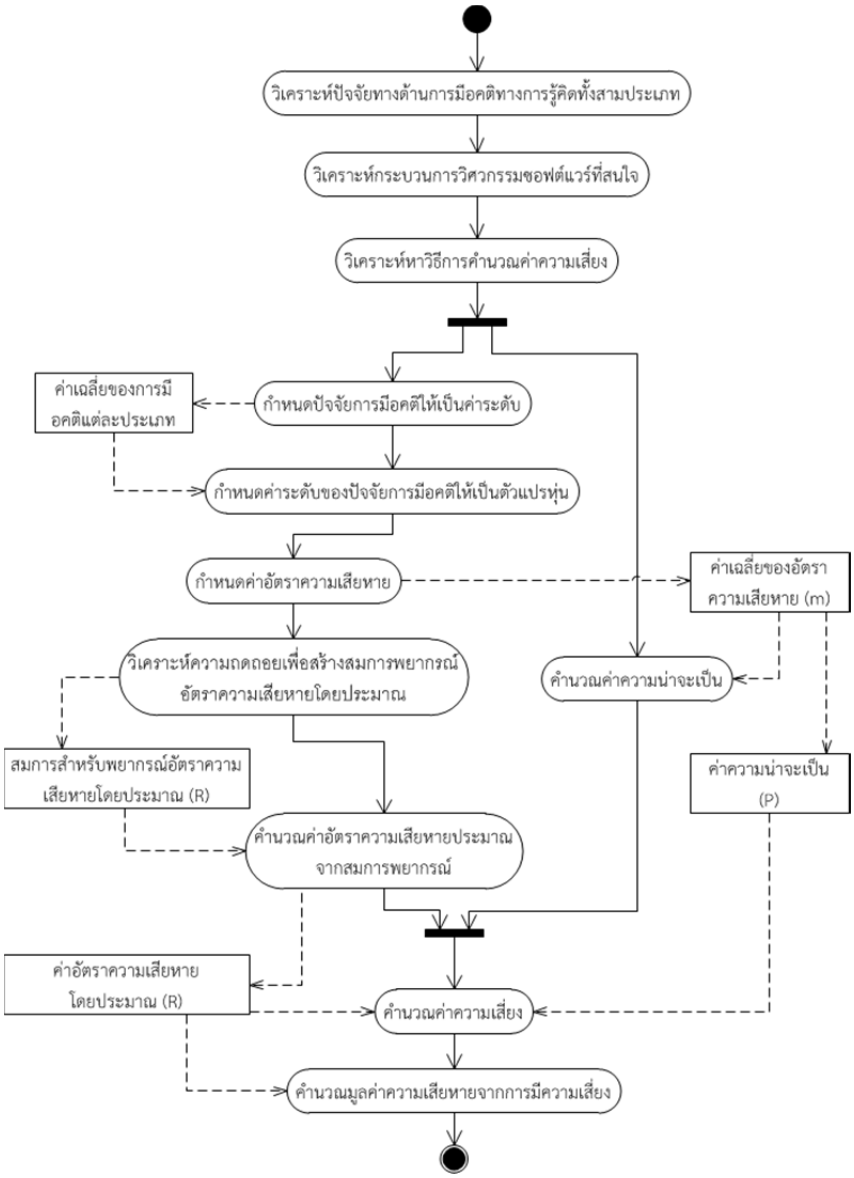
ในบทนี้แบ่งเนื้อหาออกเป็นสองส่วน ส่วนแรกอธิบายแนวคิดในการสร้างแบบจำลองการประเมินความเสี่ยงของอัตราความเสียหายของซอฟต์แวร์จากการมีอคติทางการรู้คิด สำหรับใช้ในการคำนวณหาค่าความเสี่ยงและความเสียหายอันอาจจะเกิดขึ้น โดยจะนำไปสร้างแบบจำลองสนับสนุนแบบจำลองนี้ ซึ่งข้อมูลที่จะใช้ในส่วนแรกนั้นได้มาจากข้อมูลจากแบบสอบถาม ซึ่งส่วนที่สองอธิบายการสร้างแบบสอบถามนั้น เพื่อนำผลของแบบสอบถามที่ได้ไปใช้กับแบบจำลองและเครื่องมือที่ได้สร้างขึ้น

3.1 แบบจำลองการประเมินความเสี่ยง

แบบจำลองนี้อธิบายด้วยเชิงคณิตศาสตร์ โดยแบ่งออกเป็นสองส่วนคือ ส่วนการหาอัตราความเสียหายของซอฟต์แวร์ด้วยการวิเคราะห์ความถดถอย และส่วนการหาความน่าจะเป็นด้วยความน่าจะเป็นแบบปัวซอง เพื่อที่สุดท้ายแล้วจะนำค่าที่ได้จากทั้งสองส่วนมาหาค่าความเสี่ยงจากการคูณกันของทั้งสองค่าในสมการความเสี่ยง

ซึ่งจะนำเอาแบบจำลองนี้ไปสร้างเป็นเครื่องมือเพื่อใช้ในการประเมินความเสี่ยงในบทที่ห้า โดยนำสมการประมาณค่าความเสียหายที่ได้จากการวิเคราะห์ความถดถอย ไปใช้ในการหาค่าประมาณความเสียหายจากการรับค่านำเข้าเป็นระดับของการมีอคติในแต่ละประเภทแล้วจะได้เป็นค่าประมาณความเสียหายจากการมีอคติแต่ละประเภทออกมา จากนั้นนำค่าที่ได้นั้นไปเฉลี่ยแล้วจำแนกออกเป็นแต่ละระดับความรุนแรง เพื่อที่จะได้เป็นค่าประมาณความเสียหายในแต่ละระดับความรุนแรงไปคูณกับค่าความน่าจะเป็นในแต่ละระดับความรุนแรง ซึ่งถูกกำหนดให้เป็นค่าคงที่จากการหาความน่าจะเป็นแบบปัวซองเอาไว้ก่อนแล้ว ในท้ายที่สุดก็จะได้ค่าความเสี่ยงในแต่ละระดับความรุนแรง และเมื่อนำค่าความเสี่ยงดังกล่าวมารวมกัน ก็จะได้เป็นค่าความเสี่ยงรวม เพื่อที่จะนำไปใช้ในการบริหารจัดการความเสียงต่อไป

การสร้างแบบจำลองประเมินความเสี่ยงสามารถอธิบายด้วยแผนภาพกิจกรรมแสดงให้เห็นภาพรวมของการสร้างแบบจำลอง ดังภาพที่ 3.1



ภาพที่ 3.1 แผนภาพกิจกรรมอธิบายภาพรวมของการสร้างแบบจำลองการประเมินความเสี่ยง

ในส่วนของการสร้างแบบจำลองการประเมินความเสี่ยง เริ่มต้นที่การวิเคราะห์ดูปัจจัยทางด้านอคติทางการรู้คิดที่จะใช้ในแบบจำลองการประเมินความเสี่ยงว่ามีลักษณะเป็นอย่างไร มีความสัมพันธ์กันอย่างไร หลังจากนั้นทำการวิเคราะห์หาวิธีการคำนวณค่าความเสี่ยงที่เหมาะสม จากนั้นทำการวิเคราะห์กระบวนการวิศวกรรมซอฟต์แวร์ที่สนใจและคาดว่าจะมีอคติเกิดขึ้น แล้วทำการกำหนดปัจจัยทางอคติในกระบวนการที่สนใจให้เป็นค่าระดับเพื่อนำไปใช้ในการคำนวณเป็นค่าตัวแปรหุ่น และตัวแปรอีกตัวที่จำเป็นจะต้องนำมาวิเคราะห์ความถดถอยกับตัวแปรหุ่น คืออัตราความเสียหายซึ่งจำเป็นที่จะต้องกำหนดค่าอัตราความเสียหายที่จะได้มาจากแบบสอบถามเพื่อนำไปใช้สร้างสมการพยากรณ์เพื่อ

หาค่าความเสียหายโดยประมาณ และหลังจากนั้นทำการกำหนดค่าความน่าจะเป็นของอัตราความเสียหายที่จะเกิดขึ้น โดยมีรายละเอียดขั้นตอนดังนี้

3.1.1 วิเคราะห์ปัจจัยทางด้านอคติทางการรู้คิด

ในขั้นตอนนี้ได้ศึกษาปัจจัยที่เกี่ยวข้องจากงานวิจัยและตำราทางด้านจิตวิทยา ที่เกี่ยวข้องกับงานด้านวิศวกรรมซอฟต์แวร์ โดยจากงานวิจัยที่เกี่ยวข้องของ Kannan Mohan และ Radhika Jain ในปี 2008 ได้อธิบายถึงความสัมพันธ์ของอคติทางการรู้คิดไว้ทั้งหมดสามประเภท และจากงานวิจัยตำราทางด้านจิตวิทยาต่างอื่น ๆ ก็ได้กล่าวถึงอคติทางการรู้คิดเหล่านี้เอาไว้เช่นกัน โดยเป็นอคติทางการรู้คิดในเรื่องของการตัดสินใจแก้ปัญหาในปัญหาต่าง ๆ รวมถึงการแก้ปัญหาทางวิศวกรรมซอฟต์แวร์ด้วย ในงานวิจัยนี้ได้เลือกปัจจัยดังกล่าวสามปัจจัยมาเป็น ปัจจัยความเสี่ยงดังนี้

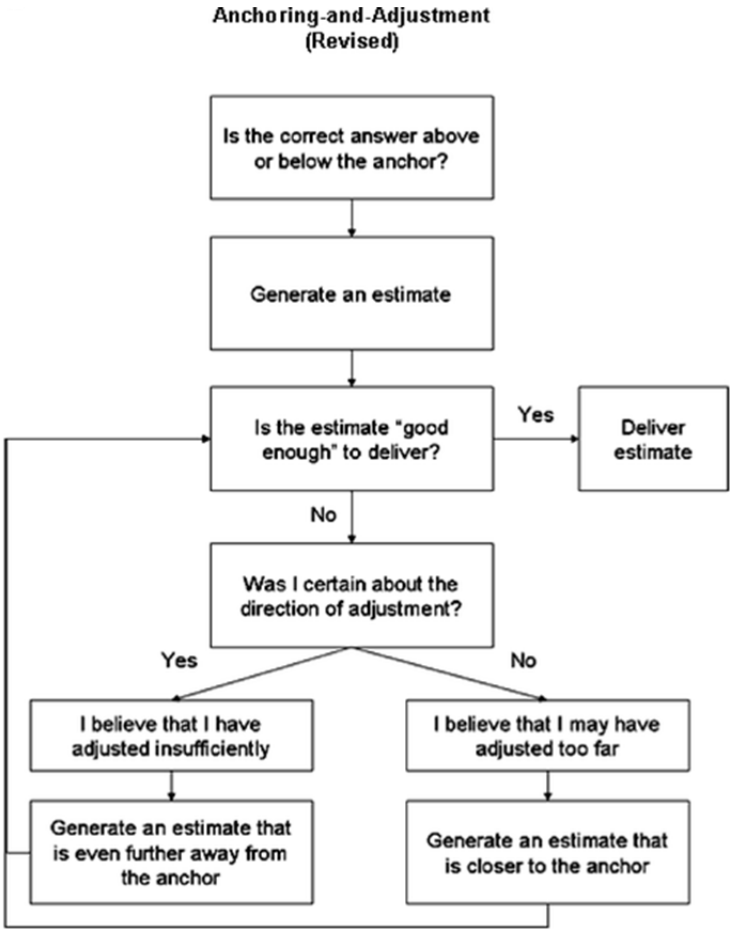
3.1.1.1 อคติ Anchoring and adjustment

เป็นการที่มนุษย์พยายามที่จะสร้างการประมาณการเกี่ยวกับปัญหาขึ้นมา เพื่อหาวิธีสำหรับการแก้ปัญหาต่าง ๆ ในการประมาณการเบื้องต้นนั้นถูกเรียกว่า Anchor หรือที่รู้จักกันในชื่อจุดอ้างอิง หรือตำแหน่งอ้างอิง เป็นตัวที่จะต้องนำไปปรับแต่งต่อไป หรือเรียกว่า Adjustment ซึ่งการ Anchoring หรือการประมาณการเบื้องต้นนั้นมีผลต่องาน หรือส่วนที่จะต้องการเปลี่ยนแปลง และทำการตัดสินใจ ยกตัวอย่างเช่น ในการที่จะเริ่มต้นงาน หรือทำโครงการใด ๆ ไม่ว่าจะเป็นการเริ่มต้นใหม่หรือทำการปรับปรุง หรือเปลี่ยนแปลงจากของเดิมนั้น จะต้องมีการประมาณการเบื้องต้นในตอนแรกของทุกงาน สำหรับผู้พัฒนาจะสามารถประมาณการจากการค้นคืนของความเชื่อเดิม ๆ หรือที่นานมาแล้ว นั่นคือจากประสบการณ์เก่าของการพัฒนา และรวมถึงประสบการณ์เฉพาะด้านนั้น ๆ ที่เคยมีมา แล้วมาเขียนโดยอนุมานถึงงานที่จะต้องทำ ซึ่งส่วนนี้อาจจะมีข้อผิดพลาดได้จากการประมาณการที่ผิดพลาด หรือประสบการณ์ที่มีไม่เพียงพอ เป็นต้น

ที่กล่าวไปเป็น Anchoring ที่ถูกสร้างจากภายใน สำหรับ Anchoring ที่ถูกสร้างจากภายนอก คือการศึกษาจากเอกสาร คู่มือระบบ หรือเอกสารความต้องการ ซึ่งการประมาณการในส่วนนี้ก็อาจจะผิดพลาดและก่อให้เกิดความเสียหายได้ ถ้าหากสิ่งเหล่านี้มีที่มาที่ไม่ถูกต้อง หรือถูกนำเอามาใช้อย่างไม่เหมาะสม เป็นต้น

จากปัญหาในการ Anchoring ที่อาจจะมีข้อผิดพลาดจากการประมาณจากความประสบการณ์ของตน หรือจากเอกสารที่มีการจัดทำที่บกพร่อง ทำให้ในการประมาณการเบื้องต้นทุกครั้งจะต้องทำการปรับแต่งให้เหมาะสมกับสถานการณ์ปัจจุบัน หรือความเป็นจริง ดังนั้นในหลายตำราและหลายงานวิจัย บางคราวก็จะเรียก Anchoring และ Adjustment คู่กัน [8] เพราะเป็นกระบวนการที่จะต้องทำต่อเนื่องกัน

อย่างไรก็ตามในหลายครั้ง การประมาณการเบื้องต้นแล้วมาผ่านกระบวนการปรับแต่งแล้ว ก็ยังไม่เหมาะสมเท่าที่ควร อาจจะต้องทำงานแบบวนรอบเพื่อให้มีการปรับแต่งให้เหมาะสมมากขึ้น ๆ ในแต่ละรอบเอง หรือแม้แต่จะต้องใช้อคติตัวอื่นเข้าร่วม เพื่อให้การปรับแต่งเหมาะสมมากยิ่งขึ้น โดยหากมีการปรับแต่งมากจนเกิดไปก็อาจจะทำให้ผลที่ออกมาคลาดเคลื่อนจากความเป็นจริงมากเกินไปอีกก็เป็นได้ และสำหรับอคติ Anchoring and adjustment อธิบายได้ดังภาพที่ 3.2



ภาพที่ 3.2 อธิบายลักษณะของอคติ Anchoring and adjustment [25]

3.1.1.2 อคติ Availability

เป็นการที่มนุษย์พยายามนึกถึงกรณีเฉพาะเจาะจงที่เกี่ยวข้องกับเรื่องที่เราจะตัดสินใจ มีการทดลองของ Tversky และ Kahneman ในปี 1973 ที่พยายามอธิบายลักษณะของอคติ Availability ไว้ดังนี้ คือ ผู้วิจัยถามผู้ร่วมวิจัยว่า ให้พิจารณาคำทั้งหมดในภาษาอังกฤษ ที่ตัวอักษร R ว่าอยู่ในตำแหน่งแรกของคำ เช่น rose, robot, rocket หรือว่าอยู่ในตำแหน่งที่สามของคำ เช่น care, strive, tarp อย่างไหนจะมีจำนวนคำมากกว่ากัน โดยผลการวิจัยออกมาพบว่า ผู้ร่วมวิจัยจำนวนสองในสามตอบว่า คำในภาษาอังกฤษ ที่ตัวอักษร R อยู่ในตำแหน่งแรก มีจำนวนคำมากกว่า ตัวอักษร R ที่อยู่ในคำในตำแหน่งที่สาม ซึ่งในความเป็นจริงแล้ว ตัวอักษร R มีปรากฏอยู่ในคำที่ตำแหน่งที่สาม มีจำนวนมากกว่า ตัวอักษร R ที่ปรากฏอยู่ในคำในตำแหน่งแรก จะเห็นได้ว่าความผิดพลาดนี้เกิดขึ้นมาจากการมี Availability ในเวลาตัดสินโดยดูว่า นึกถึงคำที่เริ่มต้นด้วยตัวอักษร R นั้นนึกได้ง่ายกว่าคำที่มีตัวอักษร R ปรากฏอยู่ในตำแหน่งที่สาม ปัญหาคือ การดึงข้อมูลมาใช้โดยมีความแตกต่างกัน ความจำของเราเก็บในรูปเสียงที่ขึ้นต้นในแต่ละคำ จึงทำให้นึกถึงคำที่เริ่มต้นด้วยตัวอักษรนั้น ๆ ได้ง่ายกว่า ดังนั้นคำที่ขึ้นต้นด้วยตัวอักษร R จึง available ในการนึกขึ้นมาได้ง่ายกว่า

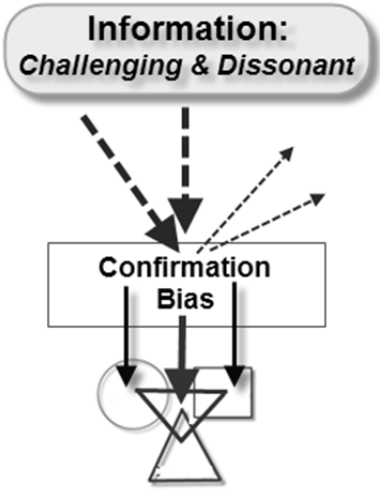
จากตัวอย่างที่ได้กล่าวไปเป็นเพียงข้อผิดพลาดเล็กน้อย ๆ ที่ไม่ค่อยมีอันตรายอันอาจก่อให้เกิดความเสียหายมาก แต่ถ้าหากมีอคตินี้กับสิ่งที่มีผลกระทบมาก ๆ ก็อาจจะก่อให้เกิดข้อผิดพลาดที่ร้ายแรงขึ้นมาได้ [7]

3.1.1.3 อคติ Confirmation

อคติในการยืนยันความเชื่อของตน โดยมนุษย์มักจะมีการยืนยันความคิด หรือความเชื่อของตน ซึ่งสามารถเกิดขึ้นได้ในหลายรูปแบบ เช่น เมื่อมนุษย์มีโอกาสที่จะหาข้อมูลใหม่ มนุษย์ก็จะมักที่จะหาข้อมูลที่ยืนยันความเชื่อของตนเองมากกว่าที่ค้านกับความเชื่อของตนเอง และเมื่อได้รับหลักฐานที่ตรงกับความเชื่อและไม่ตรงกับความเชื่อของตนเอง มนุษย์ก็มักจะให้ความสำคัญกับข้อมูลที่ตรงกับความเชื่อของตนเอง และไม่ให้ความสำคัญกับข้อมูลที่ไม่ตรงกับความเชื่อ หรืออาจจะมีการวิเคราะห์แบบมีอคติ หรือก็ทำการตีความใหม่ หรือไม่สนใจกับข้อมูลนั้นไปเลย

ตัวอย่างเช่น ในการคิดเลขอนุกรมว่าเลขอนุกรมชุดนี้ใช้กฎอะไร โดยให้อนุกรมตัวเลข 3 4 5 จริง ๆ แล้วก็ใช้กฎ การเพิ่มขึ้นทีละ 1 หน่วย แต่ผู้ตอบบางส่วนคิดว่า คำตอบน่าจะง่ายเกินไป เลยไป

พยายามหากฎที่ยากกว่า ซึ่งอาจจะทำให้หาคำตอบไม่ได้ ดังนั้น อคตินี้อาจจะเป็นปัญหาที่ทำให้ความคิดของเราผิดพลาดไป [7] ซึ่งสามารถอธิบายลักษณะของอคติ Confirmation ได้ดังภาพที่ 3.3



ภาพที่ 3.3 อธิบายลักษณะของอคติ Confirmation [26]

3.1.2 วิเคราะห์กระบวนการวิศวกรรมซอฟต์แวร์ที่สนใจ

ศึกษาหาความน่าจะเป็นในการมีอคติในกระบวนการก่อนที่จะได้มาซึ่งข้อกำหนดความต้องการซอฟต์แวร์ โดยนับตั้งแต่ขั้นตอนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ โดยในการหาความน่าจะเป็นที่จะเกิดขึ้นนี้ แบ่งออกเป็นสองขั้นตอนด้วยกัน ตามแบบจำลองข้อมูลสองขั้นตอน (Two-Stage Information Model) ดังนี้

1) ความน่าจะเป็นของการมีอคติที่จะเกิดขึ้นในระหว่างขั้นตอนการรวบรวมและกำหนดความต้องการซอฟต์แวร์เพื่อให้ได้ข้อกำหนดความต้องการซอฟต์แวร์ (Bias#1)
2) ความน่าจะเป็นจองการมีอคติที่จะเกิดขึ้นในระหว่างขั้นตอนการแปลงข้อกำหนดความต้องการซอฟต์แวร์มาเป็นข้อกำหนดการออกแบบซอฟต์แวร์ (Bias#2)

โดยแสดงการประยุกต์ใช้แบบจำลองข้อมูลสองขั้นตอนไว้ดังภาพที่ 3.4

ภาพที่ 3.4 แบบจำลองจากการประยุกต์ใช้แบบจำลองข้อมูลสองขั้นตอน

3.1.3 วิเคราะห์หาวิธีการคำนวณค่าความเสี่ยง

ศึกษาทฤษฎีที่เกี่ยวข้องกับการคำนวณหาค่าความเสี่ยง จากการนิยามค่าความเสี่ยง ในงานวิจัยนี้จะใช้เป็นค่าความคาดหวัง (Expected Value) ของความเสียหาย ค่าความเสี่ยงของเหตุการณ์หนึ่ง ๆ คำนวณได้จาก ความเสียหายหากเกิดเหตุการณ์นั้นขึ้น คูณด้วยความน่าจะเป็นที่จะเกิดเหตุการณ์นั้นขึ้น มีคำนิยามดังนี้

ความเสี่ยง = ความเสียหายเมื่อเกิดเหตุการณ์ × ความน่าจะเป็นของการเกิดเหตุการณ์

และจากคำนิยามนี้ สามารถเขียนเป็นสมการออกมาได้ดังนี้

$$Risk = R \times P \tag{3.1}$$

โดยกำหนดให้ Risk คือ ค่าความเสี่ยง

R คือ ความเสียหายในรูปอัตราความเสียหายเมื่อเกิดเหตุการณ์

P คือ ค่าความน่าจะเป็นที่จะเกิดเหตุการณ์

3.1.4 กำหนดปัจจัยการมีอคติให้เป็นค่าระดับ

ภายหลังจากที่ได้กำหนดกระบวนการที่สนใจและคาดว่าจะมีอคติเกิดขึ้นแล้ว ให้กำหนดจำนวนครั้งที่จะเกิดเหตุการณ์ที่มีอคติในแต่ละกระบวนการที่สนใจ นั่นคือใน Bias#1 และ Bias#2 โดยให้เป็นร้อยละของจำนวนครั้งที่เกิดขึ้นของอคติในการทำงานขั้นตอนนั้น ๆ เช่น ในการทำงานของเราใน 1 โครงการ ในส่วนของการเก็บรวบรวมและกำหนดความต้องการซอฟต์แวร์ (Bias#1) และการสร้างข้อกำหนดการออกแบบซอฟต์แวร์ (Bias#2) ถ้าทำงานในส่วนของการรวบรวมและกำหนดความต้องการซอฟต์แวร์ทั้งหมด 10 ครั้ง เกิดเหตุการณ์ซึ่งเป็นปัจจัยย่อยที่เป็นตัวแทนของการมีอคติแต่ละประเภทคิดเป็นร้อยละเท่าไรของจำนวน 10 ครั้ง ดังตารางที่ 3.1

ตารางที่ 3.1 ค่าระดับของจำนวนร้อยละของเหตุการณ์ที่เกิดขึ้น

จำนวนร้อยละที่เกิดเหตุการณ์ขึ้น	ค่าระดับ
0-20%	1
21-40%	2
41-60%	3
61-80%	4
81-100%	5

สำหรับค่าระดับของการมีอคติในแต่ละประเภทนั้น สามารถหาได้จากค่าเฉลี่ยค่าระดับของเหตุการณ์ซึ่งเป็นปัจจัยย่อยที่เป็นตัวแทนของการมีอคติแต่ละประเภท โดยที่อคติ Anchoring and adjustment เป็นอคติประเภทที่ 1 อคติ Availability เป็นอคติประเภทที่ 2 และอคติ Confirmation เป็นอคติประเภทที่ 3 ซึ่งอธิบายได้ดังตารางที่ 3.2

ตารางที่ 3.2 ตัวอย่างการคิดค่าเฉลี่ยของการมีอคติในแต่ละประเภท

ตัวอย่างเหตุการณ์ที่มีอคติประเภท Confirmation	ค่าระดับที่เกิดขึ้นของตัวอย่างเหตุการณ์
ตัวอย่างเหตุการณ์ที่ 1	X_{31}
ตัวอย่างเหตุการณ์ที่ 2	X_{32}
ตัวอย่างเหตุการณ์ที่ 3	X_{33}
ตัวอย่างเหตุการณ์ที่ 4	X_{34}
ตัวอย่างเหตุการณ์ที่ 5	X_{35}
ค่าเฉลี่ยของการมีอคติประเภท Confirmation	$\overline{X}_{ij}$

สามารถเขียนเป็นสมการได้ดังนี้

$$X_i = \bar{X}_{ij} \tag{3.2}$$

โดยที่ i = 1, 2, 3 แทนอคติแต่ละประเภท และ j = 1, 2, 3, ..., n แทนจำนวนตัวอย่างเหตุการณ์ที่มีอคติ โดยในท้ายที่สุดแล้วในแต่ละกระบวนการทั้งกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ (Software Requirements Gathering and Specification Process) และกระบวนการออกแบบซอฟต์แวร์ (Software Design Process) จะได้ X ทั้งหมดจำนวน 3 ตัว คือ X_1, X_2, X_3 ซึ่งแต่ละกระบวนการอาจจะกำหนดให้เป็น ReX_1, ReX_2, ReX_3 และ DsX_1, DsX_2, DsX_3 ก็ได้

3.1.5 การกำหนดค่าระดับของปัจจัยการมีอคติให้เป็นตัวแปรหุ่น

ภายหลังจากที่ได้กำหนดค่าระดับให้กับอคติแต่ละประเภทหรือปัจจัยหลัก ที่ได้มาจากค่าเฉลี่ยของปัจจัยย่อย ๆ ที่ถูกกำหนดไว้เป็นตัวอย่างเหตุการณ์ที่มีอคติในแต่ละประเภทแล้ว เนื่องด้วยค่าระดับที่ได้มาเป็นตัวแปรเชิงคุณภาพจึงจะต้องทำการแปลงค่าระดับเหล่านั้นเป็นตัวแปรชนิดหนึ่งที่ไว้ใช้แทนตัวแปรเชิงปริมาณ นั่นคือ ตัวแปรหุ่น (Dummy Variable) ด้วยวิธี Dummy Coding [20]

โดยตัวแปร ReX_1, ReX_2, ReX_3 และ DsX_1, DsX_2, DsX_3 ที่ได้มานั้นมีค่าที่เป็นไปได้อยู่ในช่วง 1 ถึง 5 เนื่องด้วยมาจากการเฉลี่ยของค่าระดับอยู่ที่ 1 ถึง 5 ยกตัวอย่างเช่นในตารางที่ 3.3

ตารางที่ 3.3 ตัวอย่างการคิดค่าเฉลี่ยของการมีอคติในแต่ละประเภท

ตัวอย่างเหตุการณ์ที่มีอคติประเภท Confirmation	ค่าระดับที่เกิดขึ้นของตัวอย่างเหตุการณ์
ตัวอย่างเหตุการณ์ที่ 1	5
ตัวอย่างเหตุการณ์ที่ 2	3
ตัวอย่างเหตุการณ์ที่ 3	4
ตัวอย่างเหตุการณ์ที่ 4	4
ตัวอย่างเหตุการณ์ที่ 5	2
ค่าเฉลี่ยของการมีอคติประเภท Confirmation	**3.6**

สำหรับตัวอย่างดังตารางค่าเฉลี่ยของการมีอคติประเภท Confirmation สมมติให้อยู่ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์หรือ ReX_3 มีค่าอยู่ในระดับเท่ากับ 3.6 และ

ก่อนที่จะนำมาทำเป็นตัวแปรหุ่นด้วยวิธี Dummy Coding นั้นให้ทำการปัดเศษค่าระดับที่เฉลี่ยออกมาได้ดังนี้

1) หากมีเศษมากกว่า 0.5 ให้ทำการปัดขึ้น
2) หากมีเศษน้อยกว่า 0.5 ให้ทำการปัดลง

ส่วนตัวอย่างดังกล่าวมีค่าเป็น 3.6 ซึ่งมีเศษเป็น 0.6 ซึ่งมากกว่า 0.5 ตรงกับเงื่อนไขที่ให้ทำการปัดเศษขึ้นดังนั้นค่าระดับใหม่ที่จะใช้ในการทำตัวแปรหุ่นจึงเท่ากับ 4 หรือสามารถเขียนเป็นสมการด้วยสูตรของ Microsoft Excel ได้ดังนี้

$$DpX_i = ROUND(pX_i) \quad (3.3)$$

โดยกำหนดให้ DpX_i แทนค่าระดับที่ผ่านการแปลงเพื่อใช้ในการทำตัวแปรหุ่น โดยมี p แทนกระบวนการที่สนใจ ได้แก่ Re, Ds และ i = 1, 2, 3 แทนอคติแต่ละประเภท ดังนั้นในตัวอย่างข้างต้นจึงสามารถเขียนได้ว่า $DReX_3 = 4$

สำหรับสาเหตุที่ต้องแปลงค่าอีกครั้งเนื่องจากตัวแปรหุ่น เป็นการที่นำตัวแปรพหุวิภาค (Polytomous Variable) ซึ่งเป็นตัวแปรที่มีจำนวนรายการมากกว่า 2 ลักษณะ หรือสามารถจำแนกออกเป็น g รายการ จะสามารถสร้างตัวแปรอิสระที่แตกต่างกันขึ้นมาได้จำนวน g-1 ตัวแปร [20] เช่น ในงานวิจัยนี้อคติแต่ละประเภท มีจำนวนรายการมากกว่า 2 ลักษณะ นั่นคือ มีลักษณะการเกิดหรือระดับการเกิดอยู่ใน 5 ลักษณะ คือ 1, 2, 3, 4, 5 หรือมี g = 5 รายการ ก็จะสามารถสร้างตัวแปรอิสระที่แตกต่างกันขึ้นมาได้จำนวน g-1 หรือนั่นคือ 5-1 = 4 ตัวแปร โดยที่ไม่ต้องใช้ตัวแปร 5 ตัวเนื่องจากจะมีตัวแปรใดตัวแปรหนึ่งถูกเลือกให้เป็นตัวแปรอ้างอิง (Reference Variable) อีกทั้งซึ่งจะทำให้เกิดความซ้ำซ้อนและไม่ได้สารสนเทศอะไรเพิ่มเติมและทำให้การคำนวณมีปัญหาและผลการวิเคราะห์ไม่สามารถให้ค่าเฉพาะที่คงที่ได้ [20]

โดยมีตัวแปรค่าระดับที่ปัดเศษแล้วของอคติทุกประเภทในทั้งสองกระบวนการที่จะนำไปสร้างเป็นตัวแปรหุ่น ดังนี้

$DReX_1$ = ค่าระดับเฉลี่ยที่ปัดเศษแล้วของอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์

$DReX_2$ = ค่าระดับเฉลี่ยที่ปัดเศษแล้วของอคติประเภท Availbility ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์

$DReX_3$ = ค่าระดับเฉลี่ยที่ปัดเศษแล้วของอคติประเภท Confirmation ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์

$DDsX_1$ = ค่าระดับเฉลี่ยที่ปัดเศษแล้วของอคติประเภท Anchoring and adjustment ในกระบวนการออกแบบซอฟต์แวร์

$DDsX_2$ = ค่าระดับเฉลี่ยที่ปัดเศษแล้วของอคติประเภท Availability ในกระบวนการออกแบบซอฟต์แวร์

$DDsX_3$ = ค่าระดับเฉลี่ยที่ปัดเศษแล้วของอคติประเภท Confirmation ในกระบวนการออกแบบซอฟต์แวร์

โดยจะยกตัวอย่างในกระบวนการเก็บรวบรวมและกำหนดความต้องการซอฟต์แวร์กำหนดให้ DpX_i เป็น $DReX_1$, $DReX_2$, $DReX_3$ โดยที่ในแต่ละรายการจะมีทั้งสิ้น 5 ลักษณะ คือ 1, 2, 3, 4, 5 โดยกำหนดให้ลักษณะที่ 1 (x_1) นั่นคือ การมีอคติเกิดขึ้นในการทำงาน 0-20% ของการทำงานทั้งหมด เป็นตัวแปรอ้างอิง ดังนั้นจึงจะเหลือตัวแปรอิสระอีก 4 ตัวแปรตามการสร้างตัวแปรอิสระที่แตกต่างกัน g-1 หรือนั่นคือ 5-1 = 4 ตัวแปร ดังนั้นจึงสรุปได้ว่า $DReX_1$, $DReX_2$, $DReX_3$ ต่างก็จะมีตัวแปรอิสระที่ผ่านการทำเป็นตัวแปรหุ่น 4 ตัว คือ (x_2, x_3, x_4, x_5) ซึ่งกำหนดชื่อให้แบ่งแยกตามประเภทของอคติได้ดังตารางที่ 3.4

ตารางที่ 3.4 ตัวอย่างตัวแปรหุ่นของตัวแปร $DReX_3$ ที่มี g=5 กลุ่ม

n	$DReX_3$	$DReX_3$ (ตัวแปรหุ่น)			
		$conx_2$	$conx_3$	$conx_4$	$conx_5$
1	2	**1**	0	0	0
2	4	0	0	**1**	0
3	5	0	0	0	**1**
4	3	0	**1**	0	0
5	1	0	0	0	0
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
n	n	n	n	n	n

จากตารางจะเห็นได้ว่า n ตัวที่ 1 ที่มี $DReX_3$ มีค่าเป็น 2 จะแปลงไปเป็นตัวแปรหุ่นที่ $conx_2$ มีค่าเท่ากับ 1 โดยที่ตัวแปรหุ่นตัวอื่น ๆ ของ n ตัวที่ 1 จะมีค่าเท่ากับ 0 โดยจะใช้หลักนี้กับ n ตัวอื่น ๆ ที่มีค่า $DReX_3$ อยู่ในช่วง 2, 3, 4, 5 แต่สำหรับ $DReX_3$ ที่มีค่าเท่ากับ 1 ซึ่งถูกเลือกไว้ให้เป็นตัวแปรอ้างอิง (Refernce Variable) จะมีค่าเป็น 0 ทุกช่องของตัวแปรหุ่น โดยจากตัวอย่างที่ยกไปก่อนหน้านี้ที่แปลง ReX_3 = 3.6 ไปเป็น $DReX_3$ = 4 นั้น ถ้าใช้หลักการนี้ก็จะได้ตัวแปรหุ่น $conx_2$, $conx_3$, $conx_4$, $conx_5$ มีค่าเป็น 0, 0, 1, 0 ตามลำดับ และสำหรับ $DReX_1$, $DReX_2$ ก็ใช้หลักการตัวแปรหุ่นนี้ในการแปลงเช่นกัน

เช่นเดียวกันกับในกระบวนการออกแบบซอฟต์แวร์ที่ไม่ได้ยกตัวอย่างไว้ทั้ง DsX_1, DsX_2, DsX_3 ก็ให้ใช้หลักการตัวแปรหุ่นนี้เช่นกัน ทั้งนี้วัตถุประสงค์ของการใช้วิธี Dummy Coding นี้ก็เพื่อที่จะแปลงตัวแปรเชิงคุณภาพไปเป็นตัวแปรหุ่นเพื่อนำไปเป็นตัวแปรต้นในการวิเคราะห์ความถดถอยกับความเสียหายของซอฟต์แวร์ในสร้างสมการการพยากรณ์ต่อไป

3.1.6 การกำหนดค่าอัตราความเสียหาย

อัตราความเสียหายต่อโครงการพัฒนาซอฟต์แวร์ที่สนใจ คืออัตราความเสียหายในด้านของความน่าเชื่อถือของซอฟต์แวร์ที่พัฒนา โดยเป็นอัตราความเสียหายที่เกิดขึ้นหาได้จากจำนวนครั้งของ

ความผิดพลาดที่เกิดขึ้นในซอฟต์แวร์ในระยะเวลาหนึ่งเดือน ซึ่งค่านี้จะได้มาจากแบบสอบถาม ซึ่งเป็นอัตราความเสียหายของแต่ละโครงการ กำหนดให้เป็นค่า Y เป็นข้อมูลเชิงปริมาณ มีหน่วยเป็น จำนวนครั้ง โดยจะจำแนกจำนวนครั้งของความเสียหายดังกล่าวออกเป็นแต่ละระดับความรุนแรงของความเสียหายว่ามีจำนวนเป็นเท่าไรใน 5 ระดับ [27] ดังนี้

1) ระดับวิกฤต (Critical) คือ ข้อผิดพลาดที่ส่งผลต่อความล้มเหลวของระบบซอฟต์แวร์ ระบบย่อย โปรแกรมหรือโมดูล โดยสมบูรณ์
2) ระดับมาก (Major) คือ ข้อผิดพลาดที่ส่งผลต่อความล้มเหลวของตัวระบบซอฟต์แวร์ ระบบย่อย โปรแกรมหรือโมดูลเอง แต่ไม่ส่งผลต่อส่วนงานอื่น ๆ ข้อผิดพลาดนี้สามารถยอมรับได้หากมีการประมวลผลด้วยวิธีอื่นที่จะทำให้ได้ผลลัพธ์ตามที่ต้องการ แต่ยังคงต้องแก้ไขข้อผิดพลาดนั้น
3) ระดับปานกลาง (Average) คือ ข้อผิดพลาดที่ไม่ได้ส่งผลต่อความล้มเหลว แต่ทำให้ระบบทำงานได้ไม่ถูกต้องสมบูรณ์ ผลลัพธ์ไม่สอดคล้องกัน หรือข้อผิดพลาดลดทอนการใช้งานระบบ
4) ระดับน้อย (Minor) คือ ข้อผิดพลาดที่ไม่ได้ก่อให้เกิดความล้มเหลว ไม่ได้ลดทอนการใช้งานระบบ และยังสามารถให้ผลลัพธ์ที่ต้องการจากการประมวลผลได้ด้วยการแก้ไขข้อผิดพลาดที่ไม่ยุ่งยาก
5) ระดับที่ละเว้นได้ (Exception) คือ ข้อผิดพลาดที่เป็นผลมาจากความไม่สอดคล้องกันกับมาตรฐาน มีความเกี่ยวเนื่องกับคุณลักษณะภายนอกของระบบ เช่น ความสวยงาม สีสัน หรือเป็นความต้องการเพื่อปรับปรุงให้ดูดีขึ้น ข้อผิดพลาดในระดับนี้อาจจะผ่อนผันหรือแม้แต่ละเว้นการแก้ไขไปได้

ซึ่งจะนำค่า Y นี้ไปใช้ในการวิเคราะห์ความถดถอยกับตัวแปรหุ่นที่ได้เตรียมไว้ เพื่อที่ต้องการจะทราบสมการในการพยากรณ์ค่าอัตราความเสียหายโดยประมาณ ($\hat{Y}$)

สำหรับค่า $\hat{Y}$ ที่ได้เป็นค่าตัวแปรเชิงปริมาณซึ่งในแต่ละกระบวนการจะมีค่า $\hat{Y}$ ทั้งหมด 3 ค่า จากอคติทั้ง 3 ประเภท ซึ่งจะนำเอาค่า $\hat{Y}$ นี้ไปหาค่าเฉลี่ย ($\bar{Y}$) เพื่อจะให้ได้ทราบค่าอัตราความเสียหายโดยเฉลี่ยจากการมีอคติทั้ง 3 ประเภทนี้ โดยค่าเฉลี่ยที่ได้นั้นจะนำไปจำแนกออกเป็นระดับความรุนแรงของความเสียหาย 5 ระดับตามสัดส่วนที่ได้กำหนดไว้จากจำนวนที่จำแนกออกมา

3.1.7 การวิเคราะห์ความถดถอยเพื่อสร้างสมการพยากรณ์อัตราความเสียหายโดยประมาณ

จากหัวข้อ 3.1.1 จนถึง 3.1.6 เป็นการตระเตรียมตัวแปรต่าง ๆ สำหรับใช้ในการวิเคราะห์ความถดถอยซึ่งมีตัวแปรหุ่นที่แทนค่าร้อยละของการมีอคติในการทำงานกับค่าอัตราความเสียหายในหนึ่งเดือนเมื่อมีเหตุการณ์ที่มีอคติเกิดขึ้นในระหว่างการทำงานจากกลุ่มตัวอย่าง n ตัวซึ่งแทนโครงการ n โครงการ โดยจำเป็นจะต้องใช้วิธี Enter สำหรับการนำเอาตัวแปรหุ่นทุกตัวเข้าไปวิเคราะห์กับอัตราความเสียหาย (Y) ซึ่งตัวแปรต่าง ๆ ที่จะนำเข้าวิเคราะห์ความถดถอย แสดงเป็นตัวหนาดังตารางที่ 3.5

ตารางที่ 3.5 ตัวอย่างตัวแปรต่าง ๆ ของ Confirmation bais ที่จะนำเข้าวิเคราะห์ความถดถอย

No.	$DReX_3$	$DReX_3$ (ตัวแปรหุ่น)				Y
		$conx_2$	$conx_3$	$conx_4$	$conx_5$	
1	2	**1**	0	0	0	**4**
2	4	0	0	**1**	0	**3**
3	5	0	0	0	**1**	**3**
4	3	0	**1**	0	0	**3**
5	1	0	0	0	0	**1**
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
n	n	n	n	n	n	n

สำหรับสิ่งที่จะได้ในการวิเคราะห์ความถดถอยหรือการได้สัมประสิทธ์หน้าตัวแปรจากตาราง 4.1-4.6 มาสร้างเป็นสมการพยากรณ์ความเสียหายโดยประมาณจากการมีอคติ ดังนี้

$$\hat{Y} = a + b_1X_1 + b_2X_2 + \cdots + b_kX_k \tag{3.5}$$

และโดยจากอคติทั้งสามประเภทจะได้สมการดังนี้

$$\hat{Y}_{anc} = a + b_2ancx_2 + b_3ancx_3 + b_4ancx_4 + b_5ancx_5$$

$$\hat{Y}_{ava} = a + b_2avax_2 + b_3avax_3 + b_4avax_4 + b_5avax_5$$

$$\hat{Y}_{con} = a + b_2conx_2 + b_3conx_3 + b_4conx_4 + b_5conx_5$$

โดยหลังจากที่ได้ค่าพยากรณ์ความเสียหายโดยประมาณจากอคติทั้งสามประเภทแล้ว ก็จะสามารถหาอัตราเสียหายโดยเฉลี่ยจากอคติทั้งสามประเภทได้จากสมการ

$$\bar{Y} = \frac{\hat{Y}_{anc}+\hat{Y}_{ava}+\hat{Y}_{con}}{3} \quad (3.6)$$

3.1.8 คำนวณค่าอัตราความเสียหายประมาณจากสมการพยากรณ์

จากสมการ 3.6 นำเอาค่า $\bar{Y}$ จะต้องนำไปจำแนกออกเป็น 5 ระดับความรุนแรง กล่าวคือ ทำให้ค่าอัตราความเสียหายเฉลี่ยรวมจากอคติสามประเภท แยกออกเป็นค่าอัตราความเสียหายจากอคติแต่ละประเภทออกเป็นในแต่ละระดับความรุนแรง ด้วยวิธีการบัญญัติไตรยางค์ จะต้องทำบัญญัติไตรยางค์ 2 ชั้น

โดยการทำบัญญัติไตรยางค์ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ ดังนี้

1) ทำบัญญัติไตรยางค์จำนวนความเสียหายของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์จากแบบสอบถามในแต่ละระดับความรุนแรงให้มีสัดส่วนเป็น 100 เช่น ในระดับความรุนแรง Critical, Major, Average, Minor, Exception มีจำนวนครั้งความเสียหายดังนี้ 3, 7, 27, 29, 28 ตามลำดับ รวมกันได้ค่า Y จากแบบสอบถามเท่ากับ 94 โดยกำหนดให้มีค่าเท่ากับ 100 แล้วจำนวนครั้งความเสียหาย 3, 7, 27, 29, 28 จะคิดเป็นเท่าไร

   วิธีคิด ถ้าค่าความเสียหายรวมเท่ากับ 94 ครั้ง มีค่าเท่ากับ 100 แล้ว

   ค่าความเสียหาย Critical เท่ากับ 3 ครั้ง จะมีค่าเท่ากับ (3x100)/94 หรือร้อยละ 3.19

   ค่าความเสียหาย Major เท่ากับ 7 ครั้ง จะมีค่าเท่ากับ (7x100)/94 หรือร้อยละ 7.45

   ค่าความเสียหาย Average เท่ากับ 27 ครั้ง จะมีค่าเท่ากับ (27x100)/94 หรือร้อยละ 28.72

   ค่าความเสียหาย Minor เท่ากับ 29 ครั้ง จะมีค่าเท่ากับ (29x100)/94 หรือร้อยละ 30.85

   ค่าความเสียหาย Exception เท่ากับ 28 ครั้ง จะมีค่าเท่ากับ (28x100)/94 หรือร้อยละ 29.79

   โดยค่าที่ได้จากบัญญัติไตรยางศ์ในชั้นนี้จะถูกกำหนดให้เป็นค่าคงที่ k เพื่อใช้ในสมการที่ 4.7 ในการทำบัญญัติไตรยางค์ครั้งที่ 2

2) ทำบัญญัติไตรยางค์จำนวนความเสียหายของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ในร้อยส่วนของแต่ละระดับความรุนแรงให้มีสัดส่วนเป็นอัตราความเสียหายโดยประมาณที่ได้จากสมการพยากรณ์ เช่น ในระดับความรุนแรง Critical, Major, Average, Minor, Exception มีจำนวนครั้งความเสียหายในร้อยส่วนดังนี้ 3.19, 7.45, 28.72, 30.85, 29.79 ตามลำดับ ถ้าหากค่า $\bar{Y}$ จากการพยากรณ์เท่ากับ 4.00 จะมีความเสียหายในแต่ละระดับเป็นเท่าไร ซึ่งสามารถหาได้จากสมการที่ 4.7 โดยมีวิธีคิดดังนี้

วิธีคิด ถ้าค่าอัตราความเสียหายเฉลี่ย ($\bar{Y}$) ร้อยละ 100 มีค่าเท่ากับ 4.00 ครั้ง แล้ว

ค่าความเสียหาย Critical ร้อยละ 3.19 จะมีค่าเท่ากับ (3.19x4.00)/100 = 0.13 ครั้ง

ค่าความเสียหาย Major ร้อยละ 7.45 จะมีค่าเท่ากับ (7.45x4.00)/100 = 0.30 ครั้ง

ค่าความเสียหาย Average ร้อยละ 28.72 จะมีค่าเท่ากับ (28.72x4.00)/100 = 1.15 ครั้ง

ค่าความเสียหาย Minor ร้อยละ 30.85 จะมีค่าเท่ากับ (30.85x4.00)/100 = 1.23 ครั้ง

ค่าความเสียหาย Exception ร้อยละ 29.79 จะมีค่าเท่ากับ (29.79x4.00)/100 = 1.19 ครั้ง

และการทำบัญญัติไตรยางค์ในกระบวนการออกแบบซอฟต์แวร์ ดังนี้

1) ทำบัญญัติไตรยางค์จำนวนความเสียหายของกระบวนการออกแบบซอฟต์แวร์จากแบบสอบถามในแต่ละระดับความรุนแรงให้มีสัดส่วนเป็น 100 เช่น ในระดับความรุนแรง Critical, Major, Average, Minor, Exception มีจำนวนครั้งความเสียหายดังนี้ 2, 12, 37, 32, 6 ตามลำดับ รวมกันได้ค่า Y จากแบบสอบถามเท่ากับ 89 โดยกำหนดให้มีค่าเท่ากับ 100 แล้วจำนวนครั้งความเสียหาย 2, 12, 37, 32, 6 จะคิดเป็นเท่าไร

วิธีคิด ถ้าค่าความเสียหายรวมเท่ากับ 89 ครั้ง มีค่าเท่ากับ 100 แล้ว

ค่าความเสียหาย Critical เท่ากับ 2 ครั้ง จะมีค่าเท่ากับ (2x100)/89 หรือร้อยละ 2.25

ค่าความเสียหาย Major เท่ากับ 12 ครั้ง จะมีค่าเท่ากับ (12x100)/89 หรือร้อยละ 13.48

ค่าความเสียหาย Average เท่ากับ 37 ครั้ง จะมีค่าเท่ากับ (37x100)/89 หรือร้อยละ 41.57

ค่าความเสียหาย Minor เท่ากับ 32 ครั้ง จะมีค่าเท่ากับ (32x100)/89 หรือร้อยละ 35.96

ค่าความเสียหาย Exception เท่ากับ 6 ครั้ง จะมีค่าเท่ากับ (6x100)/89 หรือร้อยละ 6.74

โดยค่าที่ได้จากบัญญัติไตรยางศ์ในชั้นนี้จะถูกกำหนดให้เป็นค่าคงที่ k เพื่อใช้ในสมการที่ 4.7 ในการทำบัญญัติไตรยางค์ครั้งที่ 2

2) ทำบัญญัติไตรยางค์จำนวนความเสียหายของกระบวนการออกแบบซอฟต์แวร์ในร้อยส่วนของแต่ละระดับความรุนแรงให้มีสัดส่วนเป็นอัตราความเสียหายโดยประมาณที่ได้จากสมการพยากรณ์ เช่น ในระดับความรุนแรง Critical, Major, Average, Minor, Exception มีจำนวนครั้งความเสียหายในร้อยส่วนดังนี้ 2.25, 13.48, 41.57, 35.96, 6.74 ตามลำดับ ถ้าหากค่า $\bar{Y}$ จากการพยากรณ์เท่ากับ 4.00 จะมีความเสียหายในแต่ละระดับเป็นเท่าไร ซึ่งสามารถหาได้จากสมการที่ 4.7 โดยมีวิธีคิดดังนี้

วิธีคิด ถ้าค่าอัตราความเสียหายเฉลี่ย ($\bar{Y}$) ร้อยละ 100 มีค่าเท่ากับ 4.00 ครั้ง แล้ว

ค่าความเสียหาย Critical ร้อยละ 2.25 จะมีค่าเท่ากับ (2.25x4.00)/100 = 0.09 ครั้ง

ค่าความเสียหาย Major ร้อยละ 13.48 จะมีค่าเท่ากับ (13.48x4.00)/100 = 0.54 ครั้ง

ค่าความเสียหาย Average ร้อยละ 41.57 จะมีค่าเท่ากับ (41.57x4.00)/100 = 1.66 ครั้ง

ค่าความเสียหาย Minor ร้อยละ 35.96 จะมีค่าเท่ากับ (35.96x4.00)/100 = 1.44 ครั้ง

ค่าความเสียหาย Exception ร้อยละ 6.74 จะมีค่าเท่ากับ (6.74x4.00)/100 = 0.27 ครั้ง

ซึ่งค่าความเสียหายทั้ง 5 ค่าที่ได้จากการทำบัญญัติไตรยางค์ครั้งที่ 2 นี้จะนำไปเป็นค่า R_i (Loss per Accident) โดยที่ i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception ในสมการความเสี่ยงที่จะนำไปคำนวณต่อกับค่าความน่าจะเป็นที่จะเกิดเหตุการณ์ที่มีความเสียหายเกิดขึ้น หรือ P_i (Probability per Accident) ต่อไป

3.1.9 คำนวณค่าความน่าจะเป็น

การหาค่าความน่าจะเป็น เป็นส่วนสำคัญที่จะประเมินค่าความเสี่ยง โดยจะหาค่าความน่าจะเป็นของอัตราความเสียหายจากแบบสอบถามที่รวบรวมมาจากกลุ่มตัวอย่างในการตอบคำถามที่ช่วยในการหาค่าความน่าจะเป็นของเหตุการณ์ที่จะเกิดการมีอคตินั้น ๆ ขึ้น โดยให้คำจำกัดความข้ออคตินั้น ๆ และยกสถานการณ์ที่เป็นตัวอย่างของอคติตัวนั้น ๆ ให้ผู้ร่วมวิจัยเข้าใจและสามารถนำไปจับคู่กับความเป็นจริงที่เกิดเหตุการณ์เหล่านี้ขึ้นได้

ซึ่งในกรณีนี้จะหาค่าความน่าจะเป็นที่จะเกิดเหตุการณ์จากการแจกแจงความน่าจะเป็นแบบปัวซอง ซึ่งเป็นการแจกแจงความน่าจะเป็นตัวแปรสุ่มแบบไม่ต่อเนื่องชนิดหนึ่ง นั่นคือเหตุการณ์ที่เกิดขึ้นแต่ละในช่วงระยะเวลาหนึ่งไม่ขึ้นต่อกัน โดยเป็นการแจกแจงที่อธิบายถึงจำนวนครั้งของเหตุการณ์หรือจำนวนสิ่งที่สนใจที่เกิดขึ้นในช่วงเวลาที่กำหนด โดยทราบค่าเฉลี่ย ให้จำนวนครั้งของสิ่งที่สนใจเป็นอัตราความเสียหายของซอฟต์แวร์ ในช่วงเวลาที่กำหนดนั้นกำหนดไว้ให้เป็นช่วงระยะเวลาหนึ่งเดือน โดยอัตราความเสียหายที่เกิดขึ้นในระยะเวลาหนึ่งเดือนไม่ขึ้นต่ออัตราความเสียหายที่จะเกิดในช่วงระยะเวลาหนึ่งเดือนถัดไป

สมมติว่าในช่วงเวลาหนึ่ง เหตุการณ์ E เกิดขึ้น μ ครั้ง ถ้าสังเกตดู ว่าอันที่จริงแล้ว จะเกิดเหตุการณ์ E ขึ้นกี่ครั้ง โดยให้ X เป็นจำนวนครั้งที่จะเกิดเหตุการณ์ดังกล่าว จะได้ว่าความน่าจะเป็นที่จะเกิดเหตุการณ์ E เป็นจำนวน x ครั้ง เขียนเป็นสัญลักษณ์ได้ดังนี้ P(X = x) [18], [19]

ฟังก์ชันความน่าจะเป็นแบบปัวซองของ X เป็นดังนี้

$$P(x;\mu) = P(X = x) = \frac{e^{-\mu}\mu^{x}}{X!}; x = 0,1,2,\dots,n \tag{3.7}$$

หรือเขียนได้ว่า $X \sim Poisson\ (\mu)$

กำหนดให้ $\mu = m$ จะได้

$$P(x;m) = P(X = x) = \frac{e^{-m}m^{x}}{X!}; x = 0,1,2,\dots,n \tag{3.8}$$

โดยกำหนดให้ e ≈ 2.71828...

m = จำนวนสิ่งที่สนใจที่เกิดขึ้นโดยเฉลี่ยในช่วงเวลาหรือในพื้นที่หรือขอบเขตที่กำหนดและสนใจเฉพาะความน่าจะเป็นของการเกิดเหตุการณ์ที่มีความเสียหายตั้งแต่ 1 ครั้งขึ้นไป นั่นคือ $P(X \geq 1)$ จากสมการที่ 3.8 จะได้ว่า

$$P(X \geq 1) = 1 - P(X = 0)$$

$$= 1 - \frac{e^{-m}m^0}{0!}$$

$$= 1 - e^{-m} \quad (3.9)$$

ให้ P เป็นความน่าจะเป็นที่จะเกิดเหตุการณ์ E ที่มีความเสียหายจากการมีอคติ จะเกิดเหตุการณ์ E ขึ้นกี่ครั้ง โดยให้ X เป็นจำนวนครั้งที่จะเกิดเหตุการณ์ดังกล่าว จะได้ว่าความน่าจะเป็นที่จะเกิดเหตุการณ์ E เป็นจำนวน x ครั้ง เขียนเป็นสัญลักษณ์ได้ดังนี้ P(X = x) โดยความน่าจะเป็นที่จะเกิดความเสียหายอย่างน้อย 1 ครั้ง นั่นคือ $P(X \geq 1)$ จะได้ว่า

$$P_i = 1 - e^{-m_i} \quad (3.10)$$

โดยกำหนดให้ e ≈ 2.71828...

m = อัตราความเสียหายโดยเฉลี่ยจากแบบสอบถามในช่วงเวลาหนึ่งเดือน

i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception

สำหรับสมการหาความน่าจะเป็นแบบปัวซองนี้ จะต้องหาค่า m มาจากแบบสอบถาม โดยค่า m ที่จะนำมาใช้นี้เป็นค่าเฉลี่ย Y ที่ได้จากแบบสอบถาม โดยไม่ได้เฉลี่ยมาจากค่า Y รวมทั้งหมดของระดับความรุนแรง 5 ระดับ แต่จะใช้ค่า Y ที่จำแนกออกเป็นแต่ละระดับแล้วเฉลี่ยออกมา ดังนั้น จึงจะได้ค่า m ทั้งหมด 5 ค่า ที่จะนำไปใช้ในสมการปัวซอง ดังนั้นค่าความน่าจะเป็นจากสมการปัวซอง ก็จะได้เป็นค่าความน่าจะเป็นของความเสียหายใน 5 ระดับ โดยค่า m ที่นำไปใช้แสดงด้วยตัวหนา แสดงได้ดังตารางที่ 3.6

ตารางที่ 3.6 ตัวอย่างค่า m ที่จะใช้ในสมการปัวซอง

No.	จำนวนครั้งที่เกิดระดับความรุนแรง					รวม
	Critical	Major	Average	Minor	Exception	
1	1	0	1	1	1	4
2	0	1	1	1	0	3
3	0	0	1	1	1	3
4	1	0	0	1	1	3
5	0	0	1	0	0	1
	0.40	**0.20**	**0.80**	**0.80**	**0.60**	2.80

3.1.10 การคำนวณค่าความเสี่ยง

หลังจากที่ได้ค่าความเสียหายหรือ R ทั้ง 5 ค่าจากหัวข้อ 3.1.8 และค่าความน่าจะเป็นที่จะเกิดความเสียหายหรือ P ทั้ง 5 ค่าแล้วจากหัวข้อ 3.1.9 นำค่า R และ P ไปใช้ในสมการความเสี่ยงในหัวข้อที่ 3.1.2 โดยกำหนดให้

$$Risk_i = R_i \times P_i \tag{3.11}$$

โดยกำหนดให้ i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้

5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception

ดังนั้นจึงจะได้ค่าความเสี่ยง (Risk) ทั้งหมด 5 ค่า ของความรุนแรงใน 5 ระดับ เพราะฉะนั้นความเสี่ยงรวมของอัตราความเสียหายที่มีความรุนแรงทั้ง 5 ระดับ หรือความเสี่ยงของอัตราความเสียหายรวมของแต่ละกระบวนการจึงจะเท่ากับ ผลรวมของ $Risk_i$ ดังนี้

$$pRisk = \sum_{i=1}^{5} Risk_i \tag{3.12}$$

โดยที่ p = Re, Ds นั่นคือกระบวนการที่สนใจในการหาความเสี่ยงได้แก่ กระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์

3.1.11 คำนวณมูลค่าความเสียหายจากการมีความเสี่ยง

สำหรับความเสียหายในแต่ละระดับความรุนแรงนั้น ในแต่ละองค์กรอาจจะตีความความเสียหายของแต่ระดับต่างกัน โดยเฉพาะในด้านตัวเงิน ยกตัวอย่างเช่น ในองค์กรที่ผลิตซอฟต์แวร์บริหารจัดการการบิน ซอฟต์แวร์นั้น จะยอมรับให้มีระดับความเสียหายในระดับวิกฤตได้น้อยมาก ซึ่งถ้าหากเกิดความเสียหายในระดับวิกฤติขึ้นมาแล้วจะเกิดการสูญเสียที่คิดเป็นตัวเงินต่อผู้โดยสารหนึ่งคนตามอนุสัญญากรุงวอร์ซอ [28] ได้กำหนดขอบเขตความรับผิดของผู้ขนส่งไว้มีมูลค่าเทียบเป็นเงินไทยไม่เกิน 2,000,000 บาท (250,000 ฟรังก์ฝรั่งเศส (FRF) [29]) หากมีผู้โดยสารเครื่องบินโดยเฉลี่ยประมาณ 400 คน จะคิดเป็นเงินโดยเฉลี่ย 800 ล้านบาท ทั้งนี้ยังไม่ได้คิดถึงความสูญเสียอื่น ๆ อีกเช่น ค่าเครื่องบิน ชื่อเสียง เป็นต้น โดยให้ค่าความเสียหายในระดับอื่น ๆ ที่สามารถคิดเป็นตัวเงินได้ และรวมเข้าไปด้วยนั้น สมมติว่าคิดเป็น 1,000 ล้านบาทต่อความเสียหายที่เกิดขึ้นหนึ่งครั้ง

ดังนั้น ถ้ามีค่าความเสียหายรวมในทุกระดับความรุนแรงที่คำนวณได้จากแบบจำลองนี้เท่ากับ 3.22 ครั้งต่อเดือน นั่นหมายความว่าอาจจะมีความสูญเสียทางด้านตัวเงินจากความเสียหายในทุกระดับคิดเป็นเงิน 3.22 ครั้งต่อเดือน คูณด้วย 1,000 ล้านบาท เท่ากับ 3,220 ล้านบาทต่อเดือน โดยมีความเสี่ยงเป็น 1.17 ซึ่งถ้าเทียบจากตารางเปรียบเทียบความเสี่ยงทั้งหมดที่คำนวณได้จากแบบจำลองนี้ (ตารางที่ 4.53) แล้ว พบว่ามีความเสี่ยงอยู่ในระดับปานกลาง แต่หากจะแยกดูในระดับความรุนแรงวิกฤติเพียงอย่างเดียว ซึ่งมีค่าความเสียหายที่คำนวณได้เป็น 0.10 จาก 3.22 ครั้งต่อเดือน นั่นหมายความว่าอาจจะมีความสูญเสียทางด้านตัวเงินจากความเสียหายในทุกระดับคิดเป็นเงิน 0.10 ครั้งต่อเดือน คูณด้วย 800 ล้านบาท เท่ากับ 80 ล้านบาทต่อเดือน

หรือสามารถแสดงการคำนวณความสูญเสียทางด้านตัวเงินจากความเสียหายในแต่ละระดับความรุนแรงได้จากสมการ

$$Loss\ amounts\ per\ month = LA_i \times R_i \tag{3.13}$$

โดยที่ LA_i = ค่า Loss amounts หรือมูลค่าความเสียหายทางด้านตัวเงินต่อความเสียหายหนึ่งครั้ง

R_i = ค่าความเสียหายที่ได้จากการประมาณและจำแนกออกเป็นแต่ละระดับความรุนแรง

i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception

และสามารถแสดงการคำนวณความสูญเสียทางด้านตัวเงินจากความเสียหายในทุกระดับความรุนแรงได้จากสมการ

$$\text{Summary of loss amounts per month} = \sum_{i=1}^{5} LA_i \times \overline{Y} \quad (3.14)$$

โดยที่ LA_i = ค่า Loss amounts หรือมูลค่าความเสียหายทางด้านตัวเงินต่อความเสียหายหนึ่งครั้ง

$\overline{Y}$ = ค่าความเสียหายรวมที่ได้จากการประมาณ

i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception

3.2 การสร้างแบบสอบถาม

แบบสอบถามเป็นส่วนสำคัญในการที่จะให้ได้มาซึ่งข้อมูลในการนำมาสร้างแบบจำลองวิเคราะห์ความเสี่ยงข้างต้น โดยในแบบสอบถามแบ่งออกเป็นทั้งสิ้น 4 ส่วน ดังนี้

1) ส่วนอธิบายแบบสอบถาม แสดงวัตถุประสงค์และรายละเอียดของการทำแบบสอบถาม
2) แบบสอบถามข้อมูลส่วนบุคคล สำหรับสอบถามข้อมูลของกลุ่มตัวอย่าง อย่างพอสังเขป
3) แบบสอบถามสำหรับการพบอคติในการรวบรวมและกำหนดความต้องการซอฟต์แวร์
4) แบบสอบถามสำหรับการพบอคติในกระบวนการออกแบบซอฟต์แวร์

สำหรับแนวคิดและวิธีสร้างแบบสอบถามเพื่อประเมินความเสี่ยงจากการมีอคติในทั้งสองกระบวนการนั้น จะแยกอธิบายเป็นแต่ละส่วนของแบบสอบถาม ดังนี้

3.2.1 ส่วนอธิบายแบบสอบถาม

เป็นการแสดงวัตถุประสงค์และรายละเอียดของการทำแบบสอบถาม ว่าแบบสอบถามนี้ถูกสร้างมาขึ้นเพื่อวัตถุประสงค์ใด โดยจะบอกถึงจำนวนแบบสอบถามและจำนวนหน้าให้ผู้ตอบแบบสอบถามได้ทราบ และบอกถึงลักษณะการตอบแบบสอบถามว่า เป็นอัตนัย หรือเป็นปรนัย หรือมีการกรอกในส่วนใดที่เน้นเป็นพิเศษ เป็นต้น

3.2.2 แบบสอบถามข้อมูลส่วนบุคคล

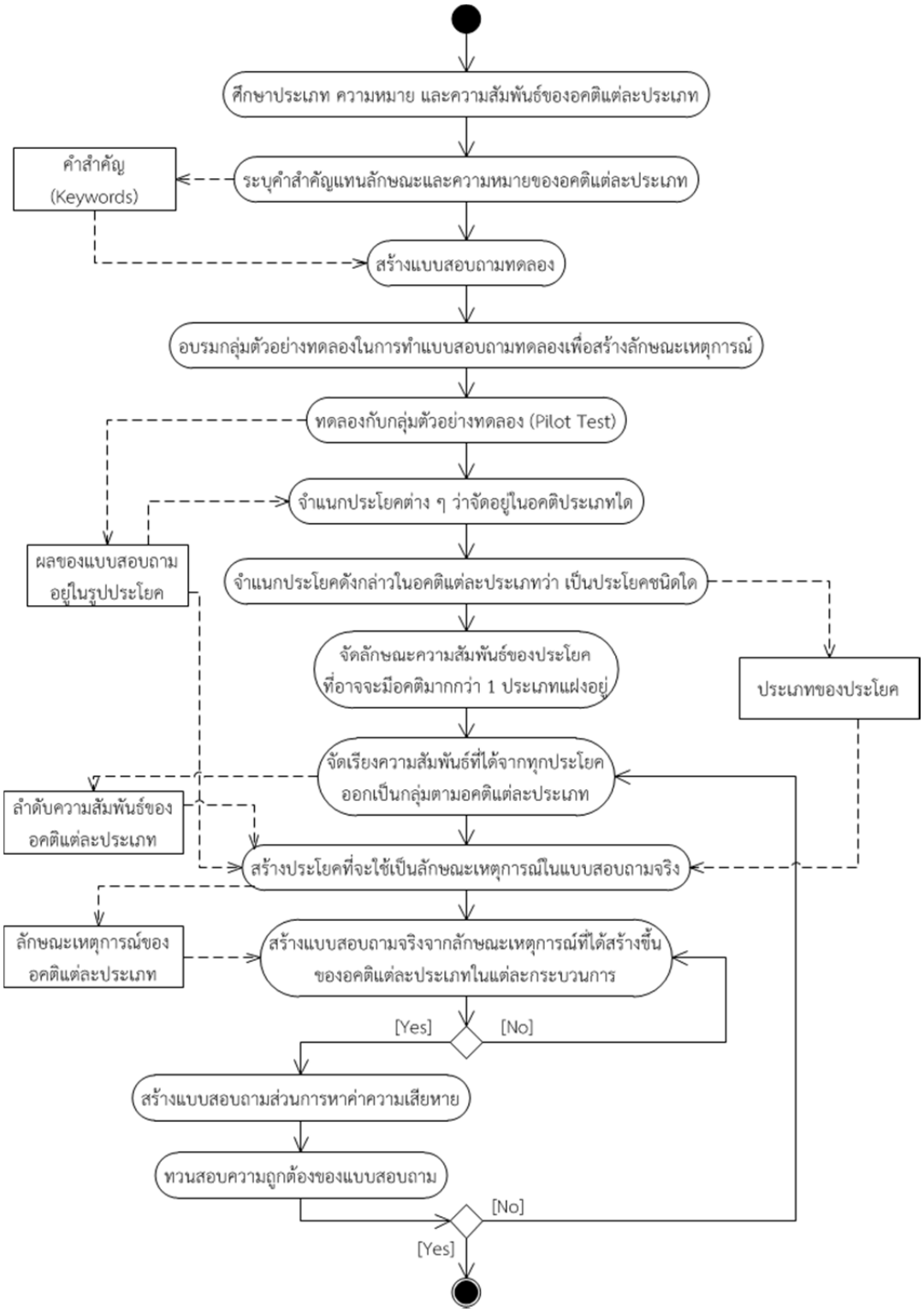
เป็นส่วนที่สอบถามเกี่ยวกับข้อมูลของผู้ตอบแบบสอบถาม เพื่อที่อาจจะได้นำข้อมูลเหล่านี้ไปใช้ประโยชน์ต่อได้ในภายหลัง ซึ่งข้อมูลที่สำคัญ ๆ ของส่วนนี้ได้แก่ เพศ อายุ อายุงาน ระดับการศึกษา ลักษณะงานและตำแหน่งงาน ประเภทธุรกิจขององค์กรที่ปฏิบัติงาน และประสบการณ์เฉพาะด้านที่เกี่ยวกับหัวข้อที่ทำการวิจัย

ทั้งนี้ในประเด็นของประเภทธุรกิจ อาจจะไม่จำเป็นหากจะมีการนำแบบจำลองนี้ไปประยุกต์ใช้ในองค์กรของตัวเอง แต่อาจจะระบุลงไปถึงกับหน่วยงานย่อยภายในองค์กรที่ปฏิบัติงานอยู่ และในส่วนของประสบการณ์เฉพาะด้านที่เกี่ยวข้อกับหัวข้อที่ทำการวิจัยนั้น สำหรับงานวิจัยนี้มุ่งเน้นในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ ดังนั้นควรจะมีคำถามเกี่ยวกับประสบการณ์เฉพาะด้านของทั้งสองกระบวนการนี้ด้วย เพื่อที่อาจจะใช้จำแนกกลุ่มบุคคลที่ตอบแบบสอบถามได้

3.2.3 แบบสอบถามสำหรับการพบอคติในทั้งสองกระบวนการ

แบบสอบถามสำหรับการพบอคติในการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และแบบสอบถามสำหรับการพบอคติในกระบวนการออกแบบซอฟต์แวร์ในส่วนที่ 2 และ 3 ของภาคผนวก ก ในส่วนนี้ยังแบ่งออกเป็น 2 ส่วนย่อย ดังนี้ ส่วนย่อยแรก เป็นแบบสอบถามเพื่อสอบถามถึงข้อมูลที่เกี่ยวกับโครงการที่ปฏิบัติงาน เพื่อที่ต้องการจะทราบลักษณะงาน ระยะเวลา และบุคคลากรในโครงการ เพื่อที่อาจจะสามารถนำผลไปอภิปรายเพิ่มเติมในส่วนอื่น ๆ ได้ ส่วนย่อยที่สอง เพื่อสอบถามเกี่ยวกับอคติในการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ โดยใช้ลักษณะเหตุการณ์ของการมีอคติในการรวบรวมค่าระดับของการมีอคติในหัวข้อที่ 3.1.4 และ

โดยเฉพาะอย่างยิ่งในส่วนที่สองนี้มีรายละเอียดในการออกแบบหลายขั้นตอน จึงนำเสนอด้วยแผนภาพกิจกรรมและอธิบายขั้นตอน ดังนี้



ภาพที่ 3.5 แผนภาพแสดงภาพรวมของการสร้างแบบสอบถาม

1) ศึกษาประเภท ความหมาย และความสัมพันธ์ของอคติแต่ละประเภท
2) ระบุคำสำคัญแทนลักษณะและความหมายของอคติแต่ละประเภท ดังตารางที่ 3.7

ตารางที่ 3.7 ตัวอย่างลักษณะคำสำคัญที่ใช้แทนอคติแต่ละประเภท

ประเภทอคติ	คำสำคัญ
Anchoring	คล้ายกับ, เหมือนกับ, จากประสบการณ์
Adjustment	ปรับแต่ง, เพิ่มเติม, ลดทอน
Availability	จากความนิยม, ที่มีอยู่ตอนนี้, ปัจจุบันนี้
Confirmation	เชื่อว่า, มีคนกล่าวว่า, มั่นใจว่า

3) สร้างแบบสอบถามทดลอง โดยสร้างให้มีส่วนในการให้กลุ่มตัวอย่างทดลองกรอกลักษณะประโยคหรือเหตุการณ์ที่มีอคติจากคำสำคัญที่กำหนดให้ โดยจำแนกออกเป็นอคติแต่ละประเภทในแต่ละกระบวนการที่สนใจ
4) อบรมกลุ่มตัวอย่างทดลอง (Pilot) ถึงประเภท ความหมาย และความสัมพันธ์ของอคติแต่ละประเภท ในการทำแบบสอบถามเพื่อสร้างลักษณะเหตุการณ์ ลักษณะประโยคที่พบในการทำงานจากคำสำคัญ (ตารางที่ 3.7) หรือคำใกล้เคียง
5) ทำการทดลองกลุ่มตัวอย่างทดลองสร้างประโยคที่มีคำสำคัญเหล่านี้ที่พบในการทำงานในแต่ละกระบวนการ
6) ทำการจำแนกประโยคต่าง ๆ ที่ได้มาจากกลุ่มตัวอย่างทดลองว่าจัดอยู่ในอคติประเภทใด โดยดูจากคำสำคัญดังกล่าว
7) จำแนกประโยคดังกล่าวในอคติแต่ละประเภทว่า เป็นประโยคชนิดใด เช่น ประโยคบอกเล่า ประโยคคำถาม ประโยคปฏิเสธ ประโยคขอร้อง ประโยคคำสั่ง เป็นต้น
8) จัดลักษณะความสัมพันธ์ของประโยคที่อาจจะมีอคติมากกว่า 1 ประเภทแฝงอยู่ ว่ามีความสัมพันธ์ไปในลักษณะใด ตามภาพที่ 2.2
9) ทำการจัดเรียงความสัมพันธ์ที่ได้จากทุกประโยคเป็นกลุ่มตามอคติทั้งสามประเภท โดยแบ่งตามจุดกำเนิดของประโยคว่ามาจากอคติประเภทใด ดังตารางที่ 3.8 และ 3.9 ดังนี้

ตารางที่ 3.8 ลำดับความสัมพันธ์ของอคติในกระบวนการรวบรวมและกำหนดความต้องการ

กระบวนการ Requirements	ความสัมพันธ์ของอคติ
Anchoring and adjustment	An
	An → Ad
Availability	Av → An
	Av → An → Ad
	Av → Co
	Av → Co → Ad
	Av → Co → An
Confirmation	Co
	Co → Ad
	Co → An
	Co → An → Ad

An = Anchoring, Ad = Adjustment, Av = Availability, Co = Confirmation

ตารางที่ 3.9 ลำดับความสัมพันธ์ของอคติในกระบวนการออกแบบ

กระบวนการ Design	ความสัมพันธ์ของอคติ
Anchoring and adjustment	Ad
	An
	An → Ad
Availability	Av → An
	Av → An → Ad
	Av → Co
	Av → Co → An
	Av → Co → An → Ad
Confirmation	Co
	Co → Ad
	Co → An
	Co → An → Ad

An = Anchoring, Ad = Adjustment, Av = Availability, Co = Confirmation

10) ทำการสร้างประโยคที่จะใช้เป็นลักษณะเหตุการณ์ในแบบสอบถามจริง โดยอ้างอิงจากคำสำคัญ ดังตัวอย่างในตารางที่ 3.7 รูปแบบประโยคที่ได้จากกลุ่มตัวอย่าง และความสัมพันธ์จากตารางที่ 3.8 และ 3.9 ข้างต้น ซึ่งแสดงไว้ดังตารางที่ 3.10 และ 3.11 ดังนี้

ตารางที่ 3.10 ลักษณะเหตุการณ์ของอคติในกระบวนการรวบรวมและกำหนดความต้องการ

ประเภทอคติ	ลักษณะเหตุการณ์
Anchoring and adjustment	1.1 จากประสบการณ์ของผม/ดิฉัน คิดว่าระบบซอฟต์แวร์ควรที่จะเป็นไปในประมาณลักษณะนี้ (ที่เป็นไปตามที่ผู้พูดคิด)
	1.2 จากประสบการณ์ คิดว่าระบบซอฟต์แวร์ควรที่จะเปลี่ยนแปลงหรือปรับปรุงมาเป็นประมาณนี้ (ที่เป็นไปตามที่ผู้พูดคิด)
Availability	2.1 เนื่องจากความนิยมในเรื่อง ... (เช่น Could Computing, Mobile Application เป็นต้น) ที่มีอยู่ในปัจจุบัน ระบบซอฟต์แวร์ควรจะให้มีความสามารถเหมือนกับความ นิยมนั้น
	2.2 ควรจะเปลี่ยนแปลงหรือปรับปรุงฟังก์ชันการทำงานให้รองรับกับกระแสนิยมใน ปัจจุบัน เทคโนโลยีล่าสุด หรือเทคโนโลยีในอนาคต
	2.3 สำหรับระบบซอฟต์แวร์นี้ ด้วยเทคโนโลยีแล้วสามารถ (หรือไม่สามารถ) ทำให้เหมือนกับความต้องการตามกระแสนิยมในปัจจุบันได้
	2.4 การเปลี่ยนแปลงหรือปรับปรุงระบบซอฟต์แวร์นี้ จากเทคโนโลยีที่มีอยู่ตอนนี้แล้วสามารถ (หรือไม่สามารถ) ทำให้เหมือนกับ ความต้องการตามกระแสนิยมในปัจจุบันได้
	2.5 คิดว่าด้วยเทคโนโลยีในตอนนี้ ระบบซอฟต์แวร์นี้ สามารถ (หรือไม่สามารถ) ทำให้เหมือนกับความต้องการตามกระแสนิยมในปัจจุบันได้

ตารางที่ 3.10 ลักษณะเหตุการณ์ของอคติในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)

ประเภทอคติ	ลักษณะเหตุการณ์
Confirmation	3.1 ผม/ดิฉัน มั่นใจว่าระบบซอฟต์แวร์ต้องแสดงผลหรือให้ผลลัพธ์เป็นอย่างนี้
	3.2 ทำไมระบบซอฟต์แวร์นี้ จึงไม่เป็นไปในลักษณะนั้น (ลักษณะอื่น ๆ ที่ผู้กล่าวคิดว่าดีกว่าลักษณะเดิม) ซึ่งถูกต้องและเหมาะสมมากกว่า
	3.3 ต้องเปลี่ยนแปลงหรือปรับปรุง ระบบซอฟต์แวร์ในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน) นั่นจะทำให้ระบบซอฟต์แวร์เดิมนั้นดีขึ้นหรือมีประสิทธิภาพมากขึ้น
	3.4 จากประสบการณ์คิดว่าระบบซอฟต์แวร์ควรจะเป็นอย่างนี้หรือเป็นแบบนี้ (ตามที่ผู้พูดคิด)
	3.5 คิดว่าควรจะเปลี่ยนแปลงหรือปรับปรุง ระบบซอฟต์แวร์ ในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน)

ตารางที่ 3.11 ลักษณะเหตุการณ์ของอคติแต่ละประเภทในกระบวนการออกแบบ

ประเภทอคติ	ลักษณะเหตุการณ์
Anchoring and adjustment	1.1 ให้ทำการเปลี่ยนแปลงหรือปรับปรุง ระบบซอฟต์แวร์ในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน) สักหน่อย เพื่อให้มีความสมบูรณ์มากขึ้น
	1.2 ผม/ดิฉัน ต้องการให้ระบบซอฟต์แวร์ถูกออกแบบมาเป็นไปในลักษณะประมาณนี้ (เป็นไปตามที่ผู้พูดคิด)
	1.3 คิดว่าการออกแบบ ระบบซอฟต์แวร์ ที่จะให้มีความสามารถประมาณนี้นั้นเพียงพอแล้วหรือยัง
	1.4 จากประสบการณ์ คิดว่าระบบซอฟต์แวร์ ควรที่จะเปลี่ยนแปลงหรือปรับปรุงการออกแบบไปเป็นลักษณะประมาณนี้
	1.5 ควรจะเปลี่ยนแปลงและปรับปรุงในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน) สักหน่อย เพื่อให้มีความสมบูรณ์มากขึ้น

ตารางที่ 3.11 ลักษณะเหตุการณ์ของอคติในกระบวนการออกแบบ (ต่อ)

ประเภทอคติ	ลักษณะเหตุการณ์
Availability	2.1 ด้วยเทคโนโลยีและกระแสนิยมในปัจจุบัน คิดว่าระบบซอฟต์แวร์นี้ควรจะออกแบบให้มีความสามารถเหมือนกับที่กำลังนิยมอยู่ในตอนนี้
	2.2 ควรจะดูแบบอย่าง แล้วนำเอาลักษณะของเทคโนโลยีและกระแสนิยมในปัจจุบัน มาใช้ในการออกแบบระบบซอฟต์แวร์ น่าจะช่วยให้มีความสมบูรณ์มากขึ้น
	2.3 ควรจะออกแบบระบบซอฟต์แวร์ ให้รองรับการเปลี่ยนแปลงและปรับปรุงในอนาคต
	2.4 มั่นใจว่าจากความนิยมและเทคโนโลยีในตอนนี้ การออกแบบให้ระบบซอฟต์แวร์เป็นไปตามความนิยมและตามเทคโนโลยีในตอนนี้ จะทำให้ระบบซอฟต์แวร์นั้นมีประสิทธิภาพ
	2.5 ระบบซอฟต์แวร์นี้ คิดว่าสามารถ (หรือไม่สามารถ) ออกแบบให้เหมือนกับอีกระบบซอฟต์แวร์ที่เป็นแบบที่นิยมใช้กันในปัจจุบันได้
	2.6 คิดว่าการออกแบบระบบซอฟต์แวร์ให้รองรับกับเทคโนโลยีที่นิยมใช้กันมากอยู่ในปัจจุบัน จะได้ใช้ประโยชน์ในอนาคต ถึงแม้ว่าจะยังไม่ได้รับร้องขอความต้องการดังกล่าวมาก็ตาม
Confirmation	3.1 มั่นใจว่าการออกแบบแบบนี้ถูกต้อง เพราะได้รับการยืนยันจากผู้ใช้งานแล้วว่า ให้ออกแบบระบบซอฟต์แวร์ ว่าต้องให้เป็นไปตามแบบนี้
	3.2 ต้องเปลี่ยนแปลงและปรับปรุง ระบบซอฟต์แวร์ในส่วนตรงนี้
	3.3 จากประสบการณ์ คิดว่าระบบซอฟต์แวร์ ที่ต้องออกแบบไปเป็นแบบนี้ จะสามารถทำได้ (หรือทำไม่ได้)
	3.4 ควรจะเปลี่ยนแปลงและปรับปรุง ระบบซอฟต์แวร์ จากตรงนี้ไปเป็นแบบนี้ ซึ่งคิดว่าทำได้ (หรือทำไม่ได้)

11) สร้างแบบสอบถามจริงจากลักษณะเหตุการณ์ที่ได้สร้างขึ้นของอคติแต่ละประเภทในแต่ละกระบวนการ โดยยกตัวอย่างจากอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ ดังนี้

ตารางที่ 3.12 ตัวอย่างลักษณะแบบสอบถามจากกระบวนการรวบรวมและกำหนดความต้องการ

ประเภทของอคติที่เกิดขึ้น/คำอธิบาย/ตัวอย่างเหตุการณ์	จำนวนครั้งที่เกิดเหตุการณ์เหล่านี้ขึ้นในการรวบรวมและกาหนดความต้องการ*
1. Anchoring & adjustment การที่มนุษย์พยายามที่จะสร้างการประมาณการเกี่ยวกับปัญหาขึ้นมา โดยมีแนวโน้มที่ ยึดติดอยู่กับความรู้เดิมที่เคยรู้มาก่อน เพื่อหาวิธีสำหรับการแก้ปัญหาต่าง ๆ โดยปัญหาในการประมาณการเบื้องต้นที่อาจจะมีข้อผิดพลาดจากการประมาณจากประสบการณ์ ของตน ทำให้ในการประมาณการเบื้องต้นทุกครั้งจะต้องทำการปรับแต่งให้เหมาะสมกับสถานการณ์ปัจจุบัน หรือความเป็นจริง	* กำหนดให้การปฏิบัติงานในการรวบรวมและกำหนดความต้องการเป็นค่าระดับประจำของจำนวนร้อยละของการทำงานทั้งหมดในโครงการที่ท่านมีส่วนร่วมในกระบวนการด้านการรวบรวมและกาหนดความต้องการมากที่สุด
(ตัวอย่าง) จากประสบการณ์ของผม/ดิฉัน คิดว่าระบบซอฟต์แวร์ควรที่จะเป็นไปในประมาณลักษณะนี้ (ที่เป็นไปตามที่ผู้พูดคิด)	5
1.1 จากประสบการณ์ของผม/ดิฉัน คิดว่าระบบซอฟต์แวร์ควรที่จะเป็นไปในประมาณลักษณะนี้ (ที่เป็นไปตามที่ผู้พูดคิด)	เลือกตอบค่าระดับ 1 ถึง 5
1.2 จากประสบการณ์ คิดว่าระบบซอฟต์แวร์ควรที่จะเปลี่ยนแปลงหรือปรับปรุงมาเป็นประมาณนี้ (ที่เป็นไปตามที่ผู้พูดคิด)	เลือกตอบค่าระดับ 1 ถึง 5
1.3 อื่น ๆ (ถ้ามี โปรดระบุ...)	เลือกตอบค่าระดับ 1 ถึง 5

12) ทำซ้ำด้วยวิธีการนี้กับอคติประเภทอื่น ๆ ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ ซึ่งข้อมูลที่ผู้ตอบแบบสอบถามมาในอคติแต่ละประเภทนั้น จะนำไปคำนวณหาค่าเฉลี่ย และทำตามวิธี Dummy coding ต่อไป

13) สร้างแบบสอบถามส่วนการหาค่าความเสียหาย ในตอนท้ายของแบบสอบถามของทั้งสองกระบวนการ เป็นส่วนในการหาค่าความเสียหาย โดยกำหนดให้ ในงานด้านการรวบรวม

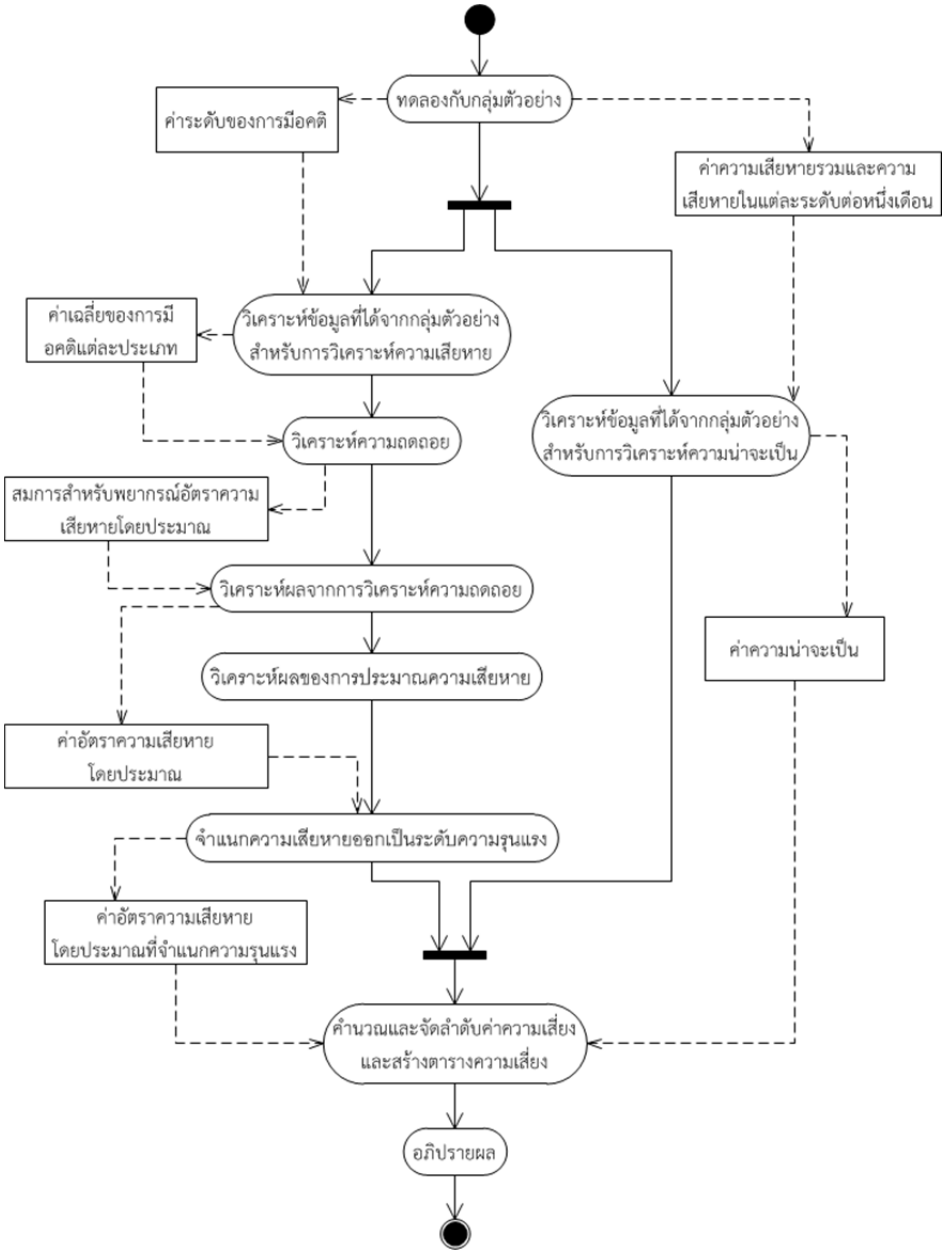
และกำหนดความต้องการซอฟต์แวร์ที่ท่านรับผิดชอบ มีเหตุการณ์ดังกล่าวข้างต้นเกิดขึ้นและเมื่อทำการส่งมอบซอฟต์แวร์ไปแล้ว เป็นระยะเวลา 1 เดือน มีจำนวนของข้อผิดพลาดที่เกิดขึ้นกับซอฟต์แวร์ที่มีการรายงานกลับมากี่ครั้ง โดยให้ผู้ตอบแบบสอบถามระบุจำนวนดังกล่าว และจำแนกจำนวนดังกล่าวออกเป็นระดับความรุนแรงใน 5 ระดับ ดังที่ได้กล่าวไปในหัวข้อ 3.1.6 ซึ่งค่าที่ได้จากการตอบแบบสอบถามส่วนนี้จะนำไปคำนวณหาค่าความเสียหายประมาณในการวิเคราะห์ความถดถอยต่อไป

14) ทวนสอบความถูกต้องแบบสอบถาม โดยเฉพาะอย่างยิ่งในส่วนของการจัดเรียงกระบวนการ การจัดเรียงประเภทของอคติและส่วนของการกรอกค่าความเสียหาย ซึ่งเป็นส่วนที่มีความละเอียดอ่อนในการใช้ภาษาค่อนข้างมาก

บทที่ 4

การทดลองและการวิเคราะห์ผล

ในบทนี้จะอธิบายการทดลอง ผลการทดลองและการวิเคราะห์ผล ที่ใช้วิธีที่ได้นำเสนอไปในบทที่ 3 เพื่อให้ได้มาซึ่งแบบจำลองการประเมินความเสี่ยงและสามารถนำแบบจำลองการประเมินความเสี่ยงนี้ไปสร้างเป็นเครื่องมือในการประเมินความเสี่ยงได้ โดยมีภาพรวมดังแผนภาพ 4.1 ดังนี้



ภาพที่ 4.1 แผนภาพกิจกรรมอธิบายภาพรวมของการทดลองและวิเคราะห์ผล

4.1 ทดลองกับกลุ่มตัวอย่าง

เพื่อนำข้อมูลมาใช้สร้างแบบจำลอง จึงจะต้องเก็บข้อมูลด้วยแบบสอบถามจากกลุ่มตัวอย่าง ซึ่งได้ทำการเลือกกลุ่มตัวอย่าง โดยทำการรวบรวมกลุ่มตัวอย่างที่เป็นผู้ปฏิบัติงานในโครงการซอฟต์แวร์จากกลุ่มผู้ที่ทำงานเกี่ยวกับซอฟต์แวร์ ในอุตสาหกรรมต่าง ๆ กัน ซึ่งจะหาได้จากสถานที่หรือโอกาสที่มีกลุ่มผู้ที่ทำงานเกี่ยวกับซอฟต์แวร์มารวมตัวกันจากหลากหลายอุตสาหกรรม โดยจะสนใจในกลุ่มนิสิตนักศึกษาระดับบัณฑิตศึกษา ที่กำลังศึกษาต่อในสาขาวิชาที่เกี่ยวกับวิศวกรรมคอมพิวเตอร์ วิทยาศาสตร์คอมพิวเตอร์ วิศวกรรมซอฟต์แวร์คอมพิวเตอร์ธุรกิจ และเทคโนโลยีสารสนเทศ และทำงานในอุตสาหกรรมซอฟต์แวร์ควบคู่กันไปด้วย และในแต่ละคนจะต้องมีประสบการณ์ทำงานในตำแหน่งที่เกี่ยวกับ ผู้จัดการโครงการ นักวิเคราะห์ธุรกิจ นักวิเคราะห์ระบบ นักออกแบบระบบ และนักพัฒนาระบบ มาแล้วอย่างน้อยไม่ต่ำกว่า 1 ปี

โดยได้กระจายแบบสอบถามไปในเรื่องกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์และกระบวนการออกแบบซอฟต์แวร์ ทั้งสองกระบวนการได้กลุ่มตัวอย่างทั้งสิ้น กระบวนการละ 51 คน และเมื่อนำมาทำการวิเคราะห์ด้วยโปรแกรมวิเคราะห์ทางสถิติแล้วพบว่ามีกลุ่มตัวอย่างที่สามารถนำมาวิเคราะห์ต่อไปได้ กระบวนการละ 31 คน และโดยปกติ กลุ่มตัวอย่างที่ทดลองจะต้องมีการแจกแจงเป็นปกติ หรือยึดทฤษฎีแนวโน้มเข้าสู่ส่วนกลาง (Central Limit Theorem) ขนาดกลุ่มตัวอย่างต้องมีขนาด 30 คนขึ้นไป [30] ซึ่งจะใช้กลุ่มตัวอย่างดังกล่าวในการคำนวณต่อไป

4.2 วิเคราะห์ข้อมูลที่ได้จากกลุ่มตัวอย่างสำหรับการวิเคราะห์ความเสียหาย

สำหรับกลุ่มตัวอย่าง 31 คนนั้น ในแต่ละกระบวนการมีกลุ่มตัวอย่างที่แตกต่างกันบ้าง ไม่ใช่เป็นกลุ่มตัวอย่างเดียวกันทั้งหมด 31 คน โดยในหัวข้อนี้จะแยกออกเป็นการวิเคราะห์ข้อมูลในสองกระบวนการนั่นคือกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์และกระบวนการออกแบบซอฟต์แวร์

โดยในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ ได้ค่าเฉลี่ยของการมีอคติในแต่ละประเภทจากแบบสอบถามในส่วนที่ 2 ของภาคผนวก ก และค่าที่ปัดเศษแล้วของค่าเฉลี่ยนั้น ดังตารางที่ 4.1

ตารางที่ 4.1 ค่าเฉลี่ยและค่าที่ปัดเศษแล้วของการมีอคติแต่ละประเภทในกระบวนการรวบรวมและกำหนดความต้องการ

No.	Average of Anchoring	Round of Anchoring	Average of Availability	Round of Availability	Average of Confirmation	Round of Confirmation
1	4.00	4	3.40	3	2.33	2
2	4.00	4	2.80	3	2.33	2
3	3.00	3	2.80	3	2.67	3
4	3.00	3	5.00	5	3.17	3
5	4.00	4	3.40	3	4.17	4
6	5.00	5	4.20	4	3.33	3
7	3.50	4	3.60	4	3.83	4
8	2.50	3	1.60	2	2.33	2
9	1.00	1	1.00	1	1.00	1
10	3.00	3	3.00	3	3.00	3
11	3.00	3	3.40	3	2.83	3
12	3.00	3	3.40	3	2.83	3
13	3.00	3	3.00	3	3.00	3
14	4.00	4	4.40	4	4.50	5
15	4.00	4	3.40	3	4.00	4
16	1.50	2	3.40	3	1.50	2
17	4.00	4	3.00	3	3.17	3
18	4.50	5	2.40	2	3.33	3
19	5.00	5	1.00	1	4.00	4
20	2.50	3	3.20	3	3.17	3
21	2.00	2	3.20	3	2.67	3
22	3.00	3	3.80	4	2.83	3
23	1.00	1	1.60	2	3.83	4
24	3.00	3	4.00	4	3.83	4

ตารางที่ 4.1 ค่าเฉลี่ยและค่าที่ปัดเศษแล้วของการมีอคติแต่ละประเภทในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)

No.	Average of Anchoring	Round of Anchoring	Average of Availability	Round of Availability	Average of Confirmation	Round of Confirmation
25	4.00	4	4.40	4	4.00	4
26	3.00	3	3.60	4	3.17	3
27	3.00	3	3.00	3	3.00	3
28	3.00	3	1.80	2	2.67	3
29	4.50	5	3.80	4	4.00	4
30	4.50	5	3.60	4	3.83	4
31	3.00	3	3.00	3	3.00	3

และเมื่อนำไปแปลงค่าให้เป็นตัวแปรหุ่นด้วยวิธี Dummy coding จะได้ดังตารางที่ 4.2

ตารางที่ 4.2 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการ

No.	Round of Anchoring	ancx2	ancx3	ancx4	ancx5
1	4	0	0	1	0
2	4	0	0	1	0
3	3	0	1	0	0
4	3	0	1	0	0
5	4	0	0	1	0
6	5	0	0	0	1
7	4	0	0	1	0
8	3	0	1	0	0
9	1	0	0	0	0
10	3	0	1	0	0

ตารางที่ 4.2 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)

No.	Round of Anchoring	ancx2	ancx3	ancx4	ancx5
11	3	0	1	0	0
12	3	0	1	0	0
13	3	0	1	0	0
14	4	0	0	1	0
15	4	0	0	1	0
16	2	1	0	0	0
17	4	0	0	1	0
18	5	0	0	0	1
19	5	0	0	0	1
20	3	0	1	0	0
21	2	1	0	0	0
22	3	0	1	0	0
23	1	0	0	0	0
24	3	0	1	0	0
25	4	0	0	1	0
26	3	0	1	0	0
27	3	0	1	0	0
28	3	0	1	0	0
29	5	0	0	0	1
30	5	0	0	0	1
31	3	0	1	0	0

ตารางที่ 4.3 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Availability ในกระบวนการรวบรวมและกำหนดความต้องการ

No.	Round of Availability	avax2	avax3	avax4	avax5
1	3	0	1	0	0
2	3	0	1	0	0
3	3	0	1	0	0
4	5	0	0	0	1
5	3	0	1	0	0
6	4	0	0	1	0
7	4	0	0	1	0
8	2	1	0	0	0
9	1	0	0	0	0
10	3	0	1	0	0
11	3	0	1	0	0
12	3	0	1	0	0
13	3	0	1	0	0
14	4	0	0	1	0
15	3	0	1	0	0
16	3	0	1	0	0
17	3	0	1	0	0
18	2	1	0	0	0
19	1	0	0	0	0
20	3	0	1	0	0
21	3	0	1	0	0
22	4	0	0	1	0
23	2	1	0	0	0
24	4	0	0	1	0
25	4	0	0	1	0

ตารางที่ 4.3 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Availability ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)

No.	Round of Availability	avax2	avax3	avax4	avax5
26	4	0	0	1	0
27	3	0	1	0	0
28	2	1	0	0	0
29	4	0	0	1	0
30	4	0	0	1	0
31	3	0	1	0	0

ตารางที่ 4.4 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Confirmation ในกระบวนการรวบรวมและกำหนดความต้องการ

No.	Round of Confirmation	conx2	conx3	conx4	conx5
1	2	1	0	0	0
2	2	1	0	0	0
3	3	0	1	0	0
4	3	0	1	0	0
5	4	0	0	1	0
6	3	0	1	0	0
7	4	0	0	1	0
8	2	1	0	0	0
9	1	0	0	0	0
10	3	0	1	0	0
11	3	0	1	0	0
12	3	0	1	0	0
13	3	0	1	0	0

ตารางที่ 4.4 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Confirmation ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)

No.	Round of Confirmation	conx2	conx3	conx4	conx5
14	5	0	0	0	1
15	4	0	0	1	0
16	2	1	0	0	0
17	3	0	1	0	0
18	3	0	1	0	0
19	4	0	0	1	0
20	3	0	1	0	0
21	3	0	1	0	0
22	3	0	1	0	0
23	4	0	0	1	0
24	4	0	0	1	0
25	4	0	0	1	0
26	3	0	1	0	0
27	3	0	1	0	0
28	3	0	1	0	0
29	4	0	0	1	0
30	4	0	0	1	0
31	3	0	1	0	0

และจากแบบสอบถามที่ระบุถึงความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน (Y) นั้น สามารถจำแนกออกเป็นแต่ละระดับความรุนแรง 5 ระดับ นั่นคือ Critical, Major, Average, Minor, Exception ได้ ดังตารางที่ 4.5

ตารางที่ 4.5 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน โดยจำแนกออกเป็นแต่ละระดับความรุนแรง ในกระบวนการรวบรวมและกำหนดความต้องการ

No.	Y	Critical	Major	Average	Minor	Exception
1	2	0	0	1	1	0
2	3	0	0	1	1	1
3	5	0	0	1	2	2
4	3	0	2	1	0	0
5	3	0	0	2	1	0
6	4	0	0	1	0	3
7	5	2	0	0	2	1
8	4	0	0	1	0	3
9	4	0	0	1	0	3
10	2	0	0	1	1	0
11	1	0	0	0	0	1
12	4	0	0	1	0	3
13	2	0	1	1	0	0
14	4	0	0	1	0	3
15	3	0	0	1	1	1
16	5	0	2	0	3	0
17	2	0	0	1	1	0
18	2	0	0	1	1	0
19	3	0	0	0	2	1
20	2	0	0	1	1	0
21	3	0	1	0	1	1
22	2	0	0	1	1	0
23	5	0	0	0	2	3
24	3	0	0	0	3	0
25	1	0	0	1	0	0
26	1	0	0	0	1	0

ตารางที่ 4.5 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน โดยจำแนกออกเป็นแต่ละระดับความรุนแรง ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)

No.	Y	Critical	Major	Average	Minor	Exception
27	5	0	1	2	2	0
28	3	1	0	1	1	0
29	3	0	0	3	0	0
30	4	0	0	2	1	1
31	1	0	0	0	0	1
Average	3.03	0.10	0.23	0.87	0.94	0.90
Summary	94	3	7	27	29	28
100%	100	3.19	7.45	28.72	30.85	29.79

จากข้อมูลด้านบน สามารถเตรียมข้อมูลที่จะนำไปวิเคราะห์ความถดถอย โดยที่จะศึกษาความสัมพันธ์ระหว่างการมีอคติทั้งสามประเภทในแต่ละระดับ กับค่าความเสียหาย (Y) ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ (ตารางที่ 4.2 รวมกับตารางที่ 4.5) ได้ดังนี้

ตารางที่ 4.6 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการ

No.	ancx2	ancx3	ancx4	ancx5	Y
1	0	0	1	0	2
2	0	0	1	0	3
3	0	1	0	0	5
4	0	1	0	0	3
5	0	0	1	0	3
6	0	0	0	1	4
7	0	0	1	0	5
8	0	1	0	0	4
	0	0	0	0	4

ตารางที่ 4.6 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Anchoring and adjustment ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)

No.	ancx2	ancx3	ancx4	ancx5	Y
10	0	1	0	0	2
11	0	1	0	0	1
12	0	1	0	0	4
13	0	1	0	0	2
14	0	0	1	0	4
15	0	0	1	0	3
16	1	0	0	0	5
17	0	0	1	0	2
18	0	0	0	1	2
19	0	0	0	1	3
20	0	1	0	0	2
21	1	0	0	0	3
22	0	1	0	0	2
23	0	0	0	0	5
24	0	1	0	0	3
25	0	0	1	0	1
26	0	1	0	0	1
27	0	1	0	0	5
28	0	1	0	0	3
29	0	0	0	1	3
30	0	0	0	1	4
31	0	1	0	0	1

ตารางที่ 4.7 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Availability ในกระบวนการรวบรวมและกำหนดความต้องการ

No.	avax2	avax3	avax4	avax5	Y
1	0	1	0	0	2
2	0	1	0	0	3
3	0	1	0	0	5
4	0	0	0	1	3
5	0	1	0	0	3
6	0	0	1	0	4
7	0	0	1	0	5
8	1	0	0	0	4
9	0	0	0	0	4
10	0	1	0	0	2
11	0	1	0	0	1
12	0	1	0	0	4
13	0	1	0	0	2
14	0	0	1	0	4
15	0	1	0	0	3
16	0	1	0	0	5
17	0	1	0	0	2
18	1	0	0	0	2
19	0	0	0	0	3
20	0	1	0	0	2
21	0	1	0	0	3
22	0	0	1	0	2
23	1	0	0	0	5
24	0	0	1	0	3
25	0	0	1	0	1
26	0	0	1	0	1

ตารางที่ 4.7 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Availability ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)

No.	avax2	avax3	avax4	avax5	Y
27	0	1	0	0	5
28	1	0	0	0	3
29	0	0	1	0	3
30	0	0	1	0	4
31	0	1	0	0	1

ตารางที่ 4.8 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Confirmation ในกระบวนการรวบรวมและกำหนดความต้องการ

No.	conx2	conx3	conx4	conx5	Y
1	1	0	0	0	2
2	1	0	0	0	3
3	0	1	0	0	5
4	0	1	0	0	3
5	0	0	1	0	3
6	0	1	0	0	4
7	0	0	1	0	5
8	1	0	0	0	4
9	0	0	0	0	4
10	0	1	0	0	2
11	0	1	0	0	1
12	0	1	0	0	4
13	0	1	0	0	2
14	0	0	0	1	4
15	0	0	1	0	3
16	1	0	0	0	5

ตารางที่ 4.8 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Confirmation ในกระบวนการรวบรวมและกำหนดความต้องการ (ต่อ)

No.	conx2	conx3	conx4	conx5	Y
17	0	1	0	0	2
18	0	1	0	0	2
19	0	0	1	0	3
20	0	1	0	0	2
21	0	1	0	0	3
22	0	1	0	0	2
23	0	0	1	0	5
24	0	0	1	0	3
25	0	0	1	0	1
26	0	1	0	0	1
27	0	1	0	0	5
28	0	1	0	0	3
29	0	0	1	0	3
30	0	0	1	0	4
31	0	1	0	0	1

และในกระบวนการออกแบบซอฟต์แวร์ ได้ค่าเฉลี่ยของการมีอคติในแต่ละประเภทจากแบบสอบถามในส่วนที่ 3 ของภาคผนวก ก และค่าที่ปัดเศษแล้วของค่าเฉลี่ยนั้น ดังตารางที่ 4.9

ตารางที่ 4.9 ค่าเฉลี่ยและค่าที่ปัดเศษแล้วของการมีอคติแต่ละประเภทในกระบวนการออกแบบ

No.	Average of Anchoring	Round of Anchoring	Average of Availability	Round of Availability	Average of Confirmation	Round of Confirmation
1	3.80	4	3.33	3	4.25	4
2	2.40	2	2.83	3	2.00	2
3	3.00	3	2.67	3	3.00	3
4	3.00	3	4.50	5	4.50	5
5	3.20	3	4.50	5	4.25	4
6	3.00	3	2.17	2	2.00	2
7	2.20	2	3.50	4	2.25	2
8	2.80	3	3.83	4	2.75	3
9	3.20	3	3.33	3	3.50	4
10	2.40	2	2.33	2	2.25	2
11	3.60	4	3.17	3	4.25	4
12	4.20	4	3.50	4	5.00	5
13	2.80	3	4.67	5	4.00	4
14	3.20	3	4.33	4	3.25	3
15	2.40	2	2.50	3	2.50	3
16	3.00	3	3.50	4	4.00	4
17	1.80	2	3.67	4	4.00	4
18	2.20	2	1.00	1	4.25	4
19	3.80	4	3.67	4	3.25	3
20	1.60	2	2.67	3	3.25	3
21	2.00	2	1.00	1	2.25	2
22	2.20	2	3.00	3	3.75	4
23	1.60	2	1.33	1	1.25	1
24	3.00	3	1.83	2	3.00	3

ตารางที่ 4.9 ค่าเฉลี่ยและค่าที่ปัดเศษแล้วของการมีอคติแต่ละประเภทในกระบวนการออกแบบ (ต่อ)

No.	Average of Anchoring	Round of Anchoring	Average of Availability	Round of Availability	Average of Confirmation	Round of Confirmation
25	4.00	4	4.00	4	4.00	4
26	3.00	3	3.00	3	3.25	3
27	2.60	3	3.17	3	2.50	3
28	2.00	2	2.33	2	1.75	2
29	3.40	3	3.50	4	4.00	4
30	4.60	5	2.67	3	4.00	4
31	4.80	5	4.00	4	4.25	4

และเมื่อนำไปแปลงค่าให้เป็นตัวแปรหุ่นด้วยวิธี Dummy coding จะได้ดังตารางที่ 4.10

ตารางที่ 4.10 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Anchoring and adjustment ในกระบวนการออกแบบ

No.	Round of Anchoring	ancx2	ancx3	ancx4	ancx5
1	4	0	0	1	0
2	2	1	0	0	0
3	3	0	1	0	0
4	3	0	1	0	0
5	3	0	1	0	0
6	3	0	1	0	0
7	2	1	0	0	0
8	3	0	1	0	0
9	3	0	1	0	0
10	2	1	0	0	0

ตารางที่ 4.10 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Anchoring and adjustment ในกระบวนการออกแบบ (ต่อ)

No.	Round of Anchoring	ancx2	ancx3	ancx4	ancx5
11	4	0	0	1	0
12	4	0	0	1	0
13	3	0	1	0	0
14	3	0	1	0	0
15	2	1	0	0	0
16	3	0	1	0	0
17	2	1	0	0	0
18	2	1	0	0	0
19	4	0	0	1	0
20	2	1	0	0	0
21	2	1	0	0	0
22	2	1	0	0	0
23	2	1	0	0	0
24	3	0	1	0	0
25	4	0	0	1	0
26	3	0	1	0	0
27	3	0	1	0	0
28	2	1	0	0	0
29	3	0	1	0	0
30	5	0	0	0	1
31	5	0	0	0	1

ตารางที่ 4.11 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Availability ในกระบวนการออกแบบ

No.	Round of Availability	avax2	avax3	avax4	avax5
1	3	0	1	0	0
2	3	0	1	0	0
3	3	0	1	0	0
4	5	0	0	0	1
5	5	0	0	0	1
6	2	1	0	0	0
7	4	0	0	1	0
8	4	0	0	1	0
9	3	0	1	0	0
10	2	1	0	0	0
11	3	0	1	0	0
12	4	0	0	1	0
13	5	0	0	0	1
14	4	0	0	1	0
15	3	0	1	0	0
16	4	0	0	1	0
17	4	0	0	1	0
18	1	0	0	0	0
19	4	0	0	1	0
20	3	0	1	0	0
21	1	0	0	0	0
22	3	0	1	0	0
23	1	0	0	0	0
24	2	1	0	0	0
25	4	0	0	1	0

ตารางที่ 4.11 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Availability ในกระบวนการออกแบบ (ต่อ)

No.	Round of Availability	avax2	avax3	avax4	avax5
26	3	0	1	0	0
27	3	0	1	0	0
28	2	1	0	0	0
29	4	0	0	1	0
30	3	0	1	0	0
31	4	0	0	1	0

ตารางที่ 4.12 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Confirmation ในกระบวนการออกแบบ

No.	Round of Confirmation	conx2	conx3	conx4	conx5
1	4	0	0	1	0
2	2	1	0	0	0
3	3	0	1	0	0
4	5	0	0	0	1
5	4	0	0	1	0
6	2	1	0	0	0
7	2	1	0	0	0
8	3	0	1	0	0
9	4	0	0	1	0
10	2	1	0	0	0
11	4	0	0	1	0
12	5	0	0	0	1
13	4	0	0	1	0

ตารางที่ 4.12 ผลของการแปลงค่าจากค่าเฉลี่ยที่ทำการปัดเศษแล้วไปเป็นตัวแปรหุ่นของการมีอคติประเภท Confirmation ในกระบวนการออกแบบ (ต่อ)

No.	Round of Confirmation	conx2	conx3	conx4	conx5
14	3	0	1	0	0
15	3	0	1	0	0
16	4	0	0	1	0
17	4	0	0	1	0
18	4	0	0	1	0
19	3	0	1	0	0
20	3	0	1	0	0
21	2	1	0	0	0
22	4	0	0	1	0
23	1	0	0	0	0
24	3	0	1	0	0
25	4	0	0	1	0
26	3	0	1	0	0
27	3	0	1	0	0
28	2	1	0	0	0
29	4	0	0	1	0
30	4	0	0	1	0
31	4	0	0	1	0

และจากแบบสอบถามที่ระบุถึงความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน (Y) นั้น สามารถจำแนกออกเป็นแต่ละระดับความรุนแรง 5 ระดับ นั่นคือ Critical, Major, Average, Minor, Exception ได้ ดังตารางที่ 4.5

ตารางที่ 4.13 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน โดยจำแนกออกเป็นแต่ละระดับความรุนแรง ในกระบวนการออกแบบ

No.	Y	Critical	Major	Average	Minor	Exception
1	3	0	0	3	0	0
2	1	0	0	1	0	0
3	4	0	0	2	2	0
4	3	0	0	1	2	0
5	5	0	1	2	1	1
6	3	0	1	2	0	0
7	3	0	2	0	1	0
8	2	0	0	1	1	0
9	2	0	0	1	1	0
10	1	0	0	0	1	0
11	1	0	0	0	0	1
12	5	0	1	2	1	1
13	5	1	1	2	1	0
14	2	0	1	1	0	0
15	2	0	0	1	1	0
16	3	0	0	1	1	1
17	5	0	2	0	3	0
18	5	1	0	4	0	0
19	2	0	2	0	0	0
20	2	0	0	1	1	0
21	3	0	0	2	1	0
22	2	0	0	1	1	0
23	3	0	1	0	1	1
24	3	0	0	0	3	0
25	1	0	0	1	0	0
26	1	0	0	0	0	1

ตารางที่ 4.13 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน โดยจำแนกออกเป็นแต่ละระดับความรุนแรง ในกระบวนการออกแบบ (ต่อ)

No.	Y	Critical	Major	Average	Minor	Exception
27	2	0	0	0	2	0
28	3	0	0	2	1	0
29	4	0	0	2	2	0
30	4	0	0	2	2	0
31	4	0	0	2	2	0
Average	2.87	0.06	0.39	1.19	1.03	0.19
Summary	89	2	12	37	32	6
100%	100	2.25	13.48	41.57	35.96	6.74

จากข้อมูลด้านบน สามารถเตรียมข้อมูลที่จะนำไปวิเคราะห์ความถดถอย โดยที่จะศึกษาความสัมพันธ์ระหว่างการมีอคติทั้งสามประเภทในแต่ละระดับ กับค่าความเสียหาย (Y) ในกระบวนการออกแบบซอฟต์แวร์ (ตารางที่ 4.10 รวมกับตารางที่ 4.13) ได้ดังนี้

ตารางที่ 4.14 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Anchoring and adjustment ในกระบวนการออกแบบ

No.	ancx2	ancx3	ancx4	ancx5	Y
1	0	0	1	0	3
2	1	0	0	0	1
3	0	1	0	0	4
4	0	1	0	0	3
5	0	1	0	0	5
6	0	1	0	0	3
7	1	0	0	0	3
8	0	1	0	0	2
9	0	1	0	0	2

ตารางที่ 4.14 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Anchoring and adjustment ในกระบวนการออกแบบ (ต่อ)

No.	ancx2	ancx3	ancx4	ancx5	Y
10	1	0	0	0	1
11	0	0	1	0	1
12	0	0	1	0	5
13	0	1	0	0	5
14	0	1	0	0	2
15	1	0	0	0	2
16	0	1	0	0	3
17	1	0	0	0	5
18	1	0	0	0	5
19	0	0	1	0	2
20	1	0	0	0	2
21	1	0	0	0	3
22	1	0	0	0	2
23	1	0	0	0	3
24	0	1	0	0	3
25	0	0	1	0	1
26	0	1	0	0	1
27	0	1	0	0	2
28	1	0	0	0	3
29	0	1	0	0	4
30	0	0	0	1	4
31	0	0	0	1	4

ตารางที่ 4.15 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Availability ในกระบวนการออกแบบ

No.	avax2	avax3	avax4	avax5	Y
1	0	1	0	0	3
2	0	1	0	0	1
3	0	1	0	0	4
4	0	0	0	1	3
5	0	0	0	1	5
6	1	0	0	0	3
7	0	0	1	0	3
8	0	0	1	0	2
9	0	1	0	0	2
10	1	0	0	0	1
11	0	1	0	0	1
12	0	0	1	0	5
13	0	0	0	1	5
14	0	0	1	0	2
15	0	1	0	0	2
16	0	0	1	0	3
17	0	0	1	0	5
18	0	0	0	0	5
19	0	0	1	0	2
20	0	1	0	0	2
21	0	0	0	0	3
22	0	1	0	0	2
23	0	0	0	0	3
24	1	0	0	0	3
25	0	0	1	0	1
26	0	1	0	0	1

ตารางที่ 4.15 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Availability ในกระบวนการออกแบบ (ต่อ)

No.	avax2	avax3	avax4	avax5	Y
27	0	1	0	0	2
28	1	0	0	0	3
29	0	0	1	0	4
30	0	1	0	0	4
31	0	0	1	0	4

ตารางที่ 4.16 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Confirmation ในกระบวนการออกแบบ

No.	conx2	conx3	conx4	conx5	Y
1	0	0	1	0	3
2	1	0	0	0	1
3	0	1	0	0	4
4	0	0	0	1	3
5	0	0	1	0	5
6	1	0	0	0	3
7	1	0	0	0	3
8	0	1	0	0	2
9	0	0	1	0	2
10	1	0	0	0	1
11	0	0	1	0	1
12	0	0	0	1	5
13	0	0	1	0	5
14	0	1	0	0	2
15	0	1	0	0	2
16	0	0	1	0	3

ตารางที่ 4.16 ค่าความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน กับตัวแปรหุ่นของอคติประเภท Confirmation ในกระบวนการออกแบบ (ต่อ)

No.	conx2	conx3	conx4	conx5	Y
17	0	0	1	0	5
18	0	0	1	0	5
19	0	1	0	0	2
20	0	1	0	0	2
21	1	0	0	0	3
22	0	0	1	0	2
23	0	0	0	0	3
24	0	1	0	0	3
25	0	0	1	0	1
26	0	1	0	0	1
27	0	1	0	0	2
28	1	0	0	0	3
29	0	0	1	0	4
30	0	0	1	0	4
31	0	0	1	0	4

4.3 วิเคราะห์ความถดถอย

ทำการวิเคราะห์ความถดถอยด้วยโปรแกรมประยุกต์ทางสถิติโดยใช้การวิเคราะห์ถดถอยพหุคูณสำหรับตัวแปรอิสระหลายตัวแปร โดยกำหนดค่าตัวแปรอิสระในโปรแกรมประยุกต์เป็น การมีอคติแต่ละประเภทในแต่ละระดับ และกำหนดค่าตัวแปรตามในโปรแกรมประยุตก์เป็น ความเสียหายของซอฟต์แวร์ในระยะเวลาหนึ่งเดือน

4.4 วิเคราะห์ผลจากการวิเคราะห์ความถดถอย

ผลของการวิเคราะห์ความถดถอยด้วยโปรแกรมประยุกต์ทางสถิติ จากกลุ่มตัวอย่าง 31 คน นั้นในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ ที่ศึกษาความสัมพันธ์ระหว่างตัวแปร

อิสระ คือ การมีอคติทั้งสามประเภทในแต่ละระดับความรุนแรง กับตัวแปรตาม คือ ค่าความเสียหายของซอฟต์แวร์ (Y) ซึ่งใช้วิธีการคัดเลือกตัวแปรอิสระในการทดสอบด้วยวิธีการ Enter นั่นคือการนำตัวแปรอิสระเข้าสมการทั้งหมด และแทนด้วยสมการที่ 3.5 ได้ผลดังนี้

4.4.1 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์จากอคติประเภท Anchoring and adjustment

ตารางที่ 4.17 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย

Variables Entered/Removed

Model	Variables Entered	Variables Removed	Method
1	ancx5, ancx2, ancx4, ancx3	.	Enter

a. All requested variables entered

b. Dependent Variable: Y

ตารางที่ 4.18 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.403[a]	.162	.033	1.256	1.799

a. Predictors: (Constant), ancx5, ancx2, ancx4, ancx3

b. Dependent Variable: Y

ตารางที่ 4.19 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย

ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	7.936	4	1.984	1.257	.312
	Residual	41.032	26	1.578		
	Total	48.968	30			

a. Predictors: (Constant), ancx5, ancx2, ancx4, ancx3

b. Dependent Variable: Y

ตารางที่ 4.20 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	4.500	.888		5.066	.000
	ancx2	-.500	1.256	-.098	-.398	.694
	ancx3	-1.786	.950	-.707	-1.880	.071
	ancx4	-1.625	.993	-.566	-1.636	.114
	ancx5	-1.300	1.051	-.380	-1.237	.227

a. Dependent Variable: Y

ตารางที่ 4.21 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์

Variables	B	SE_B	Beta	t	p-value
ancx2	-.500	1.256	-.098	-.398	.694
ancx3	-1.786	.950	-.707	-1.880	.071
ancx4	-1.625	.993	-.566	-1.636	.114
ancx5	-1.300	1.051	-.380	-1.237	.227
Constant 4.500; SE_{est} = ±1.256 R^2=.162; F=1.257; p-value<.05					

จากตาราง 4.21 จะเห็นว่าตัวแปรอิสระในแต่ละระดับของอคติประเภท Anchoring and adjustment มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อยู่ในระดับต่ำมาก โดยมี R^2 เป็น .162 นั่นสามารถร่วมกันพยากรณ์ความเสียหายของซอฟต์แวร์ได้ร้อยละ 16.2 โดยมีความคลาดเคลื่อนมาตรฐานในการพยากรณ์เท่ากับ ±1.256 และเมื่อพิจารณาค่าสัมประสิทธิ์การถดถอยของตัวพยากรณ์พบว่าตัวแปรอิสระไม่สามารถพยากรณ์ความความเสียหายของซอฟต์แวร์ได้ (p-value > .05) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05

ตรวจสอบเงื่อนไขจากค่าสถิติทดสอบ Durbin-Watson = 1.799 อยู่ในช่วง 1.5-2.5 สรุปได้ว่าค่าคลาดเคลื่อนเป็นอิสระต่อกัน เป็นไปตามเงื่อนไขของการวิเคราะห์ถดถอย และสำหรับสมการพยากรณ์ความเสียหายของซอฟต์แวร์เมื่อนำตัวแปรอิสระทุกตัวเข้าสมการในรูปคะแนนดิบ จะเป็นดังนี้

$$\hat{Y}_{anc} = 4.5 - 0.5ancx2 - 1.786ancx3 - 1.625ancx4 - 1.3ancx5 \quad (4.1)$$

ภายหลังจากได้สมการสำหรับการพยากรณ์ค่าความเสียหายของอคติประเภท Anchoring and adjustment ของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์แล้ว จะนำไปทดสอบด้วยการคำนวณค่าความเสียหายประมาณเมื่อเกิดการมีอคติประเภทนี้ในระดับต่าง ๆ ต่อไป

4.4.2 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์จากอคติประเภท Availability

ตารางที่ 4.22 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย

Variables Entered/Removed

Model	Variables Entered	Variables Removed	Method
1	avax5, avax2, avax4, avax3	.	Enter

a. All requested variables entered

b. Dependent Variable: Y

ตารางที่ 4.23 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.188	.035	-.113	1.348	2.043

a. Predictors: (Constant), avax5, avax2, avax4, avax3

b. Dependent Variable: Y

ตารางที่ 4.24 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย

ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.734	4	.434	.239	.914
	Residual	47.233	26	1.817		
	Total	48.968	30			

a. Predictors: (Constant), avax5, avax2, avax4, avax3

b. Dependent Variable: Y

ตารางที่ 4.25 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	3.500	.953		3.672	.001
	avax2	9.352E-16	1.167	.000	.000	1.000
	avax3	-.633	1.015	-.252	-.624	.538
	avax4	-.500	1.054	-.181	-.475	.639
	avax5	-.500	1.651	-.070	-.303	.764

a. Dependent Variable: Y

ตารางที่ 4.26 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์

Variables	B	SE_B	Beta	t	p-value
avax2	9.352E-16	1.167	.000	.000	1.000
avax3	-.633	1.015	-.252	-.624	.538
avax4	-.500	1.054	-.181	-.475	.639
avax5	-.500	1.651	-.070	-.303	.764
Constant 3.500; SE_{est} = ±1.348 R^2=.035; F=.239; p-value<.05					

จากตาราง 4.26 จะเห็นว่าตัวแปรอิสระในแต่ละระดับของอคติประเภท Availability มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อยู่ในระดับต่ำมาก โดยมี R^2 เป็น .035 นั่นสามารถร่วมกันพยากรณ์ความเสียหายของซอฟต์แวร์ได้ร้อยละ 3.5 โดยมีความคลาดเคลื่อนมาตรฐานในการพยากรณ์เท่ากับ ±1.348 และเมื่อพิจารณาค่าสัมประสิทธิ์การถดถอยของตัวพยากรณ์ พบว่าตัวแปรอิสระไม่สามารถพยากรณ์ความความเสียหายของซอฟต์แวร์ได้ (p-value > .05) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05

ตรวจสอบเงื่อนไขจากค่าสถิติทดสอบ Durbin-Watson = 2.043 อยู่ในช่วง 1.5-2.5 สรุปได้ว่าค่าคลาดเคลื่อนเป็นอิสระต่อกัน เป็นไปตามเงื่อนไขของการวิเคราะห์ถดถอย และสำหรับสมการพยากรณ์ความเสียหายของซอฟต์แวร์เมื่อนำตัวแปรอิสระทุกตัวเข้าสมการในรูปคะแนนดิบ จะเป็นดังนี้

$$\hat{Y}_{ava} = 3.5 - 0.633avax3 - 0.5avax4 - 0.5avax5 \quad (4.2)$$

ภายหลังจากได้สมการสำหรับการพยากรณ์ค่าความเสียหายของอคติประเภท Availability ของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์แล้ว จะนำไปทดสอบด้วยการคำนวณค่าความเสียหายประมาณเมื่อเกิดการมีอคติประเภทนี้ในระดับต่าง ๆ ต่อไป

4.4.3 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์จากอคติประเภท Confirmation

ตารางที่ 4.27 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย

Variables Entered/Removed

Model	Variables Entered	Variables Removed	Method
1	conx5, conx2, conx4, conx3	.	Enter

a. All requested variables entered

b. Dependent Variable: Y

ตารางที่ 4.28 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.362	.131	-.003	1.279	1.924

a. Predictors: (Constant), conx5, conx2, conx4, conx3

b. Dependent Variable: Y

ตารางที่ 4.29 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย

ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	6.419	4	1.605	.981	.435
	Residual	42.549	26	1.637		
	Total	48.968	30			

a. Predictors: (Constant), conx5, conx2, conx4, conx3

b. Dependent Variable: Y

ตารางที่ 4.30 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	4.000	1.279		3.127	.004
	conx2	-.333	1.477	-.078	-.226	.823
	conx3	-1.353	1.316	-.536	-1.028	.314
	conx4	-.667	1.348	-.241	-.494	.625
	conx5	-5.159E-15	1.809	.000	.000	1.000

a. Dependent Variable: Y

ตารางที่ 4.31 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์

Variables	B	SE_B	Beta	t	p-value
conx2	-.333	1.477	-.078	-.226	.823
conx3	-1.353	1.316	-.536	-1.028	.314
conx4	-.667	1.348	-.241	-.494	.625
conx5	-5.159E-15	1.809	.000	.000	1.000
Constant 4.000; SE_{est} = ±1.279 R^2=.131; F=.981; p-value<.05					

จากตาราง 4.31 จะเห็นว่าตัวแปรอิสระในแต่ละระดับของอคติประเภท Confirmation มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อยู่ในระดับต่ำมาก โดยมี R^2 เป็น .131 นั่นสามารถร่วมกันพยากรณ์ความเสียหายของซอฟต์แวร์ได้ร้อยละ 13.1 โดยมีความคลาดเคลื่อนมาตรฐานในการพยากรณ์เท่ากับ ±1.279 และเมื่อพิจารณาค่าสัมประสิทธิ์การถดถอยของตัวพยากรณ์ พบว่าตัวแปรอิสระไม่สามารถพยากรณ์ความความเสียหายของซอฟต์แวร์ได้ (p-value > .05) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05

ตรวจสอบเงื่อนไขจากค่าสถิติทดสอบ Durbin-Watson = 1.924 อยู่ในช่วง 1.5-2.5 สรุปได้ว่าค่าคลาดเคลื่อนเป็นอิสระต่อกัน เป็นไปตามเงื่อนไขของการวิเคราะห์ถดถอย และสำหรับสมการพยากรณ์ความเสียหายของซอฟต์แวร์เมื่อนำตัวแปรอิสระทุกตัวเข้าสมการในรูปคะแนนดิบ จะเป็นดังนี้

$$\hat{Y}_{con} = 4 - 0.333conx2 - 1.353conx3 - 0.667conx4 \quad (4.3)$$

ภายหลังจากได้สมการสำหรับการพยากรณ์ค่าความเสียหายของอคติประเภท Confirmation ของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์แล้ว จะนำไปทดสอบด้วยการคำนวณค่าความเสียหายประมาณเมื่อเกิดการมีอคติประเภทนี้ในระดับต่าง ๆ ต่อไป

4.4.4 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการออกแบบซอฟต์แวร์จากอคติประเภท Anchoring and adjustment

ตารางที่ 4.32 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย

Variables Entered/Removed

Model	Variables Entered	Variables Removed	Method
1	ancx5, ancx4, ancx2	.	Enter

a. Tolerance = .000 limits reached.

b. Dependent Variable: Y

ตารางที่ 4.33 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.282	.080	-.023	1.325	1.522

a. Predictors: (Constant), ancx5, ancx4, ancx2

b. Dependent Variable: Y

ตารางที่ 4.34 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย

ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	4.102	3	1.367	.779	.516
	Residual	47.382	27	1.755		
	Total	51.484	30			

a. Predictors: (Constant), ancx5, ancx4, ancx2

b. Dependent Variable: Y

ตารางที่ 4.35 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	3.000	.367		8.165	.000
	ancx2	-.273	.543	-.101	-.503	.619
	ancx4	-.600	.697	-.171	-.861	.397
	ancx5	1.000	1.006	.191	.994	.329

a. Dependent Variable: Y

ตารางที่ 4.36 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์

Variables	B	SE_B	Beta	t	p-value
ancx2	-.273	.543	-.101	-.503	.619
ancx4	-.600	.697	-.171	-.861	.397
ancx5	1.000	1.006	.191	.994	.329
Constant 3.000; SE_{est} = ±1.325 R^2=.080; F=.779; p-value<.05					

จากตาราง 4.36 จะเห็นว่าตัวแปรอิสระในแต่ละระดับของอคติประเภท Anchoring and adjustment มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อยู่ในระดับต่ำมาก โดยมี R^2 เป็น .080 นั่นสามารถร่วมกันพยากรณ์ความเสียหายของซอฟต์แวร์ได้ร้อยละ 8 โดยมีความคลาดเคลื่อนมาตรฐานในการพยากรณ์เท่ากับ ±1.325 และเมื่อพิจารณาค่าสัมประสิทธิ์การถดถอยของตัวพยากรณ์ พบว่าตัวแปรอิสระไม่สามารถพยากรณ์ความความเสียหายของซอฟต์แวร์ได้ (p-value > .05) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05

ตรวจสอบเงื่อนไขจากค่าสถิติทดสอบ Durbin-Watson = 1.522 อยู่ในช่วง 1.5-2.5 สรุปได้ว่าค่าคลาดเคลื่อนเป็นอิสระต่อกัน เป็นไปตามเงื่อนไขของการวิเคราะห์ถดถอย และสำหรับสมการพยากรณ์ความเสียหายของซอฟต์แวร์เมื่อนำตัวแปรอิสระทุกตัวเข้าสมการในรูปคะแนนดิบ จะเป็นดังนี้

$$\hat{Y}_{anc} = 4.5 - 0.273ancx2 - 0.600ancx4 - 1.000ancx5 \quad (4.4)$$

ภายหลังจากได้สมการสำหรับการพยากรณ์ค่าความเสียหายของอคติประเภท Anchoring and adjustment ของกระบวนการออกแบบซอฟต์แวร์แล้ว จะนำไปทดสอบด้วยการคำนวณค่าความเสียหายประมาณเมื่อเกิดการมีอคติประเภทนี้ในระดับต่าง ๆ ต่อไป

4.4.5 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการออกแบบซอฟต์แวร์จากอคติประเภท Availability

ตารางที่ 4.37 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย

Variables Entered/Removed

Model	Variables Entered	Variables Removed	Method
1	avax5, avax2, avax4, avax3	.	Enter

a. All requested variables entered

b. Dependent Variable: Y

ตารางที่ 4.38 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.533	.284	.174	1.191	1.900

a. Predictors: (Constant), avax5, avax2, avax4, avax3

b. Dependent Variable: Y

ตารางที่ 4.39 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย

ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	14.614	4	3.654	2.576	.061
	Residual	36.87	26	1.418		
	Total	51.484	30			

a. Predictors: (Constant), avax5, avax2, avax4, avax3

b. Dependent Variable: Y

ตารางที่ 4.40 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	3.667	.688		5.333	.000
	avax2	-1.167	.910	-.303	-1.283	.211
	avax3	-1.485	.776	-.551	-1.914	.067
	avax4	-.567	.784	-.206	-.723	.476
	avax5	.667	.972	.153	.686	.499

a. Dependent Variable: Y

ตารางที่ 4.41 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์

Variables	B	SE_B	Beta	t	p-value
avax2	-1.167	.910	-.303	-1.283	.211
avax3	-1.485	.776	-.551	-1.914	.067
avax4	-.567	.784	-.206	-.723	.476
avax5	.667	.972	.153	.686	.499
Constant 3.667; SE_{est} = ±1.191 R^2=.284; F=2.576; p-value<.05					

จากตาราง 4.41 จะเห็นว่าตัวแปรอิสระในแต่ละระดับของอคติประเภท Availability มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อยู่ในระดับค่อนข้างต่ำ โดยมี R^2 เป็น .284 นั่นสามารถร่วมกันพยากรณ์ความเสียหายของซอฟต์แวร์ได้ร้อยละ 28.4 โดยมีความคลาดเคลื่อนมาตรฐานในการพยากรณ์เท่ากับ ±1.191 และเมื่อพิจารณาค่าสัมประสิทธิ์การถดถอยของตัวพยากรณ์ พบว่าตัวแปรอิสระไม่สามารถพยากรณ์ความความเสียหายของซอฟต์แวร์ได้ (p-value > .05) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05

ตรวจสอบเงื่อนไขจากค่าสถิติทดสอบ Durbin-Watson = 1.900 อยู่ในช่วง 1.5-2.5 สรุปได้ว่าค่าคลาดเคลื่อนเป็นอิสระต่อกัน เป็นไปตามเงื่อนไขของการวิเคราะห์ถดถอย และสำหรับสมการพยากรณ์ความเสียหายของซอฟต์แวร์เมื่อนำตัวแปรอิสระทุกตัวเข้าสมการในรูปคะแนนดิบ จะเป็นดังนี้

$$\hat{Y}_{ava} = 3.667 - 1.167avax2 - 1.485avax3 - 0.567avax4 + 0.667avax5 \quad (4.5)$$

ภายหลังจากได้สมการสำหรับการพยากรณ์ค่าความเสียหายของอคติประเภท Availability ของกระบวนการออกแบบซอฟต์แวร์แล้ว จะนำไปทดสอบด้วยการคำนวณค่าความเสียหายประมาณเมื่อเกิดการมีอคติประเภทนี้ในระดับต่าง ๆ ต่อไป

4.4.6 ผลการประมาณค่าความเสียหายของซอฟต์แวร์ในกระบวนการออกแบบซอฟต์แวร์จากอคติประเภท Confirmation

ตารางที่ 4.42 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย

Variables Entered/Removed

Model	Variables Entered	Variables Removed	Method
1	conx5, conx2, conx3, conx4	.	Enter

a. All requested variables entered

b. Dependent Variable: Y

ตารางที่ 4.43 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.473	.224	.104	1.240	1.816

a. Predictors: (Constant), conx5, conx2, conx3, conx4

b. Dependent Variable: Y

ตารางที่ 4.44 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย

ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	11.518	4	2.880	1.873	.145
	Residual	39.966	26	1.537		
	Total	51.484	30			

a. Predictors: (Constant), conx5, conx2, conx3, conx4

b. Dependent Variable: Y

ตารางที่ 4.45 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	3.000	1.240		2.420	.023
	conx2	-.667	1.339	-.204	-.498	.623
	conx3	-.778	1.307	-.274	-.595	.557
	conx4	.385	1.287	.147	.299	.767
	conx5	1.000	1.518	.191	.659	.516

a. Dependent Variable: Y

ตารางที่ 4.46 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์

Variables	B	SE_B	Beta	t	p-value
conx2	-.667	1.339	-.204	-.498	.623
conx3	-.778	1.307	-.274	-.595	.557
conx4	.385	1.287	.147	.299	.767
conx5	1.000	1.518	.191	.659	.516
Constant 3.000; SE_{est} = ±1.240 R^2=.224; F=1.873; p-value<.05					

จากตาราง 4.46 จะเห็นว่าตัวแปรอิสระในแต่ละระดับของอคติประเภท Confirmation มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อยู่ในระดับต่ำมาก โดยมี R^2 เป็น .224 นั่นสามารถร่วมกันพยากรณ์ความเสียหายของซอฟต์แวร์ได้ร้อยละ 22.4 โดยมีความคลาดเคลื่อนมาตรฐานในการพยากรณ์เท่ากับ ±1.240 และเมื่อพิจารณาค่าสัมประสิทธิ์การถดถอยของตัวพยากรณ์ พบว่าตัวแปรอิสระไม่สามารถพยากรณ์ความความเสียหายของซอฟต์แวร์ได้ (p-value > .05) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05

ตรวจสอบเงื่อนไขจากค่าสถิติทดสอบ Durbin-Watson = 1.816 อยู่ในช่วง 1.5-2.5 สรุปได้ว่าค่าคลาดเคลื่อนเป็นอิสระต่อกัน เป็นไปตามเงื่อนไขของการวิเคราะห์ถดถอย และสำหรับสมการพยากรณ์ความเสียหายของซอฟต์แวร์เมื่อนำตัวแปรอิสระทุกตัวเข้าสมการในรูปคะแนนดิบ จะเป็นดังนี้

$$\hat{Y}_{con} = 3 - 0.667conx2 - 0.778conx3 + 0.385conx4 + 1.000conx5 \quad (4.6)$$

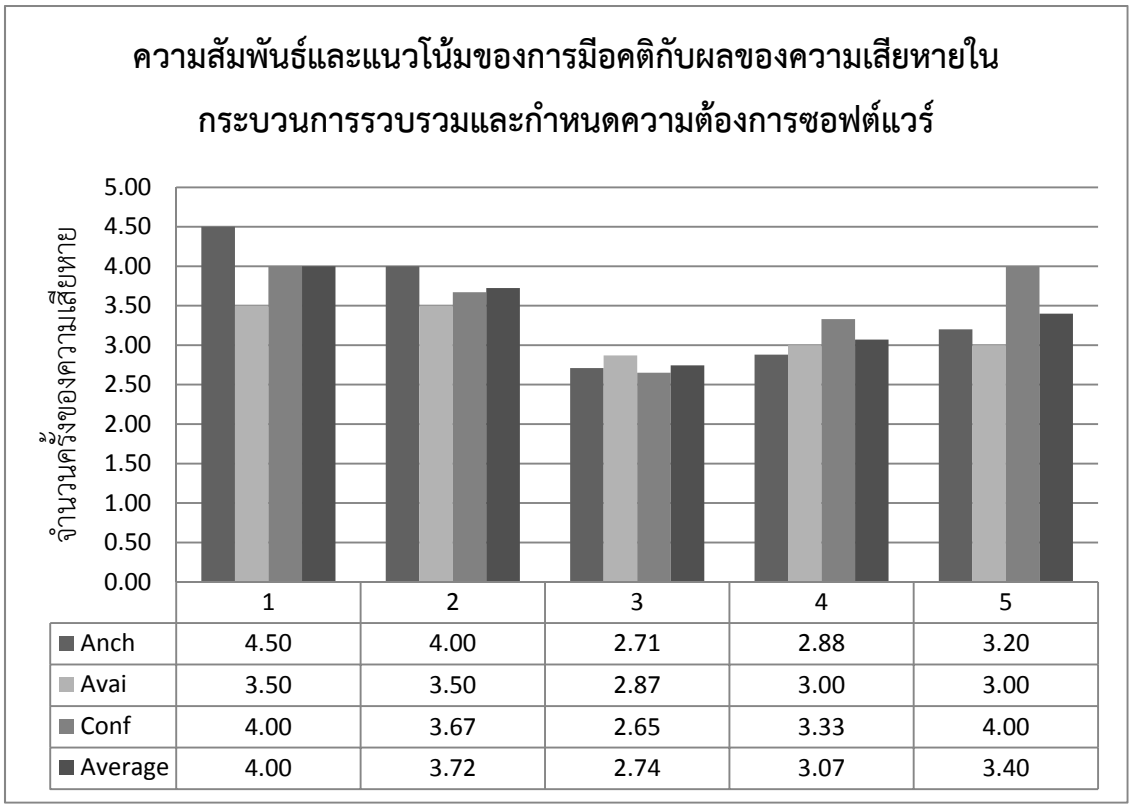
ภายหลังจากได้สมการสำหรับการพยากรณ์ค่าความเสียหายของอคติประเภท Confirmation ของกระบวนการออกแบบซอฟต์แวร์แล้ว จะนำไปทดสอบด้วยการคำนวณค่าความเสียหายประมาณเมื่อเกิดการมีอคติประเภทนี้ในระดับต่าง ๆ ต่อไป

4.5 วิเคราะห์ผลของการประมาณความเสียหาย

ภายหลังจากที่ได้สมการพยากรณ์ค่าประมาณความเสียหายจากหัวข้อที่ 4.3 แล้ว จะทำการจำแนกสมการตามกระบวนการทั้งสอง โดยในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ใช้สมการที่ 4.1-4.3 และกระบวนการออกแบบซอฟต์แวร์ใช้สมการที่ 4.4-4.6 โดยต้องการ

ศึกษาระดับความสัมพันธ์และแนวโน้มของการมีอคติกับผลของความเสียหายว่ามีลักษณะเป็นเช่นไร โดยทำการกรอกค่าการมีอคติในแต่ละระดับความรุนแรงของอคติแต่ละตัว ดังนี้

4.5.1 ระดับความสัมพันธ์และแนวโน้มของการมีอคติกับผลของความเสียหายของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์



	1	2	3	4	5
Anch	4.50	4.00	2.71	2.88	3.20
Avai	3.50	3.50	2.87	3.00	3.00
Conf	4.00	3.67	2.65	3.33	4.00
Average	4.00	3.72	2.74	3.07	3.40

ภาพที่ 4.2 กราฟแสดงระดับความสัมพันธ์และแนวโน้มของการมีอคติกับผลของความเสียหายของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์

จากภาพ 4.2 ในแกน X ที่ระบุเป็น 1, 2, 3, 4, 5 นั้นแทนระดับของการมีอคติในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์จากน้อยไปมากตามตารางที่ 3.1 พบว่า

หากมีอคติในระดับน้อยมาก (1) นั่นคือ 0-20% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่ 4.1, 4.2, 4.3 ตามลำดับ ได้ค่าดังนี้คือ 4.50, 3.50, 4.00 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคติน้อยจะได้เท่ากับ 4.00 ครั้งต่อเดือน

หากมีอคติในระดับน้อย (2) นั่นคือ 21-40% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่ 4.1, 4.2, 4.3 ตามลำดับ ได้ค่าดังนี้คือ 4.00, 3.50, 3.67 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคติน้อยจะได้เท่ากับ 3.72 ครั้งต่อเดือน

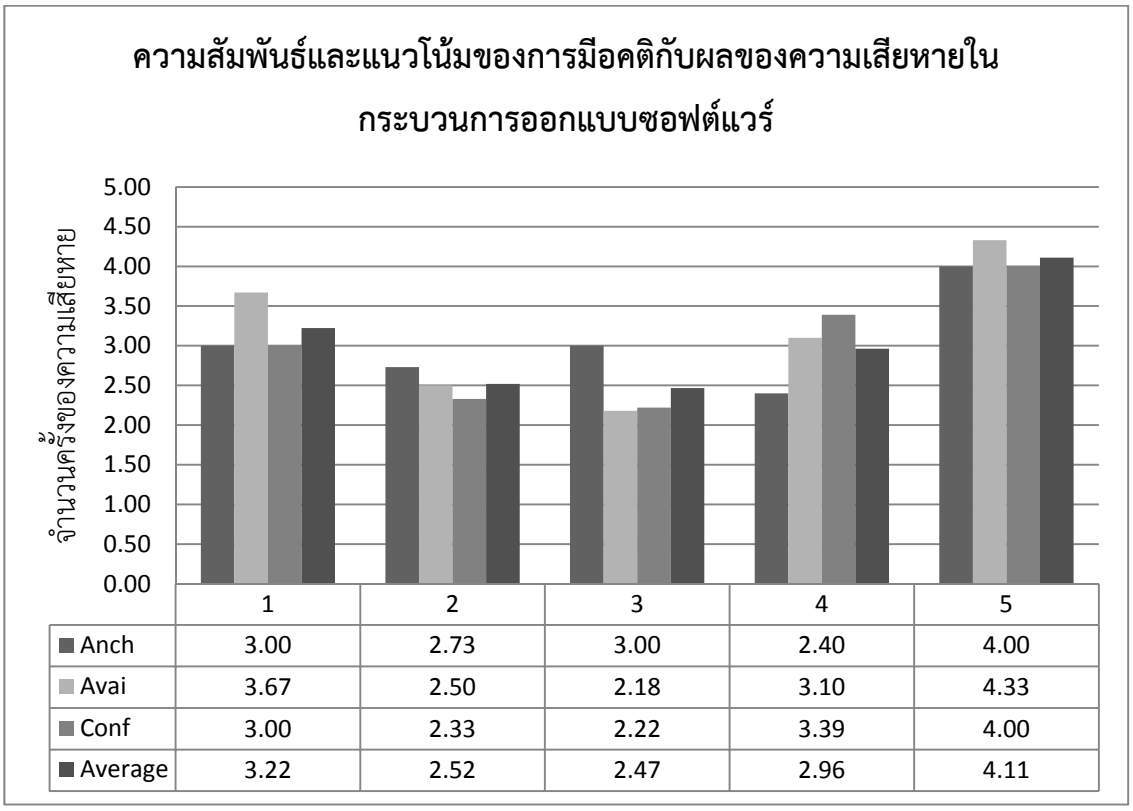
หากมีอคติในระดับปานกลาง (3) นั่นคือ 41-60% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่ 4.1, 4.2, 4.3 ตามลำดับ ได้ค่าดังนี้คือ 2.71, 2.87, 2.65 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคติน้อยจะได้เท่ากับ 2.74 ครั้งต่อเดือน

หากมีอคติในระดับมาก (4) นั่นคือ 61-80% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่ 4.1, 4.2, 4.3 ตามลำดับ ได้ค่าดังนี้คือ 2.88, 3.00, 3.33 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคติน้อยจะได้เท่ากับ 3.07 ครั้งต่อเดือน

หากมีอคติในระดับมากที่สุด (5) นั่นคือ 81-100% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่ 4.1, 4.2, 4.3 ตามลำดับ ได้ค่าดังนี้คือ 3.20, 3.00, 4.00 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคติน้อยจะได้เท่ากับ 3.40 ครั้งต่อเดือน

นั่นแสดงให้เห็นว่าการที่มีอคติในการทำงานในแต่ละระดับของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ก็อาจจะไม่จำเป็นที่จะต้องส่งผลเสียเสมอไป และการที่มีอคติทั้งสามประเภทมากหรือน้อยจนเกินไปสามารถทำให้เกิดผลเสียได้มากกว่าการที่มีอคติในระดับปานกลาง โดยทั้งสามประเภทเป็นไปในทิศทางเดียวกันที่ในการที่มีอคติในระดับปานกลางทำให้เกิดความเสียหายน้อยที่สุด

4.5.2 ระดับความสัมพันธ์และแนวโน้มของการมีอคติกับผลของความเสียหายของกระบวนการออกแบบซอฟต์แวร์



	1	2	3	4	5
Anch	3.00	2.73	3.00	2.40	4.00
Avai	3.67	2.50	2.18	3.10	4.33
Conf	3.00	2.33	2.22	3.39	4.00
Average	3.22	2.52	2.47	2.96	4.11

ภาพที่ 4.3 กราฟแสดงระดับความสัมพันธ์และแนวโน้มของการมีอคติกับผลของความเสียหายของกระบวนการออกแบบซอฟต์แวร์

จากภาพ 4.3 ในแกน X ที่ระบุเป็น 1, 2, 3, 4, 5 นั้นแทนระดับของการมีอคติในกระบวนการออกแบบซอฟต์แวร์จากน้อยไปมากตามตารางที่ 3.1 พบว่า

หากมีอคติในระดับน้อยมาก (1) นั่นคือ 0-20% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่ 4.4, 4.5, 4.6 ตามลำดับ ได้ค่าดังนี้คือ 3.00, 3.67, 3.00 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคติน้อยจะได้เท่ากับ 3.22 ครั้งต่อเดือน

หากมีอคติในระดับน้อย (2) นั่นคือ 21-40% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่

4.4, 4.5, 4.6 ตามลำดับ ได้ค่าดังนี้คือ 2.73, 2.50, 2.33 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคตินอ้ยจะได้เท่ากับ 2.52 ครั้งต่อเดือน

หากมีอคติในระดับปานกลาง (3) นั่นคือ 41-60% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่ 4.4, 4.5, 4.6 ตามลำดับ ได้ค่าดังนี้คือ 3.00, 2.18, 2.22 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคตินอ้ยจะได้เท่ากับ 2.47 ครั้งต่อเดือน

หากมีอคติในระดับมาก (4) นั่นคือ 61-80% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่ 4.4, 4.5, 4.6 ตามลำดับ ได้ค่าดังนี้คือ 2.40, 3.10, 3.39 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคตินอ้ยจะได้เท่ากับ 2.96 ครั้งต่อเดือน

หากมีอคติในระดับมากที่สุด (5) นั่นคือ 81-100% ของการทำงาน ทั้ง Anchoring and adjustment, Availability, Confirmation จะมีค่าความเสียหายโดยประมาณจากสมการพยากรณ์ที่ 4.4, 4.5, 4.6 ตามลำดับ ได้ค่าดังนี้คือ 4.00, 4.33, 4.00 ครั้งต่อเดือนตามลำดับ และเมื่อหาค่าเฉลี่ยของอคติทั้งสามประเภทนี้ที่ระดับการมีอคตินอ้ยจะได้เท่ากับ 4.11 ครั้งต่อเดือน

นั่นแสดงให้เห็นว่าการที่มีอคติในการทำงานในแต่ละระดับของกระบวนการออกแบบซอฟต์แวร์ก็อาจจะไม่จำเป็นที่จะต้องส่งผลเสียเสมอไป และการที่มีอคติในแต่ละประเภทมากหรือน้อยจนเกินไปสามารถทำให้เกิดผลเสียได้มากกว่าการที่มีอคติในระดับปานกลาง โดยเฉพาะในอคติประเภท Availability และ Confirmation ที่มีแนวโนมไปในทิศทางเดียวกัน แต่สำหรับอคติประเภท Anchoring and adjustment นั้นพบว่าในระดับการมีอคตินอ้ยมากถึงน้อยมีทิศทางไปในทางเดียวกับกับอคติ Availability และ Confirmation แต่ในระดับการมีอคติปานกลางกลับมีค่าความเสียหายที่มากขึ้นแต่ก็ไม่ได้มากไปเกินกว่าในระดับการมีอคติในระดับน้อยมาก และเมื่อระดับการมีอคติจากระดับมากถึงมากที่สุดมีทิศทางไปในทางเดียวกับกับอคติ Availability และ Confirmation โดยที่ระดับการมีอคติจากระดับมากของ Anchoring and adjustment มีความเสียหายน้อยที่สุด

4.6 จำแนกความเสียหายออกเป็นระดับความรุนแรง

โดยค่าเฉลี่ยความเสียหายโดยประมาณ ($\bar{Y}$) จากอคติทั้งสามประเภทของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์และกระบวนการออกแบบซอฟต์แวร์ จะนำไปจำแนกออกเป็นแต่ละระดับความรุนแรงด้วยสมการ 4.7 ซึ่งสมการดังกล่าวมีอธิบายถึงที่มาและการคำนวณค่าคงที่ไว้แล้วในหัวข้อ 3.1.8

$$R_i = \frac{k_i \times \bar{Y}}{100} \quad (4.7)$$

โดยที่มี k_i เป็นค่าคงที่โดย i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception โดยสำหรับกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์มีค่าคงที่เรียงตามลำดับดังนี้ 3.19, 7.45, 28.72, 30.85, 29.79 ตามลำดับ

และค่าเฉลี่ยความเสียหายโดยประมาณของกระบวนการออกแบบซอฟต์แวร์จะนำไปจำแนกออกเป็นแต่ละระดับความเสียหายด้วยสูตรเดียวกันโดยที่มี k_i เป็นค่าคงที่โดย i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception โดยสำหรับกระบวนการออกแบบซอฟต์แวร์มีค่าคงที่เรียงตามลำดับดังนี้ 2.25, 13.48, 41.57, 35.96, 6.74 ตามลำดับ

ซึ่งค่า R_i หรือค่าความเสียหายในแต่ละระดับความรุนแรงที่ได้นี้ก็จะสามารถนำไปใช้ในการหาค่าความเสี่ยงจากสมการความเสี่ยง (3.11) ได้ต่อไป

4.7 วิเคราะห์ข้อมูลที่ได้จากกลุ่มตัวอย่างสำหรับการวิเคราะห์ความน่าจะเป็น

การวิเคราะห์หาความน่าจะเป็นของการเกิดความเสียหายของซอฟต์แวร์ซึ่งกำหนดไว้ว่าจะเกิดความเสียหายขึ้นในระยะเวลาหนึ่งเดือน ซึ่งจะใช้วิธีการหาความน่าจะเป็นแบบปัวซอง ซึ่งจะเป็นต้องทราบค่าเฉลี่ยของเหตุการณ์ในช่วงระยะเวลาที่กำหนดหรือค่า m ซึ่งค่าเฉลี่ยนี้สามารถหาได้จากแบบสอบถาม ซึ่งข้อมูลที่จะได้จากแบบสอบถามแสดงโดยจำแนกออกเป็นแต่ละระดับความรุนแรงได้ดังนี้

ตารางที่ 4.47 ค่าความเสียหายในแต่ละระดับและค่า **m** ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์

No.	Critical	Major	Average	Minor	Exception
1	0	0	1	1	0
2	0	0	1	1	1
3	0	0	1	2	2
4	0	2	1	0	0
5	0	0	2	1	0
6	0	0	1	0	3
7	2	0	0	2	1
8	0	0	1	0	3
9	0	0	1	0	3
10	0	0	1	1	0
11	0	0	0	0	1
12	0	0	1	0	3
13	0	1	1	0	0
14	0	0	1	0	3
15	0	0	1	1	1
16	0	2	0	3	0
17	0	0	1	1	0
18	0	0	1	1	0
19	0	0	0	2	1
20	0	0	1	1	0
21	0	1	0	1	1
22	0	0	1	1	0
23	0	0	0	2	3
24	0	0	0	3	0
25	0	0	1	0	0
26	0	0	0	1	0

ตารางที่ 4.47 ค่าความเสียหายในแต่ละระดับและค่า m ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ (ต่อ)

No.	Critical	Major	Average	Minor	Exception
27	0	1	2	2	0
28	1	0	1	1	0
29	0	0	3	0	0
30	0	0	2	1	1
31	0	0	0	0	1
m	**0.10**	**0.23**	**0.87**	**0.94**	**0.90**

ตารางที่ 4.48 ค่าความเสียหายในแต่ละระดับและค่า m ในกระบวนการออกแบบซอฟต์แวร์

No.	Critical	Major	Average	Minor	Exception
1	0	0	3	0	0
2	0	0	1	0	0
3	0	0	2	2	0
4	0	0	1	2	0
5	0	1	2	1	1
6	0	1	2	0	0
7	0	2	0	1	0
8	0	0	1	1	0
9	0	0	1	1	0
10	0	0	0	1	0
11	0	0	0	0	1
12	0	1	2	1	1
13	1	1	2	1	0
14	0	1	1	0	0
15	0	0	1	1	0
16	0	0	1	1	1

ตารางที่ 4.48 ค่าความเสียหายในแต่ละระดับและค่า m ในกระบวนการออกแบบซอฟต์แวร์ (ต่อ)

No.	Critical	Major	Average	Minor	Exception
17	0	2	0	3	0
18	1	0	4	0	0
19	0	2	0	0	0
20	0	0	1	1	0
21	0	0	2	1	0
22	0	0	1	1	0
23	0	1	0	1	1
24	0	0	0	3	0
25	0	0	1	0	0
26	0	0	0	0	1
27	0	0	0	2	0
28	0	0	2	1	0
29	0	0	2	2	0
30	0	0	2	2	0
31	0	0	2	2	0
m	**0.06**	**0.39**	**1.19**	**1.03**	**0.19**

ซึ่งค่าเฉลี่ย m ที่ได้จากตารางนี้จะนำไปคำนวณด้วยสมการความน่าจะเป็นแบบปัวซองในหัวข้อที่ 3.1.8 โดยผลที่ได้จากการแทนค่า m ในสมการดังกล่าวจะทำให้ได้ค่าความน่าจะเป็นหรือ P_i ได้ดังนี้

ตารางที่ 4.49 ค่าความน่าจะเป็นด้วยวิธีปัวซองของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์

	Critical	Major	Average	Minor	Exception
P_i	0.09	0.20	0.58	0.61	0.59

ตารางที่ 4.50 ค่าความน่าจะเป็นด้วยวิธีปัวซองของกระบวนการออกแบบซอฟต์แวร์

	Critical	Major	Average	Minor	Exception
P_i	0.06	0.32	0.70	0.64	0.18

จากตารางพบว่า ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์และออกแบบซอฟต์แวร์มีค่าความน่าจะเป็นที่จะเกิดความเสียหายในระดับวิกฤติ (Critical) น้อยที่สุด โดยในกระบวนการแรกพบว่าค่าความน่าจะเป็นที่มีค่ามากที่สุดอยู่ในระดับที่จะเกิดความเสียหายน้อย (Minor) ส่วนกระบวนการที่สองพบว่าค่าความน่าจะเป็นที่มีค่ามากที่สุดอยู่ในระดับที่จะเกิดความเสียหายปานกลาง (Average) หลังจากที่ทราบค่าความน่าจะเป็นของการเกิดความเสียหายในแต่ละระดับแล้ว ก็จะสามารถนำค่าความน่าจะเป็นนี้ไปเป็นค่าคงที่ในการหาค่าความเสี่ยงจากสมการความเสี่ยงได้ต่อไป

4.8 คำนวณและจัดลำดับค่าความเสี่ยงและสร้างตารางความเสี่ยง

หลังจากที่ได้ค่า R_i และ P_i แล้ว ก็จะสามารถหาค่าความเสี่ยงของความเสียหายของซอฟต์แวร์ได้จากสมการความเสี่ยงที่ได้กล่าวไปในบทที่ 3 ออกมาเป็นในแต่ละระดับความรุนแรง ซึ่งสามารถนำความเสี่ยงทั้ง 5 ระดับความรุนแรงมารวมกันเป็นความเสี่ยงรวมได้ ซึ่งมีเหตุการณ์ที่จะใช้ในการคำนวณจากการมีอคติในการทำงานในทุกรูปแบบแล้วจะได้ทั้งสิ้น 125 รูปแบบต่อหนึ่งกระบวนการ

4.8.1 ค่าความเสี่ยงและตารางความเสี่ยงของกระบวนการรวมรวบและกำหนดความต้องการซอฟต์แวร์

สำหรับเหตุการณ์ที่จะใช้ในการคำนวณจากการมีอคติในการทำงานในทุกรูปแบบแล้วจะได้ทั้งสิ้น 125 รูปแบบแสดงไว้ดังตารางที่ 4.51

ตารางที่ 4.51 เหตุการณ์ที่เกิดการมีอคติในการทำงานในทุกรูปแบบ

เหตุการณ์ที่เกิดการมีอคติในการทำงานในทุกรูปแบบ				
111	112	113	114	115
121	122	123	124	125
131	132	133	134	135
141	142	143	144	145
151	152	153	154	155
211	212	213	214	215
221	222	223	224	225
231	232	233	234	235
241	242	243	244	245
251	252	253	254	255
311	312	313	314	315
321	322	323	324	325
331	332	333	334	335
341	342	343	344	345
351	352	353	354	355
411	412	413	414	415
421	422	423	424	425
431	432	433	434	435
441	442	443	444	445
451	452	453	454	455
511	512	513	514	515
521	522	523	524	525
531	532	533	534	535
541	542	543	544	545
551	552	553	554	555

สำหรับการแปลความหมายของเลขในตาราง แปลความได้ดังนี้ตัวเลขในหลักร้อยแทนการมีอคติประเภท Anchoring and adjustment ตัวเลขในหลักสิบแทนการมีอคติประเภท Availability

และตัวเลขในหลักหน่วยแทนการมีอคติประเภท Confirmation และค่าของตัวเลข 1, 2, 3, 4, 5 มีความหมายดังตารางที่ 3.1 ดังนั้นในช่องที่มีค่า 111 นั้นเท่ากับ การมีอคติในการทำงานที่ระดับน้อยมาก นั่นคือมีอคติประเภท Anchoring and adjustment, Availability, และ Confirmation อยู่ในระดับ 0-20% ของการทำงาน หรืออีกตัวอย่างหนึ่ง ในช่องที่มีค่า 135 นั้นเท่ากับ การมีอคติประเภท Anchoring and adjustment, Availability, และ Confirmation อยู่ในระดับ 0-20%, 41-60%, และ 81-100% ของการทำงาน เป็นต้น และเมื่อคำนวณความเสี่ยงตามรูปแบบข้างต้นแล้ว จะได้ค่าความเสี่ยงของแต่ละรูปแบบของการมีอคติได้ดังตารางที่ 4.52

ตารางที่ 4.52 ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ

ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ				
2.20	2.14	1.95	2.08	2.20
2.20	2.14	1.95	2.08	2.20
2.08	2.02	1.83	1.96	2.08
2.11	2.05	1.86	1.96	2.11
2.11	2.05	1.86	1.96	2.11
2.11	2.05	1.85	1.98	2.11
2.11	2.05	1.85	1.98	2.11
1.98	1.93	1.75	1.87	1.98
2.01	1.95	1.76	1.89	2.01
2.01	1.95	1.76	1.89	2.01
1.87	1.81	1.62	1.76	1.87
1.87	1.81	1.62	1.76	1.87
1.76	1.70	1.52	1.62	1.76
1.79	1.72	1.52	1.66	1.79
1.79	1.72	1.52	1.66	1.79

ตารางที่ 4.52 ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ (ต่อ)

ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ				
1.89	1.85	1.66	1.78	1.89
1.89	1.85	1.66	1.78	1.89
1.79	1.73	1.52	1.66	1.79
1.81	1.76	1.58	1.69	1.81
1.81	1.76	1.58	1.69	1.81
1.96	1.89	1.72	1.84	1.96
1.96	1.89	1.72	1.84	1.96
1.85	1.79	1.61	1.72	1.85
1.87	1.81	1.62	1.76	1.87
1.87	1.81	1.62	1.76	1.87

จากตาราง 4.52 ยกตัวอย่างค่า 2.20 ในคอลัมน์แรกและแถวแรกของตาราง ได้มากจากการที่มีอคติในรูปแบบ 111 และคำนวณได้ค่าความเสียหายออกมาได้ 4 ครั้ง แล้วนำไปจำแนกเพื่อให้ได้ค่า R_i ทั้ง 5 ค่า แล้วนำมาคูณกับค่า P_i ทั้ง 5 ค่า จะได้ค่า $Risk_i$ ทั้ง 5 ค่า แล้วนำมารวมกันเป็นความเสี่ยงรวมดังปรากฏในตาราง

สามารถจัดลำดับความเสี่ยงจากตารางข้างต้นเพื่อดูลำดับความเสี่ยงจากมากไปหาน้อย ในเครื่องมือการเรียงลำดับของ Excel ด้วยการสลับแถวไปเป็นคอลัมน์ แล้วทำการเรียงจากน้อยไปหามากในแต่ละคอลัมน์ จากนั้นทำการสลับแถวไปเป็นคอลัมน์อีกครั้ง แล้วทำการจัดเรียงแต่ละคอลัมน์จากมากไปหาน้อย จะได้ค่าความเสี่ยงของการมีอคติที่จัดลำดับแล้ว

หรือทำการเรียงจากมากไปน้อยในแต่ละคอลัมน์ จากนั้นดูจากหัวของแต่ละคอลัมน์ นำมาเปรียบเทียบกันจากน้อยไปมาก แล้วเรียงใหม่ทั้งคอลัมน์จากน้อยไปมากเช่นกัน ซึ่งทั้งสองวิธีจะได้ผลเหมือนกัน ดังนี้

ตารางที่ 4.53 ตารางจัดลำดับความเสี่ยง

ตารางจัดลำดับความเสี่ยง					
มาก	1.95	2.08	2.14	2.20	2.20
	1.95	2.08	2.14	2.20	2.20
	1.86	1.98	2.05	2.11	2.11
	1.86	1.98	2.05	2.11	2.11
	1.85	1.96	2.05	2.11	2.11
	1.85	1.96	2.05	2.11	2.11
	1.83	1.96	2.02	2.08	2.08
	1.76	1.89	1.95	2.01	2.01
	1.76	1.89	1.95	2.01	2.01
	1.75	1.87	1.93	1.98	1.98
	1.72	1.84	1.89	1.96	1.96
	1.72	1.84	1.89	1.96	1.96
	1.66	1.78	1.85	1.89	1.89
	1.66	1.78	1.85	1.89	1.89
	1.62	1.76	1.81	1.87	1.87
	1.62	1.76	1.81	1.87	1.87
	1.62	1.76	1.81	1.87	1.87
	1.62	1.76	1.81	1.87	1.87
	1.61	1.72	1.79	1.85	1.85
	1.58	1.69	1.76	1.81	1.81
	1.58	1.69	1.76	1.81	1.81
	1.52	1.66	1.73	1.79	1.79
	1.52	1.66	1.72	1.79	1.79
	1.52	1.66	1.72	1.79	1.79
	1.52	1.62	1.70	1.76	1.76
น้อย					มาก

หากใช้แบบจำลองนี้แล้วพบว่ามีความเสี่ยงอยู่ที่ 1.87 นั่นหมายความว่ามีความเสี่ยงอยู่ในระดับปานกลางค่อนข้างสูงจากตารางจัดลำดับความเสี่ยงนี้

4.8.2 ค่าความเสี่ยงและตารางความเสี่ยงของกระบวนการออกแบบซอฟต์แวร์

สำหรับเหตุการณ์ที่จะใช้ในการคำนวณจากการมีอคติในการทำงานในทุกรูปแบบแล้วจะได้ทั้งสิ้น 125 รูปแบบแสดงไว้ดังตารางที่ 4.51 ในหัวข้อ 4.7.1 เช่นกัน และเมื่อคำนวณความเสี่ยงตามรูปแบบข้างต้นแล้ว จะได้ค่าความเสี่ยงของแต่ละรูปแบบของการมีอคติได้ดังนี้

ตารางที่ 4.54 ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ

ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ				
1.87	1.75	1.71	1.92	2.05
1.64	1.51	1.48	1.71	1.83
1.57	1.45	1.43	1.64	1.77
1.75	1.62	1.60	1.83	1.94
1.99	1.87	1.85	2.06	2.18
1.81	1.69	1.64	1.88	2.01
1.59	1.47	1.43	1.64	1.78
1.53	1.39	1.37	1.60	1.72
1.70	1.57	1.55	1.78	1.89
1.94	1.81	1.78	2.02	2.13
1.87	1.75	1.71	1.92	2.05
1.64	1.51	1.48	1.71	1.83
1.57	1.45	1.43	1.64	1.77
1.75	1.62	1.60	1.83	1.94
1.99	1.87	1.85	2.06	2.18

ตารางที่ 4.54 ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ (ต่อ)

ความเสี่ยงที่จะเกิดความเสียหายทุกรูปแบบของการมีอคติ				
1.75	1.61	1.60	1.82	1.94
1.51	1.39	1.37	1.60	1.72
1.47	1.33	1.32	1.55	1.64
1.64	1.51	1.48	1.71	1.83
1.88	1.75	1.74	1.97	2.06
2.05	1.92	1.91	2.13	2.26
1.83	1.70	1.69	1.91	2.02
1.77	1.64	1.61	1.85	1.97
1.94	1.82	1.79	2.02	2.14
2.18	2.05	2.03	2.28	2.39

จากตาราง 4.54 ยกตัวอย่างค่า 1.87 ในคอลัมน์แรกและแถวแรกของตาราง ได้มากจากการที่มีอคติในรูปแบบ 111 และคำนวณได้ค่าความเสียหายออกมาได้ 4 ครั้ง แล้วนำไปจำแนกเพื่อให้ได้ค่า R_i ทั้ง 5 ค่า แล้วนำมาคูณกับค่า P_i ทั้ง 5 ค่า จะได้ค่า $Risk_i$ ทั้ง 5 ค่า แล้วนำมารวมกันเป็นความเสี่ยงรวมดังปรากฏในตาราง

สามารถจัดลำดับความเสี่ยงจากตารางข้างต้นเพื่อดูลำดับความเสี่ยงจากมากไปหาน้อย ในเครื่องมือการเรียงลำดับของ Excel ด้วยการสลับแถวไปเป็นคอลัมน์ แล้วทำการเรียงจากน้อยไปหามากในแต่ละคอลัมน์ จากนั้นทำการสลับแถวไปเป็นคอลัมน์อีกครั้ง แล้วทำการจัดเรียงแต่ละคอลัมน์จากมากไปหาน้อย จะได้ค่าความเสี่ยงของการมีอคติที่จัดลำดับแล้ว

หรือทำการเรียงจากมากไปน้อยในแต่ละคอลัมน์ จากนั้นดูจากหัวของแต่ละคอลัมน์ นำมาเปรียบเทียบกันจากน้อยไปมาก แล้วเรียงใหม่ทั้งคอลัมน์จากน้อยไปมากเช่นกัน ซึ่งทั้งสองวิธีจะได้ผลเหมือนกัน ดังนี้

ตารางที่ 4.55 ตารางจัดลำดับความเสี่ยง

ตารางจัดลำดับความเสี่ยง					
มาก	2.03	2.05	2.18	2.28	2.39
	1.91	1.92	2.05	2.13	2.26
	1.85	1.87	1.99	2.06	2.18
	1.85	1.87	1.99	2.06	2.18
	1.79	1.82	1.94	2.02	2.14
	1.78	1.81	1.94	2.02	2.13
	1.74	1.75	1.88	1.97	2.06
	1.71	1.75	1.87	1.92	2.05
	1.71	1.75	1.87	1.92	2.05
	1.69	1.70	1.83	1.91	2.02
	1.64	1.69	1.81	1.88	2.01
	1.61	1.64	1.77	1.85	1.97
	1.60	1.62	1.75	1.83	1.94
	1.60	1.62	1.75	1.83	1.94
	1.60	1.61	1.75	1.82	1.94
	1.55	1.57	1.70	1.78	1.89
	1.48	1.51	1.64	1.71	1.83
	1.48	1.51	1.64	1.71	1.83
	1.48	1.51	1.64	1.71	1.83
	1.43	1.47	1.59	1.64	1.78
	1.43	1.45	1.57	1.64	1.77
	1.43	1.45	1.57	1.64	1.77
	1.37	1.39	1.53	1.60	1.72
	1.37	1.39	1.51	1.60	1.72
	1.32	1.33	1.47	1.55	1.64
น้อย					มาก

หากใช้แบบจำลองนี้แล้วพบว่ามีความเสี่ยงอยู่ที่ 2.18 นั่นหมายความว่ามีความเสี่ยงอยู่ในระดับสูงจากตารางจัดลำดับความเสี่ยงนี้

4.9 อภิปรายผล

ในการนำค่าความเสี่ยงไปใช้ในการบริหารจัดการความเสี่ยงนั้น ควรพิจารณาทั้งภาพรวมและในแต่ละระดับความรุนแรงด้วย ถึงแม้ว่าแบบจำลองและเครื่องมือที่สนับสนุนแบบจำลองนี้จะให้ค่าความเสี่ยงที่อาจจะมีความสามารถในการพยากรณ์ความเสียหายได้ค่อนข้างต่ำ โดยอ้างอิงจากผลการพยากรณ์จากสมการความเสียหาย แต่ก็ใช่ว่าจะไม่ส่งผลต่อความเสียหายเลย ดังนั้นการบริหารจัดการความเสี่ยงควรจะต้องให้ความสำคัญกับประเด็นทางด้านอคติที่ได้นำเสนอไปด้วย

สำหรับผลของการพยากรณ์ความเสียหายพบว่าระดับความสัมพันธ์และแนวโน้มของการมีอคติกับผลของความเสียหายของทั้งสองกระบวนการนั้น ให้ผลไปในแนวโน้มเดียวกัน กล่าวคือ มีแนวโน้มที่เมื่อมีอคติน้อยหรือมากจนเกินไปจะทำให้เกิดความเสียหายมากกว่าการที่มีอคติในระดับปานกลาง โดยสามารถมองลงไปในรายละเอียดในแต่ละกระบวนการได้ดังนี้

4.9.1 การมีอคติกับผลของความเสียหายของกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์

ลักษณะของการมีอคติกับผลของความเสียหายของกระบวนการนี้มีแนวโน้มที่เมื่อมีอคติในประเภทต่าง ๆ น้อยหรือมากจนเกินไปจะทำให้เกิดความเสียหายมากกว่าการที่มีอคติในประเภทต่าง ๆ ในระดับปานกลาง ซึ่งอาจจะกล่าวได้ดังนี้

สำหรับนักวิเคราะห์ระบบ ผู้ทำงานอยู่ในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์นั้น เป็นผู้ที่มีหน้าที่ประสานงานกับผู้ใช้ภายในนั่นคือทีมพัฒนาและผู้ใช้ภายนอกนั่นคือลูกค้าหรือผู้ใช้งาน เป็นผู้ที่จะต้องสามารถแปลงความต้องการของผู้ใช้งานให้เป็นความต้องการของของระบบได้ ต้องเป็นผู้ที่ต้องทำงานร่วมกับทีมในการพัฒนาและผู้ใช้งานในการยืนยันว่าความต้องการของผู้ใช้ตรงกับสิ่งที่จะพัฒนา และเป็นผู้ที่คอยกำกับดูแลการพัฒนาระบบใหม่ที่ได้มาจากความต้องการนั้น จะเห็นได้ว่าเป็นลักษณะที่จะต้องใช้ทักษะในการสื่อสารและทักษะในการปฏิสัมพันธ์ระหว่างบุคคลอย่างสูง ซึ่งในการทำงานนั้นก็อาจจะมีอคติแต่ละประเภทแฝงอยู่

โดยอคติประเภท Anchoring and adjustment เป็นอคติที่กล่าวว่ามนุษย์มักจะมีการประมาณการหรือคาดการณ์จากประสบการณ์ของตนเองในสิ่งที่ทำก่อนเสมอ และก็มักจะปรับปรุงการประมาณการนั้นให้เหมาะสมกับสถานการณ์ปัจจุบัน

แต่ถ้าหากมีการประมาณการและการปรับให้เข้ากับสถานการณ์ที่น้อยเกินไปก็อาจจะไม่สะท้อนความเป็นจริงที่เป็นอยู่ ก็อาจจะนำมาซึ่งความเสียหายที่อาจจะเกิดขึ้นได้ แต่ในขณะเดียวกันหากมีการประมาณการและปรับแต่งที่มากจนเกินไปก็อาจจะเกินกว่าสิ่งที่จะต้องทำจริง ๆ ซึ่งอาจจะก่อให้เกิดความเสียหายขึ้นมาได้เช่นกัน เช่น ในการหาความต้องการของผู้ใช้งานที่จะแปลงไปเป็นความต้องการของระบบ หากเรามีอคติประเภทนี้น้อย ทำให้เราอาจจะไม่สามารถประมาณความสามารถของระบบเพื่อเสนอแนะแก่ผู้ใช้งานได้ หรือก็ไม่สามารถที่จะแปลงความต้องการนั้นไปเป็นความต้องการของระบบได้ เนื่องจากขาดประสบการณ์ในการจับคู่ความต้องการกับความสามารถของระบบที่ควรจะเป็น และในขณะเดียวกันถ้าหากมีอคติประเภทนี้มากจนเกินไป ก็อาจจะทำให้การแปลงความต้องการเป็นความสามารถของระบบนั้น ให้ระบบมีความสามารถมากเกินความจำเป็น เนื่องด้วยจากการประมาณและการปรับแต่งที่มากเกินไป เป็นต้น ซึ่งทั้งสองเหตุการณ์ก็อาจจะทำให้ข้อกำหนดความต้องการซอฟต์แวร์ที่ได้มานั้น ไม่ตรงกับความต้องการของผู้ใช้งานจริงได้

ในอคติประเภท Availability เป็นอคติที่กล่าวถึงการที่มนุษย์นั้นมีความโน้มเอียงที่จะเชื่อข้อมูลจากสื่อหรือเชื่อเหตุการณ์ที่เพิ่งเกิดขึ้น เหตุการณ์ที่ง่ายต่อการระลึกถึง โดยจะนำเหตุการณ์หรือเรื่องราวนั้นมาใช้กับเหตุการณ์ในปัจจุบัน ซึ่งอาจจะที่จะละเลยถึงความเหมาะสมกับเหตุการณ์ปัจจุบัน

ถ้าหากมีอคติประเภทนี้ นั่นคือ ระลึกถึงสิ่งที่ระลึกขึ้นมาง่าย เช่น สิ่งที่ได้รับฟังจากสื่ออยู่เป็นประจำ ซึ่งถ้านักวิเคราะห์ระบบหรือผู้ปฏิบัติงานในการรวบรวมและกำหนดความต้องการของผู้ใช้มีอคติประเภทนี้ ก็อาจจะทำให้เกิดปัญหาในการรวบรวมความต้องที่ไม่ตรงกับความต้องการที่แท้จริงขึ้นมาได้ เนื่องจากเป็นอคติที่มนุษย์มีความโน้มเอียงที่จะเชื่อข้อมูลที่ระลึกได้ง่าย เช่น เชื่อว่าสถาปัตยกรรมแบบก้อนเมฆ กำลังเป็นที่นิยม ในการพูดคุยกับผู้ใช้งานในระหว่างการรวบรวมและกำหนดความต้องการ ในบางครั้งก็อาจจะมีการนำเสนอข้อมูลเหล่านี้ให้ผู้ใช้ และหากผู้ใช้สนใจก็อาจจะมีแนวความคิดในการบิดความต้องการของตนให้เข้ากับสิ่งที่นำเสนอ เป็นต้น ซึ่งถ้าหากมีอคติประเภทนี้น้อย มีความเป็นไปได้ที่จะทำให้ความต้องการของระบบมีความไม่ทันสมัย เพราะมักจะอ้างกับสิ่งที่รู้มา

นานแล้วที่ไม่ใช่ข้อมูลในปัจจุบัน หรือถ้าหากมีอคติประเภทนี้มากไป ก็มีความเป็นไปได้ที่จะทำให้ความต้องการของระบบบิดเบือนจากความเป็นจริงไปมาก เป็นต้น

ในอคติประเภท Confirmation เป็นอคติที่กล่าวถึงการที่มนุษย์มักจะเชื่อในข้อมูลที่ตรงกับแนวความคิดหรือสนับสนุนแนวคิดของตนมากกว่าที่จะฟังข้อมูลที่หักล้างกับแนวคิดของตน นอกจากนี้ยังพยายามหาข้อมูลเพื่อมาสนับสนุนแนวคิดของตนอีกด้วย เพื่อเสริมความมั่นใจหรือเสริมในการตัดสินใจในเรื่องใดเรื่องหนึ่ง

ตัวอย่างของอคติประเภทนี้ เช่น ในการรวบรวมและกำหนดความต้องการ หลายครั้งผู้ใช้งานอาจจะมีแนวคิดเป็นของตัวเอง เชื่อมั่นในสิ่งที่ตนเองคิดว่าถูกต้องและเชื่อว่ามีความเป็นไปได้ที่จะนำแนวความคิดนั้น ๆ ไปสร้างเป็นความต้องการของระบบได้ ซึ่งความเชื่อดังกล่าวมาจากการที่ผู้ใช้มีข้อมูลที่สนับแนวความคิดของผู้ใช้ว่าสิ่งที่ตนคิดสามารถทำได้ ในขณะเดียวกันชุดความคิดลักษณะนี้ก็สามารถเกิดกับผู้รวบรวมความต้องการเองเช่นกัน สำหรับความเสียหายที่จะเกิดจากการมีอคติประเภทนี้ขึ้นอยู่กับการที่ความเชื่อนั้นถูกต้องหรือไม่ หากถูกต้องและสามารถมีความเป็นไปได้ในการพัฒนาก็อาจจะไม่ก่อให้เกิดความเสียหาย แต่ถ้าหากเชื่อในสิ่งที่ไม่สามารถเป็นไปได้ หรือยากต่อการพัฒนาด้วยเทคโนโลยีในขณะนั้น ก็อาจจะเสียเวลาในพัฒนาต่อไปได้

ดังนั้นจึงจะเห็นได้ว่าการมีอคติทั้งสามประเภทในระดับมากหรือน้อยจนเกินไปก็ก่อให้เกิดความเสียหายมากได้ แต่สำหรับการมีอคติในระดับปานกลาง กล่าวคือมีการทำงานด้วยความเหมาะสมกับสถานการณ์ ซึ่งไม่มากเกินไปหรือน้อยเกินไปก็ช่วยให้ลดความเสี่ยงที่จะเกิดเหตุการณ์ดังที่กล่าวมา

4.9.2 การมีอคติกับผลของความเสียหายของกระบวนการออกแบบซอฟต์แวร์

ลักษณะของการมีอคติกับผลของความเสียหายของกระบวนการนี้มีแนวโน้มที่เมื่อมีอคติในประเภทต่าง ๆ น้อยหรือมากจนเกินไปจะทำให้เกิดความเสียหายมากกว่าการที่มีอคติในประเภทต่าง ๆ ในระดับปานกลาง เช่นเดียวกับกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ แต่การมีอคติประเภท Anchoring and adjustment ที่ระดับปานกลาง เป็นอคติประเภทเดียวที่ไม่ได้ทำให้เกิดความเสียหายของซอฟต์แวร์ที่น้อยที่สุด ที่ซึ่งจะกล่าวโดยละการอธิบายลักษณะของอคติซึ่งได้อธิบายไปแล้วในหัวข้อ 4.9.1 ได้ดังนี้

สำหรับนักออกแบบ หรือผู้ทำงานอยู่ในกระบวนการออกแบบซอฟต์แวร์ เป็นผู้ที่จะต้องสร้างแบบจำลองการทำงานของระบบ ออกมาในรูปแบบของสถานการณ์ กระดานเรื่องราว สถาปัตยกรรมข้อมูล และหน้าจอการทำงาน และยังต้องทำงานร่วมกับฝ่ายบริหาร ฝ่ายวิศวกร ในการออกแบบ มีความเข้าใจในธุรกิจและสามารถเขียนแบบจำลองธุรกิจในการออกแบบการแก้ปัญหาได้ และสามารถสร้างข้อกำหนดของการออกแบบได้ จากส่วนประกอบสำคัญต่าง ๆ ทั้ง ฮาร์ดแวร์ ซอฟต์แวร์ และระบบเครือข่ายของระบบซอฟต์แวร์ เป็นต้น

ในอคติประเภท Anchoring and adjustment หากนักออกแบบหรือผู้ทำงานในกระบวนการออกแบบซอฟต์แวร์มีอคติประเภทนี้มากก็อาจจะทำให้การออกแบบมีแนวความคิดที่จะออกแบบระบบตามแนวความคิดหรือประสบการณ์เดิมของตนเป็นเบื้องต้น โดยอาจจะนำเอาความต้องการในข้อกำหนดความต้องการมาใส่ในภายหลัง ซึ่งนั่นก็อาจจะทำให้โครงของการออกแบบไม่เข้ากับความต้องการจริงเท่าที่ควร เป็นต้น และถ้าหากมีอติประเภทนี้น้อย ก็อาจจะส่งผลให้การออกแบบต้องเริ่มจากความต้องการจากข้อกำหนดความต้องการเพียงอย่างเดียว โดยที่ไม่มีการประมาณถึงการออกแบบระบบที่มีลักษณะสถาปัตยกรรมที่เหมาะสมกับความต้องการไว้ก่อน ซึ่งก็อาจจะออกแบบระบบโดยที่ไม่ได้อ้างอิงหลักสถาปัตยกรรมที่ถูกต้องก็เป็นได้

ในอคติประเภท Availability ถ้านักออกแบบกำลังสนใจเรื่องที่ได้รับจากสื่ออยู่เป็นประจำมาก เช่นเป็นเทคโนโลยีใหม่ที่กำลังมีอิทธิพลอยู่ในปัจจุบัน การที่มีอคติประเภทนี้มากจนเกินไปอาจจะทำให้มีแนวโน้มที่จะออกแบบระบบให้สอดรับกับเทคโนโลยีใหม่ตัวนั้น อาจจะมีสถาปัตยกรรมที่ซับซ้อนมากเกินความจำเป็นของระบบ หรือถ้ามีอคติในเรื่องนี้น้อยจนเกินไป ทำให้ไม่มองถึงด้านความเข้ากันได้ของระบบที่กำลังจะพัฒนากับเทคโนโลยีใหม่ ๆ ในอนาคต หรืออาจจะออกแบบระบบโดยใช้แต่ความรู้และประสบการณ์เก่าที่อาจจะไม่เข้ากับระบบปัจจุบันที่เป็นเทคโนโลยีหรือสถาปัตยกรรมใหม่ เป็นต้น

ในอคติประเภท Confirmation นักออกแบบระบบ หลายครั้งที่อาจมีความไม่แน่นอนของการตัดสินใจในการเลือกวิธีการหรือสถาปัตยกรรมที่จะใช้ในการออกแบบ เนื่องด้วยไม่มีข้อมูลหรือแนวคิดที่สนับสนุนเรื่องดังกล่าว การเลือกวิธีการหรือสถาปัตยกรรมที่จะใช้ในการออกแบบที่ไม่มีการยืนยันนั่นก็อาจจะไม่ใช่วิธีการที่ดีก็ได้ แต่ในขณะเดียวกันถ้าหากมีการยืนยันแน่นอนลงไปในวิธีการหรือสถาปัตยกรรมที่จะใช้ในการออกแบบ หากแต่ว่าวิธีนั้นเป็นวิธีการที่ไม่เหมาะสมกับระบบปัจจุบัน หรือ

การที่เรามีความเชื่อในข้อมูลผิด ๆ การเลือกวิธีการหรือสถาปัตยกรรมนั้นด้วยความมั่นใจแล้วดำเนินการลงไปก็อาจจะก่อให้เกิดความเสียหายได้เช่นกัน

ดังนั้นจึงจะเห็นได้ว่าการมีอคติทั้งสามประเภทในระดับมากหรือน้อยจนเกินไปก็ก่อให้เกิดความเสียหายมากได้ แต่สำหรับการมีอคติในระดับปานกลาง กล่าวคือมีการทำงานด้วยความเหมาะสมกับสถานการณ์ ซึ่งไม่มากเกินไปหรือน้อยเกินไปก็ช่วยให้ลดความเสี่ยงที่จะเกิดเหตุการณ์ดังที่กล่าวมาได้เช่นกันกับกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์

บทที่ 5

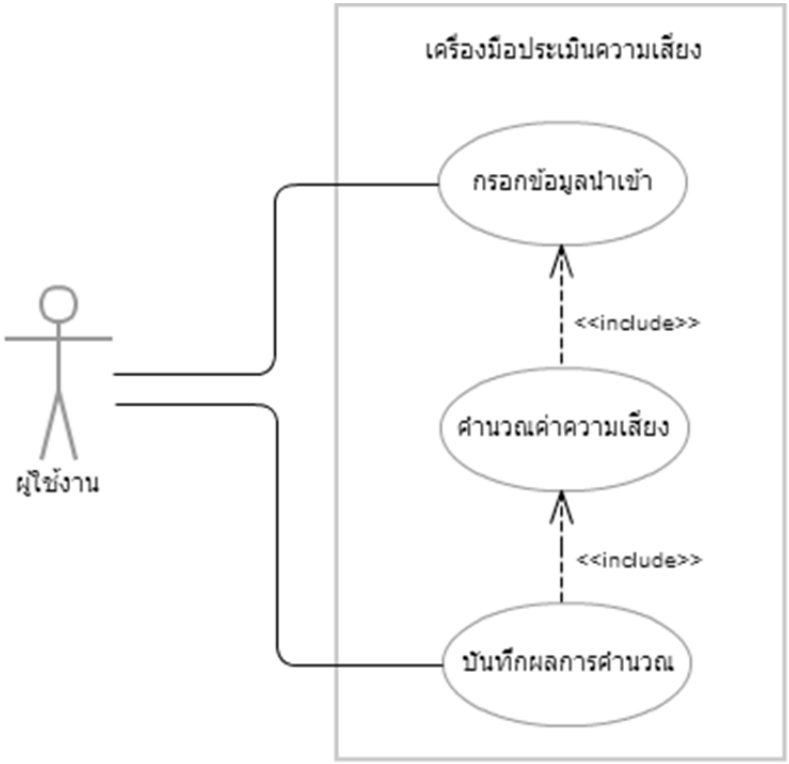
เครื่องมือสนับสนุนแบบจำลองและการทดสอบเครื่องมือ

จากการที่จะได้อธิบายเกี่ยวกับการสร้างแบบจำลองและการสร้างแบบสอบถามไปแล้ว หัวข้อนี้จะอธิบายเกี่ยวกับการสร้างแบบเครื่องมือในการประเมินความเสี่ยงชื่อ “Biskuit” ที่นำเอาแบบจำลองดังกล่าวมาเป็นวิธีการในการคำนวณค่าความเสี่ยง ทั้งนี้ เครื่องมือนี้ยังสามารถนำไปใช้ในการหาความเสี่ยงจากการที่ผู้นำไปใช้ มีการเก็บข้อมูลในลักษณะเดียวกันหรือคล้ายกันกับแบบสอบถามที่ได้เสนอไป กล่าวคือ มีข้อมูลที่แสดงถึงจำนวนของการมีอคติในการทำงานในกระบวนการทั้งสองว่ามีอยู่ในระดับมากน้อยเพียงใด ก็สามารถที่จะประเมินความเสียหายและความเสี่ยงในแต่ละระดับความรุนแรงได้

โดยภาพรวมเครื่องมือที่พัฒนาขึ้นนี้ มีส่วนให้เลือกว่า ต้องการที่จะทราบความเสี่ยงและความเสียหายในกระบวนการใด และกรอกข้อมูลการเกิดการมีอคติในแต่ละประเภทว่าอยู่ในระดับใดของการทำงานโดยอ้างอิงจาก ตารางที่ 3.1 แสดงค่าระดับของจำนวนร้อยละของเหตุการณ์ที่เกิดขึ้น และมีส่วนให้เลือกว่าจะวิเคราะห์เพียงแค่ความเสี่ยงกับค่าความเสียหาย หรือจะวิเคราะห์กับความสูญเสียทางตัวเงินด้วย หากเลือกวิเคราะห์กับความสูญเสียทางด้านตัวเงิน จะมีช่องให้กรอกความสูญเสียทางด้านตัวเงินต่อความเสียหายในแต่ละระดับความรุนแรง ซึ่งแบ่งออกเป็น 5 ระดับ ซึ่งจำนวนเงินนี้ขึ้นอยู่กับแต่ละองค์กรว่าจะประเมินจำนวนเงินในแต่ละระดับอย่างไร โดยที่จำนวนเงินดังกล่าวเป็นความสูญเสียที่เกิดขึ้นต่อความเสียหายหนึ่งครั้ง

5.1 แผนภาพยูสเคส

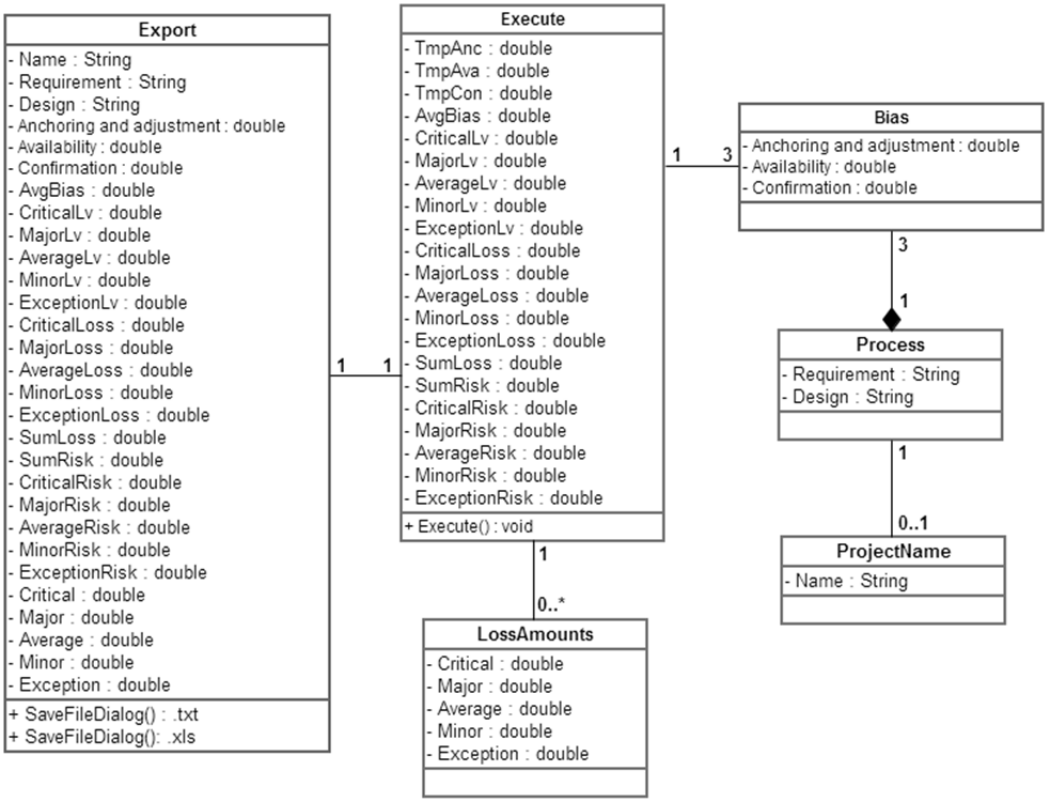
สำหรับเครื่องมือนี้มีแผนภาพยูสเคสแสดงปฏิสัมพันธ์ระหว่างผู้ใช้งานกับเครื่องมือ ดังภาพที่ 5.1



ภาพที่ 5.1 แผนภาพยูสเคสของเครื่องมือสำหรับการประเมินความเสี่ยง

5.2 แผนภาพคลาส

สำหรับเครื่องมือนี้มีแผนภาพคลาสแสดงความสัมพันธ์ของคลาสและแสดงถึงโครงสร้างของข้อมูลที่ใช้ในเครื่องมือ ดังภาพที่ 5.2



ภาพที่ 5.2 แผนภาพคลาสของเครื่องมือสำหรับการประเมินความเสี่ยง

จากภาพที่ 5.2 แผนภาพคลาสประกอบไปด้วยรายละเอียดในแต่ละคลาสดังนี้

1) Export สำหรับการบันทึกข้อมูลผลการคำนวณออกมาในสองรูปแบบคือไฟล์ .txt และไฟล์ .xls
2) Execute สำหรับการคำนวณค่าประมาณความเสี่ยง ค่าประมาณความเสียหายและมูลค่าของความเสียหาย
3) Bias สำหรับกำหนดค่าสัมประสิทธิ์ความถดถอยเพื่อใช้ในการคำนวณระดับของการมีอคติกับความเสียหาย
4) ProjectName สำหรับระบุชื่อโครงการเพื่อใช้ในการนำไปแสดงผล
5) Process สำหรับระบุกระบวนการที่สนใจเพื่อใช้ในการเลือกวิธีการคำนวณ
6) LossAmounts สำหรับการใส่มูลค่าของความเสียหายในแต่ละระดับความรุนแรง

5.3 สมการในการคำนวณ

เครื่องมือนี้มีการทำงานภายในด้วยแบบจำลองการประเมินความเสี่ยงซึ่งมีสมการทั้งหมดเรียงตามลำดับการใช้งานโดยเครื่องมือนี้มีดังนี้

ตารางที่ 5.1 สมการทางคณิตศาสตร์ทั้งหมดที่ใช้ในการสร้างเครื่องมือสนับสนุนแบบจำลอง

สูตร	ค่านำเข้า	ค่านำออก	วัตถุประสงค์
$\hat{Y}_{anc} = a + b_2 ancx_2 + b_3 ancx_3 + b_4 ancx_4 + b_5 ancx_5$	$ancx_2$ $ancx_3$ $ancx_4$ $ancx_5$	$\hat{Y}_{anc}$	เพื่อหาค่าประมาณความเสียหายของอคติประเภท Anchoring and adjustment โดยรับค่านำเข้ามาจากการเลือกระดับของการมีอคติซึ่งในแต่ละระดับจะถูกจับคู่กับค่านำเข้า หากเลือกที่ระดับใด ค่านำเข้าที่ระดับนั้นจะมีค่าเท่ากับ 1
$\hat{Y}_{ava} = a + b_2 avax_2 + b_3 avax_3 + b_4 avax_4 + b_5 avax_5$	$avax_2$ $avax_3$ $avax_4$ $avax_5$	$\hat{Y}_{ava}$	เพื่อหาค่าประมาณความเสียหายของอคติประเภท Availability โดยรับค่านำเข้ามาจากการเลือกระดับของการมีอคติซึ่งในแต่ละระดับจะถูกจับคู่กับค่านำเข้า หากเลือกที่ระดับใด ค่านำเข้าที่ระดับนั้นจะมีค่าเท่ากับ 1

ตารางที่ 5.1 สมการทางคณิตศาสตร์ทั้งหมดที่ใช้ในการสร้างเครื่องมือสนับสนุนแบบจำลอง (ต่อ)

สูตร	ค่านำเข้า	ค่านำออก	วัตถุประสงค์
$\hat{Y}_{con} = a + b_2 conx_2 + b_3 conx_3 + b_4 conx_4 + b_5 conx_5$	$conx_2$ $conx_3$ $conx_4$ $conx_5$	$\hat{Y}_{con}$	เพื่อหาค่าประมาณความเสียหายของอคติประเภท Confirmation โดยรับค่านำเข้ามาจากการเลือกระดับของการมีอคติซึ่งในแต่ละระดับจะถูกจับคู่กับค่านำเข้า หากเลือกที่ระดับใด ค่านำเข้าที่ระดับนั้นจะมีค่าเท่ากับ 1
$\bar{Y} = \frac{\hat{Y}_{anc} + \hat{Y}_{ava} + \hat{Y}_{con}}{3}$	$\hat{Y}_{anc}$ $\hat{Y}_{ava}$ $\hat{Y}_{con}$	$\bar{Y}$	หาอัตราเสียหายโดยเฉลี่ยจากอคติทั้งสามประเภท

ตารางที่ 5.1 สมการทางคณิตศาสตร์ทั้งหมดที่ใช้ในการสร้างเครื่องมือสนับสนุนแบบจำลอง (ต่อ)

สูตร	ค่านำเข้า	ค่านำออก	วัตถุประสงค์
$R_i = \frac{k_i \times \bar{Y}}{100}$	$\bar{Y}$ k_i	R_i	ต้องการที่จะได้ค่าความเสียหายหรือ R_i i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception และมี k_i เป็นค่าคงที่โดย i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception โดยสำหรับกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์มีค่าคงที่เรียงตามลำดับดังนี้ 3.19, 7.45, 28.72, 30.85, 29.79 ตามลำดับ และสำหรับกระบวนการออกแบบซอฟต์แวร์มีค่าคงที่เรียงตามลำดับดังนี้ 2.25, 13.48, 41.57, 35.96, 6.74
$P_i = 1 - e^{-m_i}$	e m_i	P_i	เพื่อต้องการหาค่าความน่าจะเป็นจากสมการปัวซองที่มีความน่าจะเป็นที่จะเกิดความเสียหายอย่างน้อย 1 ครั้ง โดยที่ e เป็นค่าคงที่ประมาณ 2.71828 และ m = อัตราความเสียหายโดยเฉลี่ยจากแบบสอบถามในช่วงเวลาหนึ่งเดือน ที่มี i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception โดยสำหรับกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์มีค่าคงที่เรียงตามลำดับดังนี้ 0.10, 0.23, 0.87, 0.94, 0.90 ตามลำดับ และสำหรับกระบวนการออกแบบซอฟต์แวร์มีค่าคงที่เรียงตามลำดับดังนี้ 0.06, 0.39, 1.19, 1.03, 0.19

ตารางที่ 5.1 สมการทางคณิตศาสตร์ทั้งหมดที่ใช้ในการสร้างเครื่องมือสนับสนุนแบบจำลอง (ต่อ)

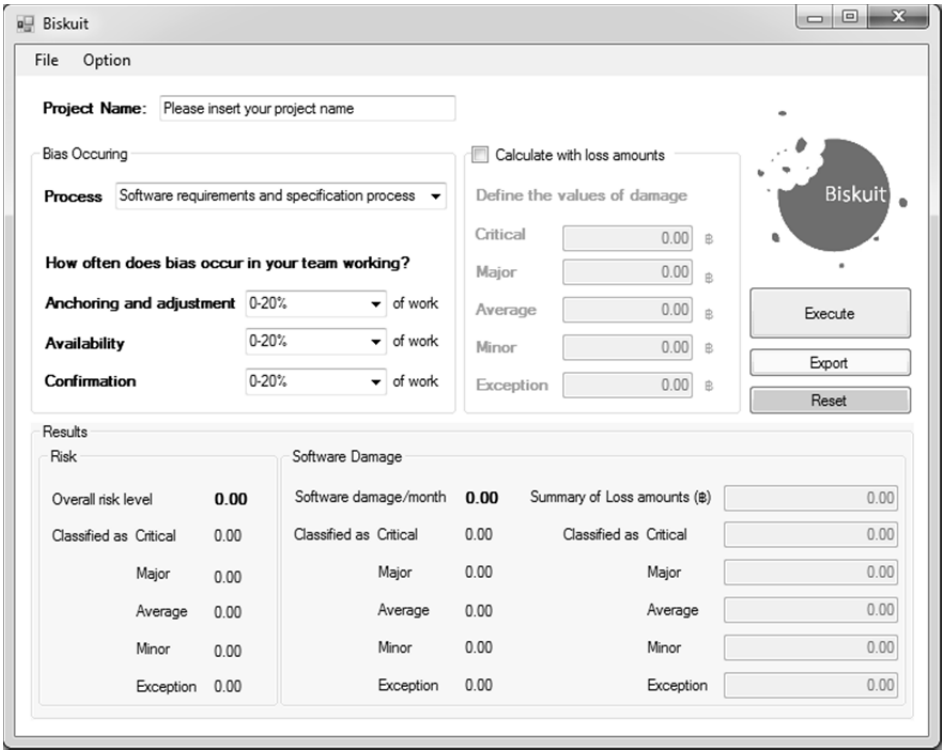
สูตร	ค่านำเข้า	ค่านำออก	วัตถุประสงค์
$Risk_i = R_i \times P_i$	R_i P_i	$Risk_i$	ต้องการหาค่าความเสี่ยงจากความรุนแรงในแต่ละระดับ i = 5, 4, 3, 2, 1 จากความรุนแรงของความเสียหาย 5 ระดับ ซึ่งกำหนดให้ 5 = Critical, 4 = Major, 3 = Average, 2 = Minor, 1 = Exception โดยรับค่าความเสียหายที่ได้จากการคำนวณคูณด้วยค่าความน่าจะเป็นที่ได้จากการคำนวณ
$pRisk = \sum_{i=1}^{5} Risk_i$	$Risk_i$	pRisk	ต้องการหาค่าความเสี่ยงรวมจาก 5 ระดับความรุนแรงของอคติแต่ละประเภท โดยที่ p = Re, Ds นั่นคือกระบวนการที่สนใจ
$Loss\ amounts\ per\ month = LA_i \times R_i$	LA_i R_i	Loss amounts per month	คำนวณความสูญเสียทางด้านตัวเงินจากความเสียหายในแต่ละระดับความรุนแรง โดยที่ LA_i = ค่า Loss amounts หรือมูลค่าความเสียหายทางด้านตัวเงินต่อความเสียหายหนึ่งครั้ง และ R_i = ค่าความเสียหายที่ได้จากการประมาณและจำแนกออกเป็นแต่ละระดับความรุนแรง
$Summary\ of\ loss\ amounts\ per\ month = \sum_{i=1}^{5} LA_i \times \bar{Y}$	LA_i $\bar{Y}$	Summary of Loss amounts per month	คำนวณความสูญเสียทางด้านตัวเงินจากความเสียหายในทุกระดับความรุนแรง โดยที่ $\bar{Y}$ = ค่าความเสียหายรวมที่ได้จากการประมาณ

5.4 เครื่องมือที่ใช้ในการพัฒนา

เครื่องมือที่นำมาใช้ในการพัฒนาโปรแกรม คือ ไมโครซอฟท์วิชวลสตูดิโอ 2010 (Microsoft Visual Studio 2010) โดยทำงานอยู่บนระบบปฏิบัติการไมโครซอฟท์วินโดวส์เซเว่นอัลติเมท (Microsoft Windows 7 Ultimate)

5.5 ส่วนต่อประสานกับผู้ใช้งาน

ส่วนต่อประสานกับผู้ใช้งาน (User Interface) ของโปรแกรม “Biskuit” แบ่งออกเป็นสองส่วนหลัก โดยส่วนหลักที่หนึ่ง อยู่ทางครึ่งบนของโปรแกรม เป็นส่วนที่รับค่านำเข้า (Input) และส่วนปุ่มกดสำหรับการดำเนินการ และส่วนหลักที่สอง อยู่ทางครึ่งล่างของโปรแกรม เป็นส่วนที่แสดงผลจากการคำนวณภายหลังกดปุ่ม Execute ในส่วนหลักที่หนึ่ง ดังภาพที่ 5.3



ภาพที่ 5.3 หน้าจอเริ่มต้นของโปรแกรม Biskuit

โดยที่ในส่วนหลักที่หนึ่ง ประกอบด้วยห้าส่วนย่อย ดังนี้

1) ส่วนย่อย Project Name สำหรับการรับข้อมูลชื่อโครงการ
2) ส่วนย่อย Bias Occuring สำหรับการรับข้อมูลการมีอคติแต่ละประเภท ในแต่ละระดับ และในแต่ละกระบวนการ
3) ส่วนย่อย Calculate with loss amounts สำหรับการรับข้อมูลมูลค่าความเสียหายในการเกิดความเสียหายในแต่ละระดับ
4) ส่วนปุ่มกดสำหรับการดำเนินการ จำนวน 3 ปุ่ม คือ ปุ่ม Execute สำหรับกดเพื่อทำการคำนวณค่าอัตราความเสียหายและค่าความเสี่ยง ปุ่ม Export สำหรับการส่งออกข้อมูลที่ได้จากการคำนวณ และปุ่ม Reset สำหรับล้างการกรอกข้อมูลไปเป็นค่าเริ่มต้น
5) แถบเมนูด้านบนสุด ประกอบด้วยเมนู File มีเมนูย่อยสำหรับ Export สำหรับการส่งออกข้อมูลที่ได้จากการคำนวณและ Exit เพื่อออกจากโปรแกรม และเมนู Option มีเมนูย่อยสำหรับ Execute และ Reset เช่นเดียวกับข้อก่อนหน้า

และโดยที่ในส่วนหลักที่สอง ประกอบด้วยสองส่วนย่อย ดังนี้

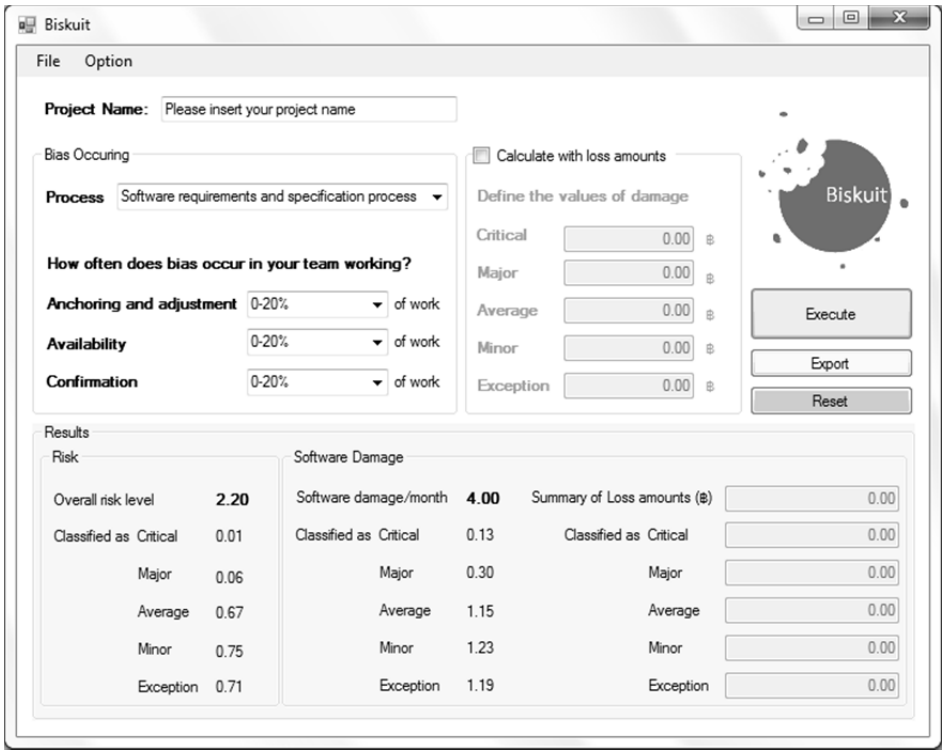
1) ส่วนย่อย Risk สำหรับแสดงผลการคำนวณค่าความเสี่ยงรวมและค่าความเสี่ยงในแต่ละระดับความรุนแรง จากการมีอคติในแต่ละประเภท ในแต่ละระดับ และในแต่ละกระบวนการ
2) ส่วนย่อย Software Damage สำหรับแสดงผลการคำนวณค่าอัตราความเสียหายรวมและค่าอัตราความเสียหายในแต่ละระดับความรุนแรง จากการมีอคติในแต่ละประเภท ในแต่ละระดับ และในแต่ละกระบวนการ โดยในส่วนนี้ยังแสดงมูลค่าความเสียหายจากการเกิดความเสียหายขึ้นด้วย

สำหรับลำดับการทำงานของโปรแกรม “Biskuit” เป็นไปตามลำดับดังนี้

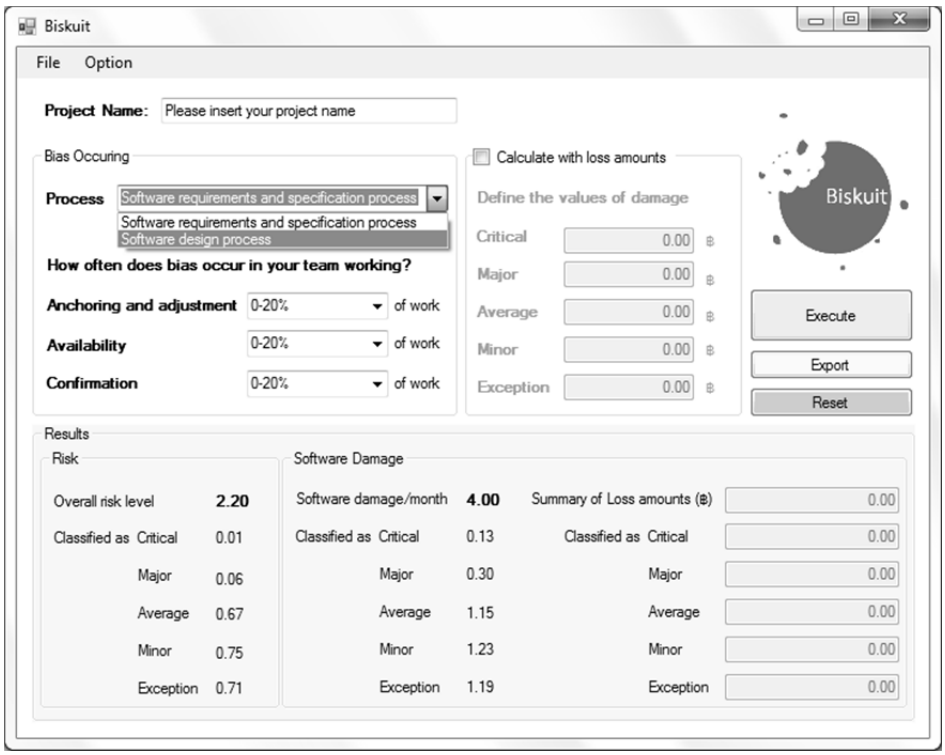
1) การเปิดโปรแกรมขึ้นมาจะพบส่วนสำหรับกรอกข้อมูล โดยมีการตั้งค่าเริ่มต้นเป็นการมีอคติทุกประเภทในการทำงานในกระบวนการรวบรวมและกำหนดความต้องการอยู่ในระดับ 0-20%

2) การกดปุ่ม Execute ในครั้งแรกภายหลังเปิดโปรแกรมขึ้นมา โปรแกรมจะคำนวณผลจากค่าเริ่มต้นที่ถูกตั้งเอาไว้ แล้วแสดงผลในส่วนของ Result
3) สามารถเลือกกระบวนการที่สนใจที่ ComboBox ในแถบ Process
4) สามารถกำหนดการมีอคติในแต่ละระดับของอคติแต่ละประเภทได้จากการเลือกที่ ComboBox ในแถบของอคติแต่ละประเภท
5) สามารถเลือกการคำนวณมูลค่าความเสียหายได้จากการทำเครื่องหมายถูกในช่อง CheckBox หน้าคำว่า Calculate with loss amounts
6) สามารถกรอกมูลค่าความเสียหายเป็นจำนวนเงินได้ใน TextBox ภายหลังจากการทำเครื่องหมายถูกในช่อง CheckBox หน้าคำว่า Calculate with loss amounts
7) การกดปุ่ม Execute ภายหลังจากเลือกการคำนวณมูลค่าความเสียหายเป็นตัวเงิน โปรแกรมจะคำนวณผลจากค่าที่ถูกเลือกเอาไว้ แล้วแสดงผลในส่วนของ Result
8) สามารถกดปุ่ม Export จากหน้าโปรแกรมหรือในแถบเมนู เพื่อนำผลที่ได้จากการคำนวณมาอยู่ในรูปของ .txt และ .xls
9) สามารถทำการบันทึกในรูปแบบใดรูปแบบหนึ่ง จากนั้นโปรแกรมจะแสดงผลการบันทึกในทันทีตามรูปแบบไฟล์ที่ได้เลือกไว้
10) สามารถกดปุ่ม Reset จากหน้าโปรแกรมหรือในแถบเมนู เพื่อล้างการกรอกข้อมูลให้เป็นค่าเริ่มต้น

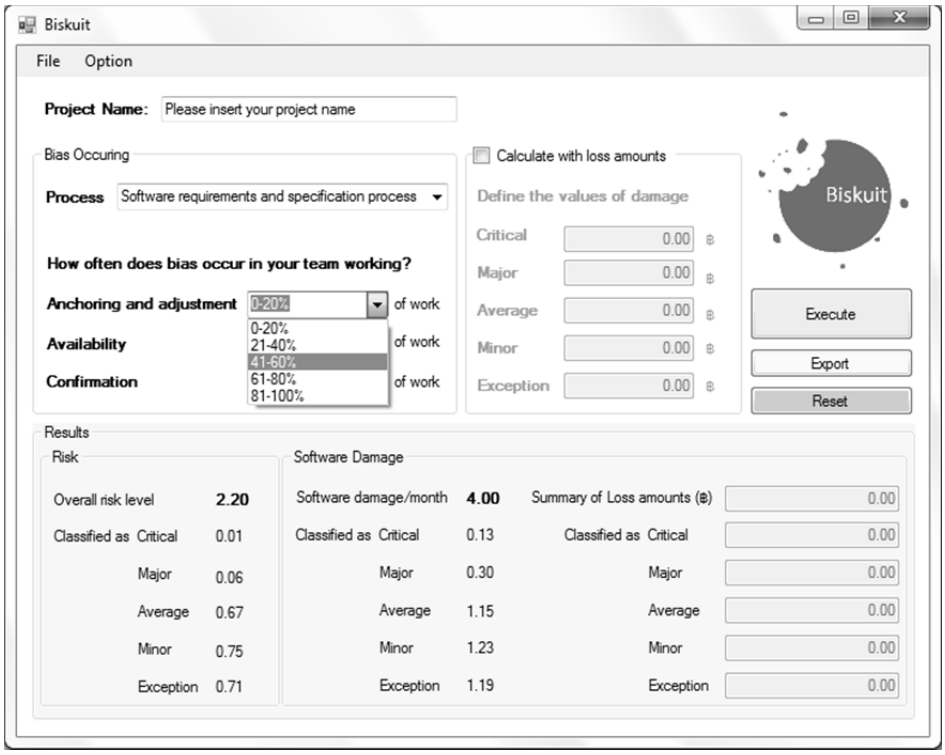
โดยลำดับการทำงานสามารถแสดงได้ดังภาพที่ 5.4 ถึง 5.13



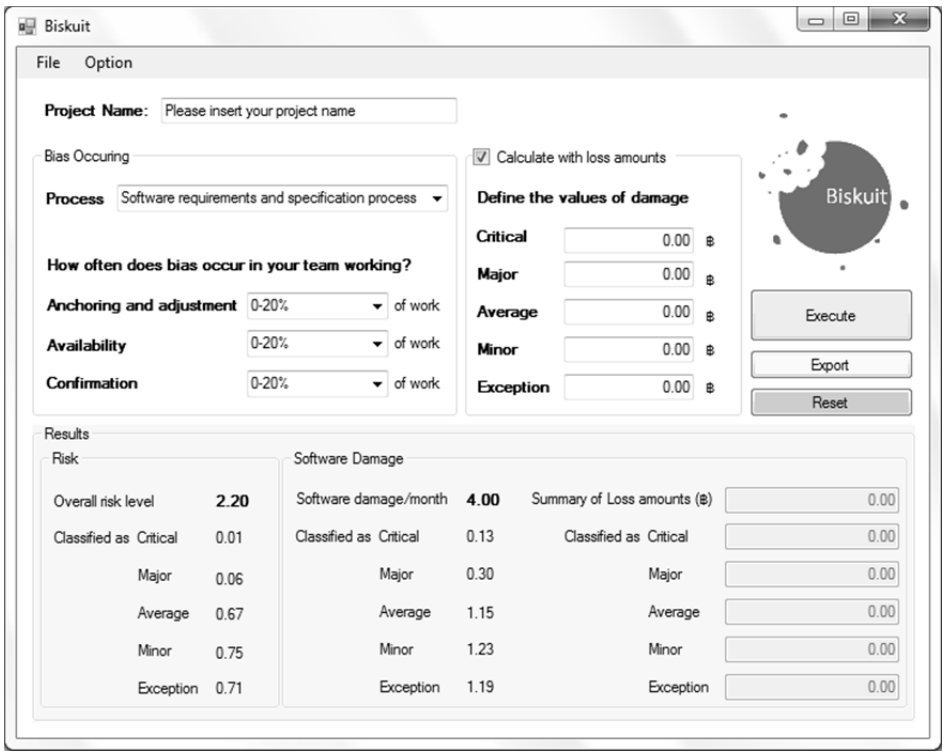
ภาพที่ 5.4 แสดงผลของการกดปุ่ม Execute ในครั้งแรกภายหลังเปิดโปรแกรมขึ้นมา



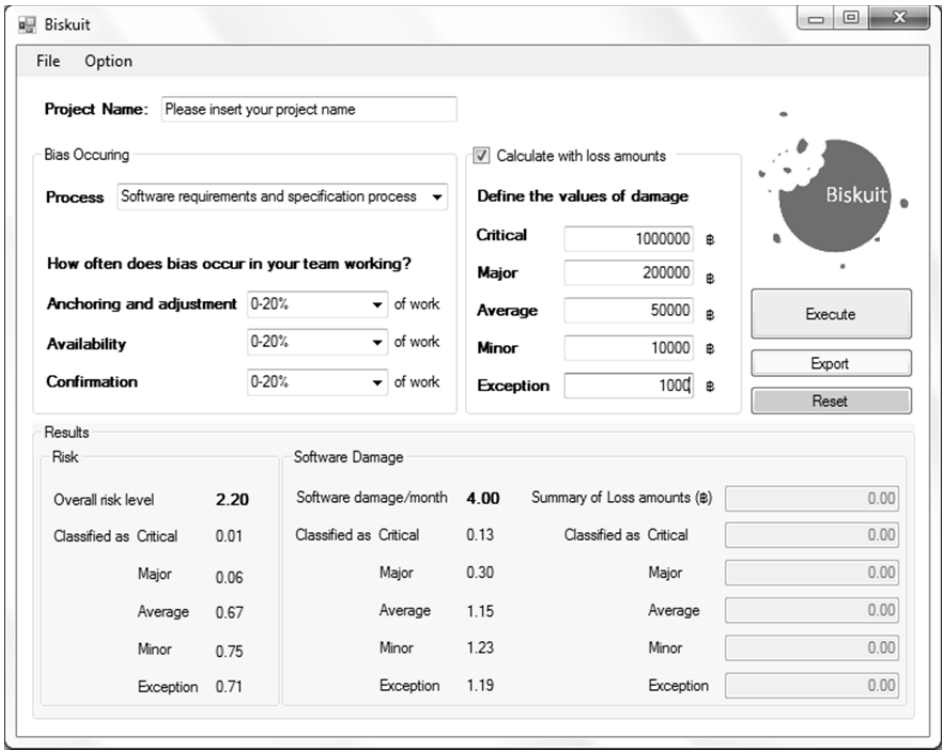
ภาพที่ 5.5 แสดงการเลือกกระบวนการที่ต้องการคำนวณใน Combo Box



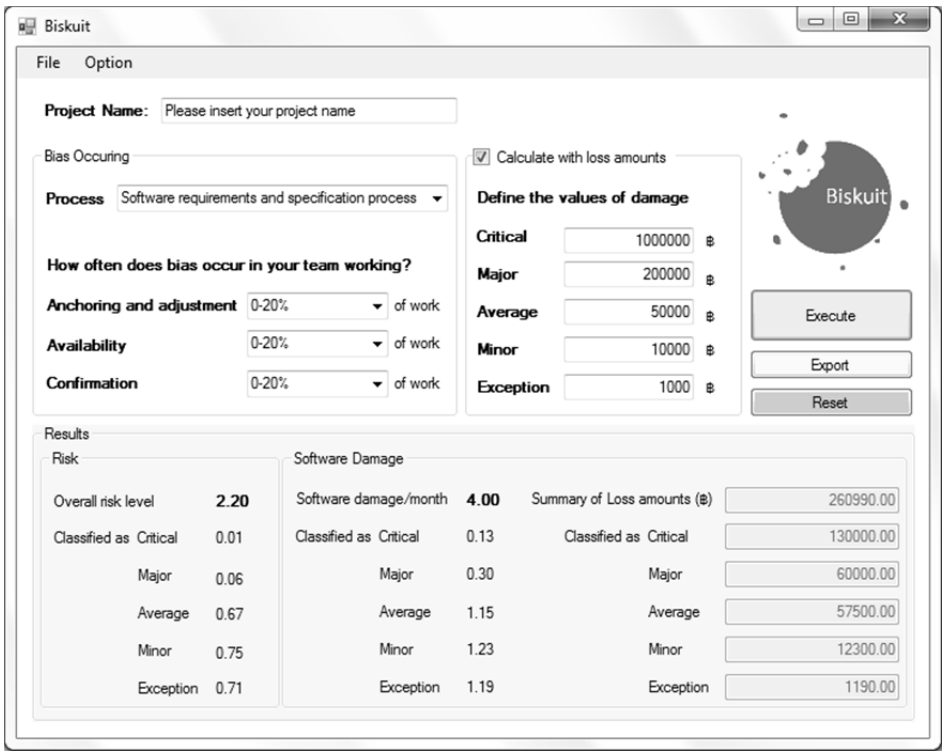
ภาพที่ 5.6 แสดงการเลือกระดับของการมีอคติในการทำงานที่ต้องการคำนวณใน Combo Box



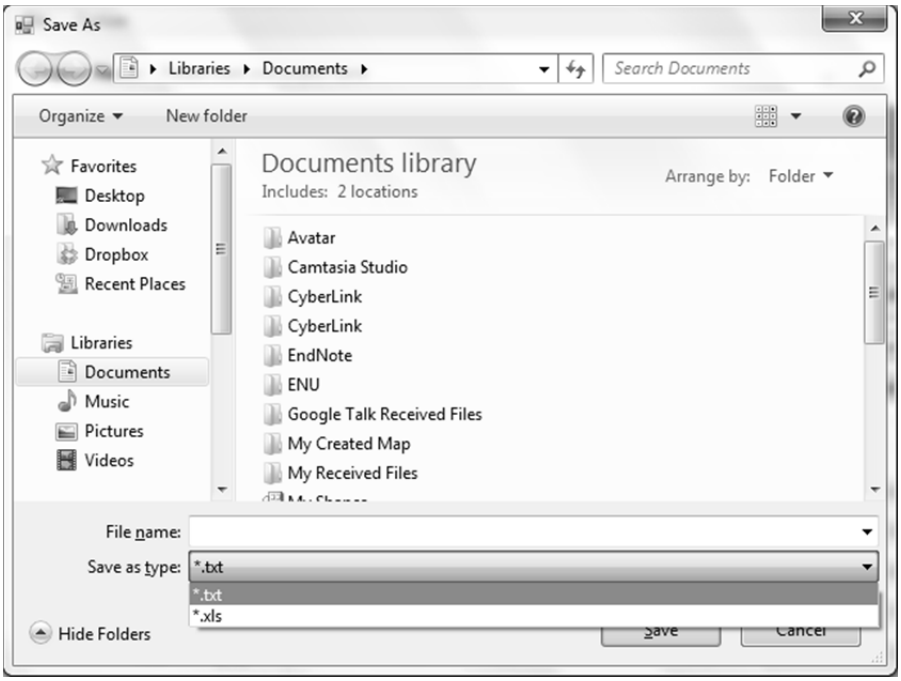
ภาพที่ 5.7 แสดงการเลือกรูปแบบการคำนวณโดยคิดมูลค่าความเสียหายที่เป็นตัวเงิน



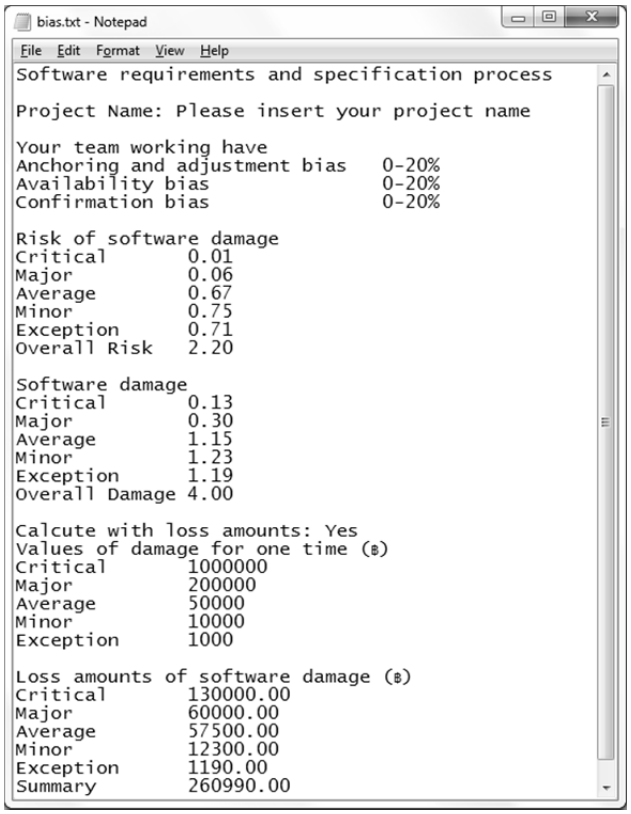
ภาพที่ 5.8 การกรอกมูลค่าความเสียหายที่เป็นตัวเงินเพื่อที่จะนำใช้ในการคำนวณ



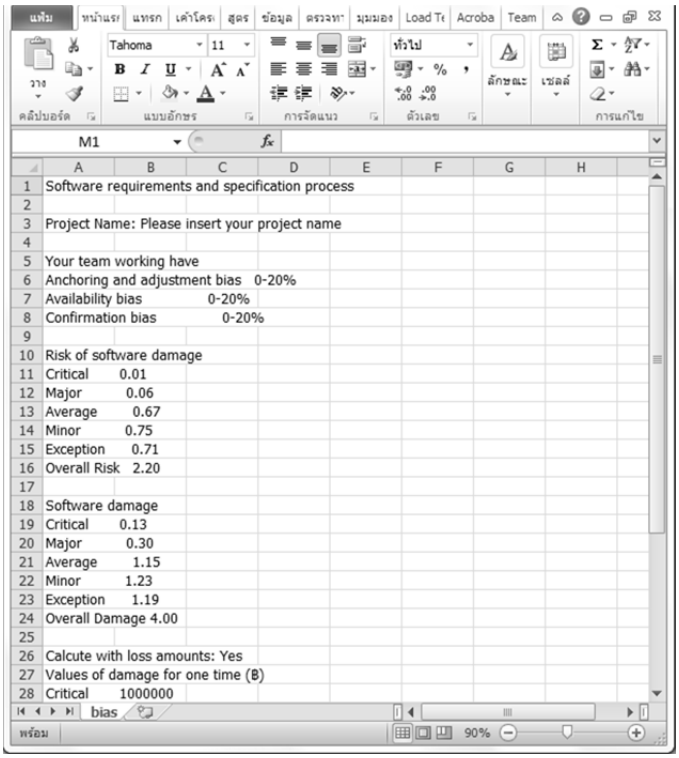
ภาพที่ 5.9 แสดงผลการกดปุ่ม Execute ภายหลังจากเลือกการคำนวณมูลค่าความเสียหายเป็นตัวเงิน



ภาพที่ 5.10 แสดงผลการกดปุ่ม Export เพื่อนำผลที่ได้จากการคำนวณมาอยู่ในรูปของ .txt และ .xls

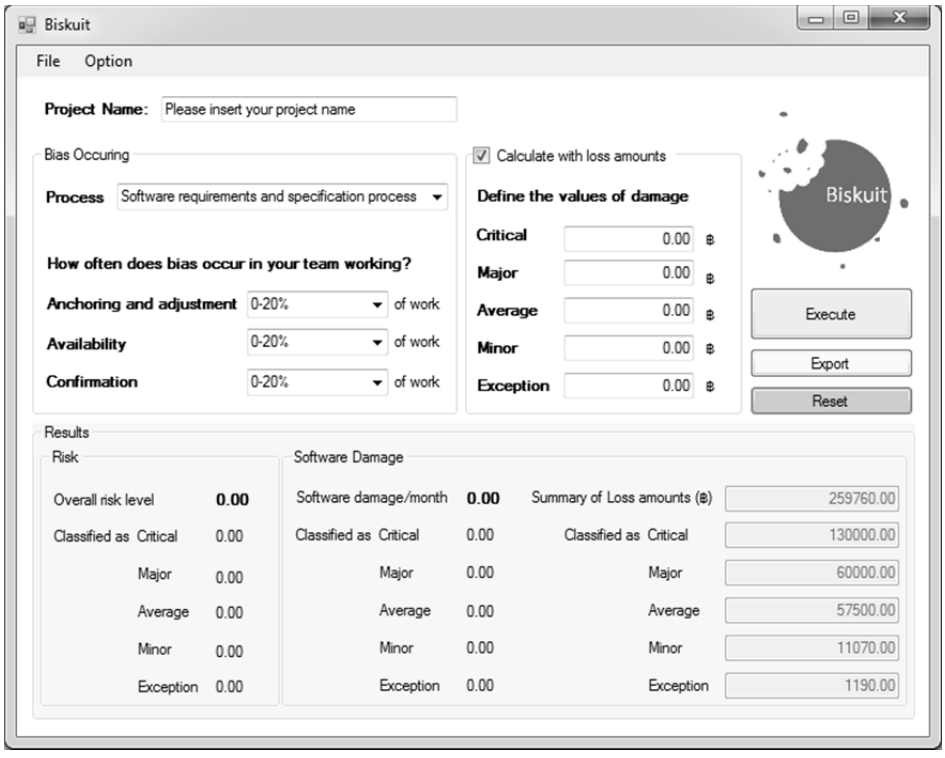


ภาพที่ 5.11 แสดงผลการ Export เพื่อนำผลที่ได้จากการคำนวณมาอยู่ในรูปของ .txt



	A	B	C	D	E	F	G	H
1	Software requirements and specification process							
2								
3	Project Name: Please insert your project name							
4								
5	Your team working have							
6	Anchoring and adjustment bias	0-20%						
7	Availability bias	0-20%						
8	Confirmation bias	0-20%						
9								
10	Risk of software damage							
11	Critical	0.01						
12	Major	0.06						
13	Average	0.67						
14	Minor	0.75						
15	Exception	0.71						
16	Overall Risk	2.20						
17								
18	Software damage							
19	Critical	0.13						
20	Major	0.30						
21	Average	1.15						
22	Minor	1.23						
23	Exception	1.19						
24	Overall Damage	4.00						
25								
26	Calcute with loss amounts: Yes							
27	Values of damage for one time (B)							
28	Critical	1000000						

ภาพที่ 5.12 แสดงผลการ Export เพื่อนำผลที่ได้จากการคำนวณมาอยู่ในรูปของ .xls



ภาพที่ 5.13 แสดงผลการกดปุ่ม Reset

สำหรับในแถบเมนูด้านบน (Menu Bar) ในส่วนของ File ประกอบไปด้วย Dropdown list ที่มีการทำงานเหมือนปุ่ม Export นั่นคือตัวเลือก “Export” และตัวเลือก “Exit” จะทำการปิดโปรแกรม และในส่วนของ Option ประกอบไปด้วย Dropdown list ที่มีการทำงานเหมือนปุ่ม Execute และปุ่ม Reset นั่นคือตัวเลือก “Execute” และตัวเลือก “Reset” ตามลำดับ

5.6 การทดสอบเครื่องมือ

การทดสอบเครื่องมือสนับสุนแบบจำลองการประเมินความเสี่ยงนี้ ทำการทดสอบความถูกต้องในการแสดงผลและการทำงานของเครื่องมือ ดังตารางที่ 5.2

ตารางที่ 5.2 การทดสอบการแสดงผลและการทำงานของเครื่องมือ

หมายเลขกรณีทดสอบ:	TC01	**ชื่อกรณีทดสอบ:**	การแสดงผลและการทำงานของเครื่องมือ
ระบบ:	Biskuit	**ระบบย่อย:**	-
ผู้จัดทำ:	นพธนิษฐ์ โชติสาร	**วันที่จัดทำ:**	18-เม.ย.-56
ผู้ทำการทดสอบ:	นพธนิษฐ์ โชติสาร	**วันที่ทำการทดสอบ:**	19-เม.ย.-56
คำอธิบาย:	ทดสอบการแสดงผลและขั้นตอนการทำงานของเครื่องมือ		
สถานะก่อนการทดสอบ:			
อยู่ที่หน้าแรกหลังจากเปิดโปรแกรม Biskuit ขึ้นมาใช้งาน			

ตารางที่ 5.2 การทดสอบการแสดงผลและการทำงานของเครื่องมือ (ต่อ)

ขั้นตอน	การทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ	หมายเหตุ
1	ระบุกระบวนการที่สนใจ	โปรแกรมแสดงกระบวนการที่สนใจได้	Pass	
2	ระบุค่าการมือคติในแต่ละกระบวนการ	โปรแกรมแสดงการเลือกระดับการมือคติในแต่ละกระบวนการได้	Pass	
3	ระบุชื่อโครงการ	สามารถใส่ชื่โครงการได้	Pass	
4	ทำตามขั้นตอนที่ 1, 2, 3 แล้วกดปุ่ม Execute	สามารถกดปุ่ม Execute และแสดงผลในส่วน Result ได้	Pass	
5	เลือก Checkbox ในส่วนของ Calculate with loss amounts	สามารถเลือกได้ และสามารถกรอกจำนวนเงินในทั้ง 5 ช่องได้	Pass	
6	ทำตามขั้นตอนที่ 1, 2, 3, 5 แล้วกดปุ่ม Execute	สามารถกดปุ่ม Execute และแสดงผลในส่วน Result ได้ โดยมีจำนวนเงินปรากฏในส่วน Result ได้	Pass	
7	ทำซ้ำในขั้นตอนที่ 4 และ 6 ตามลำดับในอีกกระบวนการหนึ่ง	แสดงผลเช่นเดียวกับขั้นตอนที่ 4 และ 6	Pass	

ตารางที่ 5.2 การทดสอบการแสดงผลและการทำงานของเครื่องมือ (ต่อ)

ขั้นตอน	การทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ	หมายเหตุ
8	กดปุ่ม Export	แสดงหน้าจอให้ทำการเซฟไฟล์เป็นใน 2 รูปแบบคือ .txt และ .xls	Pass	
9	กรอกชื่อและนามสกุลไฟล์แล้วกด Save	โปรแกรมแสดงผลของการคำนวณที่เพิ่งทำการบันทึกไปตามแต่นามสกุลไฟล์ที่เลือกไว้	Pass	
10	ทำซ้ำขั้นตอนที่ 8 และ 9 แต่ไปทำในแถบเมนูด้านบน	แสดงผลเช่นเดียวกับขั้นตอนที่ 8 และ 9	Pass	
11	ทำซ้ำขั้นตอนที่ 7 แต่ไปทำในแถบเมนูด้านบน	แสดงผลเช่นเดียวกับขั้นตอนที่ 7	Pass	
12	กดปุ่ม Reset	ข้อมูลทุกส่วนกลับสู้สถานะตั้งต้น ยกเว้น Project Name	Pass	
13	ทำซ้ำขั้นตอนที่ 12 แต่ไปทำในแถบเมนูด้านบน	แสดงผลเช่นเดี่ยวกับขั้นตอนที่ 12	Pass	
14	กดปุ่ม Export ทั้งด้านนอกและในแถวเมนู ขณะที่ข้อมูลอยู่ในสถานะเริ่มต้น	โปรแกรมแสดงกล่องข้อความว่าให้กดปุ่ม Execute ก่อนจึงจะสามารถกดปุ่ม Export ได้	Pass	
15	เลือก File > Exit	โปรแกรมปิดตัวลง	Pass	

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

จากการทดลองและผลการทดลองสำหรับคำนวณหาค่าความเสี่ยงของอัตราความเสียหายของซอฟต์แวร์จากการมีอคติทางการรู้คิดแล้วสามารถสรุปผลการวิจัย ระบุข้อจำกัดของการใช้แบบจำลองและเครื่องมือที่สนับสนุนแบบจำลองนั้น และให้ข้อเสนอแนะในการวิจัยและการใช้แบบจำลองดังต่อไปนี้

6.1 สรุปผลการวิจัย

จากการที่มีคำถามเกี่ยวกับปัจจัยอื่น ๆ ที่อาจจะส่งผลกระทบต่อโครงการพัฒนาซอฟต์แวร์โดยสนใจในมนุษยปัจจัย ด้านการมีอคติทางการรู้คิด และได้รวบรวมข้อมูลจากกลุ่มตัวอย่างกลุ่มที่ใช้สร้างลักษณะเหตุการณ์จากคำสำคัญที่แสดงถึงลักษณะของการมีอคติที่ได้กำหนดไว้ เพื่อที่จะนำลักษณะเหตุการณ์ดังกล่าวไปใช้ในการศึกษาการมีอคติประเภทต่าง ๆ ในแต่ละระดับ ว่ามีผลต่อความเสียหายของซอฟต์แวร์อย่างไร

โดยจะสำรวจจากกลุ่มตัวอย่างที่เป็นผู้ปฏิบัติงานในโครงการซอฟต์แวร์อีกกลุ่ม และนำผลที่ได้มาสร้างแบบจำลองการประเมินความเสี่ยงตามขั้นตอนที่ได้แสดงไว้ในบทที่ 3 ซึ่งผลจากแบบสอบถาม จำเป็นที่จะต้องแปลงค่าตัวแปรอิสระ นั่นคือการมีอคติประเภทต่าง ๆ ในแต่ละระดับ ซึ่งเป็นตัวแปรเชิงคุณภาพ ไปเป็นตัวแปรหุ่น เพื่อที่จะสามารถทำการทดลองด้วยวิธีการวิเคราะห์ความถดถอยกับตัวแปรตามที่เป็นตัวแปรเชิงปริมาณต่อไปได้

จากการทดลองและผลการทดลองในบทที่ 4 พบว่าแบบจำลองที่สร้างขึ้นใช้ในการพยากรณ์ความเสียหายของซอฟต์แวร์จากการมีอคติทางการรู้คิดในแต่ละระดับ (น้อยมาก น้อย ปานกลาง มาก มากที่สุด) ของทั้งสามประเภทนั่นคือ Anchoring and adjustment, Availability, และ Confirmation นั้น หากสนใจเพียงแค่ต้องการค่าสัมประสิทธิ์ที่ได้จากการวิเคราะห์ความถดถอยเพื่อมาสร้างสมการพยากรณ์ โดยอนุโลม ที่ไม่สามารถพยากรณ์ได้ในระดับนัยสำคัญทางสถิติที่ระดับ .05 ก็จะสามารถสร้างสมการพยากรณ์ในการพยากรณ์ของอคติแต่ละประเภทในการมีอคติในแต่ละระดับได้ ซึ่งจากการวิเคราะห์ทางสถิติ พบว่าสามารถพยากรณ์ความเสียหายได้ในระดับที่ต่ำถึงค่อนข้างต่ำ ซึ่งถ้า

หากนำค่าความเสียหายที่ได้ไปคำนวณกับค่าความน่าจะเป็นที่ได้จากวิธีการแบบปัวซองในสมการความเสี่ยงแล้ว ก็จะสามารถประมาณค่าความเสี่ยงออกมาได้ แต่ด้วยความสามารถในการพยากรณ์มีต่ำจึงทำให้ค่าประมาณความเสี่ยงมีความสามารถในการณ์พยากรณ์ต่ำด้วย

ถึงแม้ว่าจะไม่สามารถพยากรณ์ออกมาได้แม่นยำถึงระดับมากหรือมากที่สุด แต่ก็ใช่ว่าจะไม่มีความเสียหายที่เกิดจากประเด็นทางด้านการมีอคติทางการรู้คิดเลย ซึ่งจากตัวอย่างที่ยกไว้ในหัวข้อการอภิปรายผลจากผลที่ได้ในรูปแบบกราฟแสดงความสัมพันธ์และแนวโน้ม ก็พบว่ามีความสมเหตุสมผลในระดับหนึ่งกับลักษณะของการมีอคติในเชิงจิตวิทยา ซึ่งโดยสรุปสามารถกล่าวได้ว่า การมีอคติก็ไม่ได้จะส่งผลในแง่ลบเสมอไป อคติก็สามารถที่จะส่งผลในแง่บวกเช่นกัน ซึ่งในความเป็นจริงแล้ว การมีอคติคือการมีมุมมอง (Perception) ในสิ่งสิ่งหนึ่งจากมุมต่าง ๆ กัน ที่มุมมองนั้นก็ไม่จำเป็นที่จะส่งผลในแง่ลบเพียงแง่เดียว หรือส่งผลบวกเพียงแง่เดียว เช่นกันกับการมีอคติกับการทำงานในงานวิศวกรรมซอฟต์แวร์ ทั้งในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ และกระบวนการออกแบบซอฟต์แวร์ดังที่ได้ทำการวิจัย ในบางครั้งก็พบว่าการมีอคติมากเกินไป หรือน้อยเกินไปก็อาจจะส่งผลให้เกิดความเสียหายได้มากกว่าการมีอคติในระดับปานกลางได้เช่นกัน

ดังนั้นในการบริหารจัดการความเสี่ยง ก็ควรที่จะให้ความสำคัญกับประเด็นนี้เช่นกัน เพราะก็อาจจะเป็นสาเหตุหนึ่งของความล้มเหลวหรือความสำเร็จของโครงการได้เช่นกัน ดังที่ได้กล่าวไว้ในบทนำว่า มนุษย์เป็นทรัพยากรที่สำคัญในโครงการพัฒนาซอฟต์แวร์ เป็นตัวชี้นำความล้มเหลวหรือความสำเร็จของโครงการ ดังนั้นปัจจัยต่าง ๆ ที่เกี่ยวกับมนุษย์จึงเป็นปัจจัยที่สำคัญ ซึ่งรวมถึงการมีอคติทางการรู้คิดในการทำงาน ที่อาจจะเป็นสาเหตุหนึ่งของความล้มเหลวหรือความสำเร็จของโครงการได้เช่นกัน

6.2 ข้อจำกัด

1) แบบจำลองการประเมินความเสี่ยงนี้ใช้วิธีการแปลงค่าจากตัวแปรอิสระซึ่งเป็นตัวแปรเชิงคุณภาพมาเป็นตัวแปรหุ่นเพื่อที่จะได้ทำการวิเคราะห์ความถดถอยกับตัวแปรตามซึ่งเป็นตัวแปรเชิงปริมาณได้ ซึ่งเป็นวิธีหนึ่งในการอนุโลมตามการวิจัยทางสังคมศาสตร์ ซึ่งในงานวิจัยนี้มีปัจจัยหรือตัวแปรอิสระที่เป็นปัจจัยเชิงสังคมศาสตร์นั่นคือการมีอคติอยู่ด้วยเช่นกัน ซึ่งตัวแปร

อิสระที่ถูกเก็บค่ามาเป็นตัวแปรเชิงคุณภาพ ก่อนที่จะทำการวิเคราะห์การถดถอยจึงต้องจำเป็นที่จะต้องแปลงค่าเป็นตัวแปรหุ่นก่อนเสมอ

2) แบบจำลองการประเมินความเสี่ยงนี้อธิบายได้แค่การมีอคติแต่ละตัวในแต่ละระดับนั้นมีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อย่างไร แต่ไม่สามารถอธิบายได้ว่า อคติแต่ละตัวมีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อย่างไร เนื่องจากการเก็บข้อมูลที่ได้มาเก็บมาเป็นในรูปแบบตัวแปรเชิงคุณภาพ สำหรับข้อมูลที่สามารถอธิบายได้ว่าอคติแต่ละตัวนั้นมีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อย่างไรนั้น เป็นข้อมูลในชุดที่ทำการสำรวจกับกลุ่มตัวอย่างเพื่อที่จะนำเอาคำสำคัญ รูปประโยค จากการตอบของกลุ่มตัวอย่างมาสร้างเป็นลักษณะเหตุการณ์ตัวอย่างสำหรับการสร้างแบบสอบถามเพื่อที่จะนำเอาข้อมูลมาสร้างแบบจำลองนี้เท่านั้น

3) เนื่องจากเรื่องอคติทางการรู้คิดเป็นเรื่องทางเชิงจิตวิทยา ซึ่งเป็นเรื่องนอกวงการวิศวกรรมซอฟต์แวร์ การที่จะอธิบายให้บุคคลที่อยู่ในสายงานวิศวกรรมซอฟต์แวร์เข้าใจถึงแนวคิดที่นำเสนอ และเข้าใจถึงอคติในแต่ละประเภทด้วยการอธิบายทั้งในเชิงเอกสารหรือเชิงการบรรยายในช่วงระยะเวลาสั้น ๆ นั้น อาจจะเป็นไปได้ยากหรือเป็นไปได้บ้าง ดังนั้นผู้ตอบแบบสอบถามจึงมีความรู้ความเข้าใจไม่เพียงพอ และอาจจะทำให้การตอบแบบสอบถามได้ไม่ดีพอ หรือไม่ให้ความสำคัญมากพอ ทำให้ในการวิเคราะห์ข้อมูล จึงจำเป็นต้องมีการเรียบเรียงจัดกลุ่ม ลดทอนข้อมูลที่ผิดพลาด เพื่อให้สามารถวิเคราะห์ผลออกมาได้ และผลการทดลองก็อาจจะมีความคลาดเคลื่อนได้

4) ค่าความเสี่ยงของทั้งสองกระบวนการดังกล่าว ไม่สามารถนำมาคำนวณร่วมกันได้ เนื่องจากข้อจำกัดของกลุ่มตัวอย่าง ที่กลุ่มตัวอย่างบางกลุ่มอาจจะใช้อยู่ในการคำนวณกับกระบวนการหนึ่ง และในขณะเดียวกันกลุ่มตัวอย่างอีกกลุ่มก็อาจจะใช้อยู่ในการคำนวณของทั้งสองกระบวนการ ดังนั้นแบบจำลองนี้จึงไม่ได้ถูกออกแบบมาให้หาค่าความเสี่ยงของทั้งสองกระบวนการในคราวเดียวกัน

5) เนื่องจากกระบวนการวิศวกรรมซอฟต์แวร์ยังมีกระบวนการย่อย ๆ อีกมากมายที่อยู่ในกระบวนการตั้งแต่เริ่มจนผลิตเสร็จออกมาเป็นซอฟต์แวร์ ดังนั้นค่าความเสี่ยงจากอคติทางการรู้คิดของทั้งสองกระบวนการจึงเป็นแค่ส่วนหนึ่งของการหาความเสี่ยงทั้งหมดที่อาจจะมี

ปรากฏอยู่ในกระบวนการอื่น ดังนั้นความสามารถในการพยากรณ์ของความเสี่ยงที่ได้จึงอาจจะไม่สูงมากนัก

6.3 ข้อเสนอแนะ

6.3.1 ข้อเสนอแนะในการวิจัย

การพัฒนาแบบจำลองและแนวคิดในการประเมินความเสี่ยงนี้ อาจจะสามารกระทำได้ต่อไปในอนาคต ดังนี้

1) การวิจัยนี้สามารถบอกค่าความเสี่ยงได้เพียงบางส่วนเท่านั้น ซึ่งการที่จะให้การคำนวณออกมาแม่นยำมากกว่านี้ อาจจะจำเป็นที่จะต้องใช้กลุ่มตัวอย่างที่มากขึ้นกว่านี้
2) การสนใจในการหาความสัมพันธ์ระหว่างการมีอคติแต่ละประเภทกับความเสียหายของซอฟต์แวร์ ทั้งในกระบวนการเดิมและกระบวนการอื่น ๆ อีกในวิศวกรรมซอฟต์แวร์
3) ยังมีปัจจัยทางด้านจิตวิทยาอื่น ๆ อีกในการที่จะส่งผลต่อโครงการซอฟต์แวร์ได้ ซึ่งล้วนแล้วแต่เป็นประเด็นเกี่ยวกับมนุษย์ที่ทำงานในโครงการซอฟต์แวร์ทั้งสิ้น
4) นอกเหนือจากปัจจัยทางด้านจิตวิทยากับวิศวกรรมซอฟต์แวร์ ดังเช่นในงานวิจัยนี้แล้ว ยังสามารถทำการวิจัยในรูปแบบข้ามสายวิชาหรือสายงานในลักษณะอื่น ๆ ได้อีก ทั้งนี้ก็เพื่อเป็นการเปิดโลกทัศน์ของการวิจัยในสายวิศวกรรมซอฟต์แวร์ ในการศึกษาหลักวิชาของสายงานอื่น ๆ เพื่อมาปรับปรุงหรือประยุกต์ใช้ให้เกิดประโยชน์ในสายงานวิศวกรรมซอฟต์แวร์

6.3.2 ข้อเสนอแนะในการใช้แบบจำลองและเครื่องมือสนับสนุนแบบจำลอง

การใช้งานแบบจำลองนี้สามารถใช้งานผ่านเครื่องมือสนับสนุนแบบจำลองชื่อ “Biskuit” ซึ่งจำเป็นที่ต้องการข้อมูลนำเข้าที่เป็นระดับการมีอคติประเภทต่าง ๆ ในการทำงานในแต่ละกระบวนการ ซึ่งการที่จะได้มาซึ่งข้อมูลนี้ ใช้เครื่องมือจำเป็นที่จะต้องทำการสำรวจบุคลากรในองค์กร หน่วยงาน หรือทีมงานเสียก่อน ว่ามีระดับการมีอคติในการทำงานอยู่ในระดับใด ซึ่งอาจจะประยุกต์เอาแบบสอบถามที่ใช้งานในวิจัยนี้ไปใช้ในการสำรวจ ซึ่งจำเป็นต้องนำไปปรับใช้อย่างเหมาะสมให้เข้ากับองค์กร หน่วยงาน หรือทีมงาน โดยอาจจะจำเป็นต้องอธิบายถึงประเภทและลักษณะของอคติต่าง ๆ ดังกล่าวก่อน หลังจากนั้นจึงจะนำข้อมูลมากรอกในเครื่องมือ และทำการคำนวณต่อไป

ซึ่งค่าที่ได้จากเครื่องมือสนับสนุนแบบจำลองนี้อาจจะพยากรณ์ได้ไม่แม่นยำมากนัก เป็นเพียงส่วนหนึ่งที่จะชี้ให้เห็นว่าองค์กรหรือหน่วยงานมีความเสี่ยงอยู่ในระดับเท่าใดของแบบจำลองนี้ ซึ่งการจะตัดสินใจต่าง ๆ เกี่ยวกับการบริหารความเสี่ยงก็ควรที่จะคำนึงถึงความเสี่ยงเหล่านี้เอาไว้ด้วย เพื่อที่จะพยายามลดหรือปรับเปลี่ยนทัศนคติในการทำงานให้มีอคติอยู่ในระดับที่มีความเสี่ยงต่ำลง

รายการอ้างอิง

[1] Antony Tang. Software Designers, Are You Biased? ACM, 2011.

[2] Andrew P. Snow, Mark Keil, and Linda Wallace. The Effects of Optimistic and Pessimistic Biasing on Software Project Status Reporting. Information & Management, 44. pp. 130–141. 2007.

[3] A.P. Snow, and M. Keil. The Challenge of Accurate Project Status Reporting: a Two Stage Model Incorporating Status Errors and Reporting Bias. Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS-34), Kihei, Hawaii, pp. 1–10. 2001.

[4] A.P. Snow, and M. Keil. The Challenge of Accurate Software Project Status Reporting : A Two-stage Model Incorporating Status Errors and Reporting Bias. IEEE Transactions on Engineering Management 49 (4) (2002) : 491–504.

[5] Baron J. Thinking and deciding (4th edition). New York, NY: Cambridge University Press, 2007.

[6] Ariely D. Predictably irrational: The hidden forces that shape our decisions. New York, NY: HarperCollins, 2008.

[7] คัคนางค์ มณีศรี. จิตวิทยาทั่วไป (General Psychology). พิมพ์ครั้งที่ 2. กรุงเทพมหานคร: ช่อระกา การพิมพ์. 2555.

[8] Kannan Mohan, and Radhika Jain. Using Traceability to Mitigate Cognitive Biases in Software Development. Communications of the ACM - CACM, vol. 51, no. 9, pp. 110-114. 2008

[9] Stacy W., and MacMillan J. Cognitive Bias in Software Engineering. Communications of the ACM - CACM, 38 (6), pp. 57 – 63. 1995.

[10] Parsons J., and Saunders C. Cognitive Heuristics in Software Engineering: Applying and Extending Anchoring and Adjustment to Artifact Reuse. IEEE Transactions on Software Engineering 30 (12) (December 2004) : 873-888.

[11] Fisher K., and Statman M. Cognitive Biases in Market Forecasts: The frailty of forecasting. The Journal of Portfolio Management (2000) : 1-10.

[12] Jorge Aranda, and Steve M. Easterbrook. Anchoring and Adjustment in Software Estimation. ACM Sigsoft Software Engineering Notes, vol. 30, no. 5, pp. 346-355. 2005.

[13] Kahneman D., and Tversky A. On the Psychology of Prediction. Psychological Review - PSYCHOL REV, vol. 80, no. 4, pp. 237-251. 1973.

[14] Kydd C.T. Cognitive Biases in the Use of Computerbased Decision Support Systems. Omega, 17 (4), pp. 335-344. 1989.

[15] ISO 31000:2009 Risk management — Principles and guidelines.

[16] Boehm, B. W. A Spiral Model of Software Development and Enhancement. IEEE Computer, 21(5): 61-72. 1988.

[17] Olofsson, Peter. Probability, Statistics, and Stochastic Processes. Houston, Texas: Wiley-Interscience, 2005.

[18] Watcharee Wattanapokasin, and Wanchai Rivepiboon. Cross-Cultural Risk Assessment Model. International Conference on Signal Processing Systems - ICSPS. 2009.

[19] กัลยา วานิชย์บัญชา. หลักสถิติ. พิมพ์ครั้งที่ 7. กรุงเทพมหานคร: โรงพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย. 2545.

[20] ชวลิต ทับสีรัก. ตัวแปรเชิงคุณภาพกับการวิเคราะห์การถดถอย [Online]. 2555. Available from: http://grad.msu.ac.th/jem/home/journal_file/241.pdf [1 กรกฎาคม 2555]

[21] ศิริชัย กาญจนวาสี. การวิเคราะห์พหุระดับ. พิมพ์ครั้งที่ 4. กรุงเทพฯ : โรงพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย. 2550.

[22] Luiz Fernando Capretz, and Faheem Ahmed. Why Do We Need Personality Diversity in Software Engineering? ACM Sigsoft Software Engineering Notes, vol. 35, no. 2, pp. 1-11. 2010.

[23] Gul Calikli, and Ayse Basar Bener. Empirical Analyses of the Factors Affecting Confirmation Bias and the Effects of Confirmation Bias on Software Developer/Tester Performance. ACM SIGSOFT Software Engineering Notes, vol. 35, no. 2, 2010.

[24] Hathaichanok Suwanjang, and Nakornthip Prompoon. Framework for

Developing a Software Cost Estimation Model for Software Modification Based on a Relational Matrix of Project Profile and Software Cost Using and Analogy Estimation Method. The 2012 International Conference on Software and Information Engineering – ICSIE 2012. 2012.

[25] Simmons, J. P., LeBoeuf, R. A., and Nelson, L. D. The Effect of Accuracy Motivation on Anchoring and Adjustment: Do People Adjust from Provided Anchors? Journal of Personality and Social Psychology 99 (6) (2002) : 917-932.

[26] Cynthia Mullen Kunsman. Understanding Cognitve Dissonance Part V [Online]. 2009. Available from: http://undermuchgrace.blogspot.com/2009/03/understanding-cognitve-dissonance-part.html [2009, MARCH 13]

[27] Pavankumar Pothuraju. Defect Serverity and Defect Priority [Online]. 2005. Available from: http://geekswithblogs.net/ppothuraju/archive/2005/01/22/20664.aspx [2005, JANUARY 22]

[28] รัฐ จำเดิมเผด็จศึก. ความต่างข้อกฎหมายเรื่องเครื่องบินตก [Online]. 2534. Available from: http://info.gotomanager.com/news/printnews.aspx?id=6084 [2008]

[29] Wikipedia. Warsaw Convention [Online]. 2013. Available from: http://en.wikipedia.org/wiki/Warsaw_Convention [2013, February 28]

[30] บุญธรรม กิจปรีดาบริสุทธิ์. สถิติวิเคราะห์เพื่อการวิจัย. พิมพ์ครั้งที่ 4. กรุงเทพฯ : จามจุรีโปรดักท์. 2549.

ภาคผนวก

ภาคผนวก ก

แบบสอบถามที่ใช้ในการเก็บข้อมูล

แบบสอบถามงานวิจัย

เรื่อง แบบจำลองการประเมินความเสี่ยงของอัตราความเสียหายของซอฟต์แวร์จากการมีอคติทางการรู้คิด ห้องปฏิบัติการวิศวกรรมซอฟต์แวร์ ศูนย์เชี่ยวชาญเฉพาะทางด้านวิศวกรรมซอฟต์แวร์

สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

วัตถุประสงค์: เพื่อใช้ในการรวบรวมความเห็นจากกลุ่มตัวอย่าง ในเรื่องเกี่ยวกับประสบการณ์ในการปฏิบัติงานในโครงการซอฟต์แวร์ที่อาจจะมีอคติทางการรู้คิดเกิดขึ้นระหว่างปฏิบัติงานได้ โดยมีปรากฏอยู่ในลักษณะประโยคที่ใช้ในการสื่อสารระหว่างกัน ซึ่งการมีอคติทางการรู้คิดนี้ อาจจะทำให้เกิดความเสียหายต่อผลลัพธ์ของงาน นั่นคือซอฟต์แวร์หรือระบบได้ งานวิจัยนี้จึงมุ่งที่จะศึกษาหาความเสี่ยงและผลกระทบต่อผลลัพธ์ของงานอันมีสาเหตุมาจากการมีอคติทางการรู้คิดในการปฏิบัติงานในโครงการซอฟต์แวร์

จำนวนแบบสอบถาม: แบบสอบถามนี้มีจำนวนหน้าทั้งหมด 16 หน้า โดยแบ่งออกเป็นดังนี้

1. ใบปะหน้า แสดงวัตถุประสงค์และรายละเอียดของการทำแบบสอบถาม จำนวน 1 หน้า
2. แบบสอบถามข้อมูลส่วนบุคคล สำหรับสอบถามข้อมูลของกลุ่มตัวอย่าง อย่างพอสังเขป จำนวน 3 หน้า
3. แบบสอบถามสำหรับการพบอคติในการรวบรวมและกำหนดความต้องการซอฟต์แวร์ จำนวน 6 หน้า
4. แบบสอบถามสำหรับการพบอคติในกระบวนการออกแบบซอฟต์แวร์ จำนวน 6 หน้า

ลักษณะการตอบแบบสอบถาม: กลุ่มตัวอย่างจะต้องตอบแบบสอบถามนี้ในส่วนที่ 1 - 3 ดังนี้

1. แบบสอบถามส่วนที่ 1 เป็นลักษณะการตอบคำถามในรูปแบบ
   1.1 อัตนัย เกี่ยวกับข้อมูลส่วนบุคคลของกลุ่มตัวอย่าง
   1.2 ทำเครื่องหมายในตัวเลือกที่กำหนด เกี่ยวกับข้อมูลส่วนบุคคลของกลุ่มตัวอย่าง
2. แบบสอบถามส่วนที่ 2 และ 3 เป็นลักษณะการตอบคำถามในรูปแบบ
   2.1 อัตนัย เกี่ยวกับข้อมูลของโครงการที่ได้ปฏิบัติงาน
   2.2 กรอกค่าระดับ แสดงจำนวนอคติที่เกิดขึ้นจากการปฏิบัติงานของท่าน
   2.3 ทำเครื่องหมายกากบาทหน้าผลกระทบในแง่บวก ลบ หรือไม่บวกไม่ลบ จากการมีเหตุการณ์ตัวอย่างเกิดขึ้น
   2.4 กรอกจำนวนครั้งที่มีการรายงานข้อผิดพลาดของซอฟต์แวร์กลับมาในระยะเวลา 1 เดือน คิดเป็นจำนวนทั้งสิ้นกี่ครั้งจากการทำงานภายหลังที่ได้ส่งมอบซอฟต์แวร์ไปแล้ว โดยจำแนกจำนวนของความเสียหายของซอฟต์แวร์ที่ได้รับรายงานกลับมาออกเป็นแต่ละระดับความรุนแรง 5 ระดับ คือ ระดับวิกฤต ระดับมาก ระดับปานกลาง ระดับน้อย และระดับที่ละเว้นได้

แบบสอบถามส่วนที่ 1

ข้อมูลส่วนบุคคลของกลุ่มตัวอย่าง

<u>แบบสอบถามส่วนที่ 1: แบบสอบถามข้อมูลส่วนบุคคล</u>

ข้อแนะนำ: แบบสอบถามส่วนนี้จะสอบถามถึงข้อมูลส่วนบุคคลของกลุ่มตัวอย่างที่ทำงานในส่วนที่เกี่ยวข้องกับเทคโนโลยีสารสนเทศขององค์กร โดยให้เลือกตอบ กรอกรายละเอียด และทำเครื่องหมายถูกตามข้อที่เป็นจริง

1) เพศ ☐ ชาย ☐ หญิง
2) อายุ _____ ปี
3) อายุงานโดยรวม _____ ปี
4) วุฒิการศึกษาระดับปริญญาตรี
    a. ด้านคอมพิวเตอร์
        ☐ คอมพิวเตอร์ธุรกิจ (Business Computer)
        ☐ วิศวกรรมคอมพิวเตอร์ (Computer Engineering)
        ☐ วิทยาการคอมพิวเตอร์ (Computer Science)
        ☐ วิทยาการสารสนเทศ (Information Science)
        ☐ ระบบสารสนเทศ (Information System)
        ☐ เทคโนโลยีสารสนเทศ (Information Technology)
        ☐ วิศวกรรมซอฟต์แวร์ (Software Engineering)
        ☐ อื่น ๆ โปรดระบุ _____
    b. ด้านอื่น ๆ (ที่นอกเหนือจากด้านคอมพิวเตอร์)
        โปรดระบุ _____
5) กำลังศึกษาต่ออยู่ในระดับ
    ☐ ประกาศนียบัตรบัณฑิต ☐ ปริญญาโท ☐ ปริญญาเอก
    หลักสูตร _____
    สาขาวิชา/ภาควิชา _____
    สถานศึกษา _____
6) หน้าที่ที่รับผิดชอบที่ท่านปฏิบัติงานอยู่ในปัจจุบัน
    ตำแหน่ง _____
    ฝ่าย/แผนก _____
    อายุงานในตำแหน่งปัจจุบัน _____ ปี
    ลักษณะงาน
    _____
    _____
    _____

7) องค์กร กลุ่มอุตสาหกรรมและหมวดธุรกิจ ที่ท่านปฏิบัติงานอยู่ในปัจจุบัน

☐ เกษตรและอุตสาหกรรมอาหาร (ธุรกิจการเกษตร, อาหารและเครื่องดื่ม)

☐ สินค้าอุปโภคบริโภค (แฟชั่น, ของใช้ในครัวเรือนและสำนักงาน, ของใช้ส่วนตัวและเวชภัณฑ์)

☐ ธุรกิจการเงิน (ธนาคาร, เงินทุนและหลักทรัพย์, ประกันภัยและประกันชีวิต)

☐ สินค้าอุตสาหกรรม (ยานยนต์, วัสดุอุตสาหกรรมและเครื่องจักร, กระดาษและวัสดุการพิมพ์, ปิโตรเคมีและเคมีภัณฑ์, บรรจุภัณฑ์, เหล็ก)

☐ อสังหาริมทรัพย์และก่อสร้าง (วัสดุก่อสร้าง, กองทุนรวมอสังหาริมทรัพย์, พัฒนาอสังหาริมทรัพย์)

☐ ทรัพยากร (พลังงานและสาธารณูปโภค, เหมืองแร่)

☐ บริการ (การท่องเที่ยวและสันทนาการ, ขนส่งและโลจิสติกส์, สื่อและสิ่งพิมพ์, บริการเฉพาะกิจ, พาณิชย์, การแพทย์)

☐ เทคโนโลยี (ชิ้นส่วนอิเล็กทรอนิกส์, เทคโนโลยีสารสนเทศและการสื่อสาร)

☐ ส่วนราชการ (ทหาร, ตำรวจ, อาจารย์, นักวิชาการ, ฯลฯ)

☐ รัฐวิสาหกิจ (ไฟฟ้า, ประปา, การสื่อสาร, การคมนาคม, ฯลฯ)

☐ องค์การมหาชน (สังกัดหน่วยงานต่าง ๆ ของรัฐ)

☐ อื่น ๆ โปรดระบุ _____

8) มีประสบการณ์ในการปฏิบัติงานที่เกี่ยวข้องกับ

☐ การรวบรวมและกำหนดความต้องการของซอฟต์แวร์/ระบบ _____ ปี

☐ การออกแบบซอฟต์แวร์/ระบบ _____ ปี

☐ การพัฒนาซอฟต์แวร์/ระบบ _____ ปี

☐ การบริหารจัดการโครงการซอฟต์แวร์/ระบบ _____ ปี

และใช้ตารางที่ 1 ในการตอบแบบสอบถามในส่วนที่ 2 และ 3

ตารางที่ 1. ค่าระดับของจำนวนร้อยละของการทำงานทั้งหมดในโครงการที่ท่านมีส่วนร่วมในแต่ละกระบวนการมากที่สุด

จำนวนร้อยละที่เกิดเหตุการณ์ขึ้น	ค่าระดับ
0-20%	1
21-40%	2
41-60%	3
61-80%	4
81-100%	5

แบบสอบถามส่วนที่ 2

สำหรับการพบอคติในการรวบรวมและกำหนดความต้องการซอฟต์แวร์

<u>แบบสอบถามส่วนที่ 2: แบบสอบถามสำหรับการพบอคติในการรวบรวมและกำหนดความต้องการซอฟต์แวร์</u>

ส่วนที่ 2.1: แบบสอบถามข้อมูลเกี่ยวกับโครงการที่ปฏิบัติงาน

ข้อแนะนำ: กรอกรายละเอียดของโครงการที่ได้ปฏิบัติงานที่ท่านมีส่วนร่วมในกระบวนการด้านการรวบรวมและกำหนดความต้องการมากที่สุด

1) ชื่อโครงการที่ท่านปฏิบัติงานด้านการรวบรวมและกำหนดความต้องการ* _____
   (* หากเป็นความลับขององค์กร ให้ตอบแค่พอเข้าใจว่าเป็นโครงการอะไร โดยไม่ต้องระบุชื่อชัดเจน)
2) ลักษณะของโครงการ
   _____
   _____
   _____
   _____
3) ระยะเวลาในการทำโครงการ _____ ปี _____ เดือน _____ วัน
4) จำนวนบุคลากรในโครงการ _____ คน

ส่วนที่ 2.2: แบบสอบถามเกี่ยวกับอคติในการรวบรวมและกำหนดความต้องการ

ข้อแนะนำ: ศึกษาประเภทอคติ คำอธิบาย และตัวอย่างเหตุการณ์ที่เกิดอคติที่กำหนดให้ หรือหากมีเหตุการณ์ที่พบจริงในการปฏิบัติงานสามารถกรอกเพิ่มเติมได้ในช่องอื่น ๆ แล้วกรอกตัวเลขเป็นจำนวนเปอร์เซ็นต์ของการทำงานที่เกิดเหตุการณ์เหล่านั้นใน<u>โครงการที่ท่านมีส่วนร่วมในกระบวนการด้านการรวบรวมและกำหนดความต้องการมากที่สุด</u> (ที่กรอกไว้ในส่วนที่ 2.1) พร้อมทั้งระบุการส่งผลต่อกระบวนการด้านการรวบรวมและกำหนดความต้องการของเหตุการณ์ที่กำหนดให้ ว่าเป็นแง่บวก แง่ลบ หรือไม่บวกไม่ลบ

ส่วนที่ 2.2.1: แบบสอบถามอคติประเภทที่ 1 (Anchoring & Adjustment) ใน<u>การรวบรวมและกำหนดความต้องการ</u>

ประเภทของอคติที่เกิดขึ้น/คำอธิบาย/ตัวอย่างเหตุการณ์	จำนวนครั้งที่เกิดเหตุการณ์ตัวอย่างขึ้นในการรวบรวมและกำหนดความต้องการ*
1. Anchoring & Adjustment *การที่มนุษย์พยายามที่จะสร้างการประมาณการเกี่ยวกับปัญหาขึ้นมา โดยมีแนวโน้มที่ยึดติดอยู่กับความรู้เดิมที่เคยรู้มาก่อน เพื่อหาวิธีสำหรับการแก้ปัญหาต่าง ๆ โดยปัญหาในการประมาณการเบื้องต้นที่อาจจะมีข้อผิดพลาดจากการประมาณจากประสบการณ์ของตน ทำให้ในการประมาณการเบื้องต้นทุกครั้งจะต้องทำการปรับแต่งให้เหมาะสมกับสถานการณ์ปัจจุบัน หรือความเป็นจริง*	* ระบุค่าระดับตามตารางที่ 1 ของการเกิดเหตุการณ์ตัวอย่างขึ้นในการปฏิบัติงานด้านการรวบรวมและกำหนดความต้องการซอฟต์แวร์
<u>(ตัวอย่าง)</u> จากประสบการณ์ของผม/ดิฉัน คิดว่าระบบซอฟต์แวร์ควรที่จะเป็นไปในประมาณลักษณะนี้ (ที่เป็นไปตามที่ผู้พูดคิด)	5
1.1 จากประสบการณ์ของผม/ดิฉัน คิดว่าระบบซอฟต์แวร์ควรที่จะเป็นไปในประมาณลักษณะนี้ (ที่เป็นไปตามที่ผู้พูดคิด)	
1.2 จากประสบการณ์ คิดว่าระบบซอฟต์แวร์ควรที่จะเปลี่ยนแปลงหรือปรับปรุงมาเป็นประมาณนี้ (ที่เป็นไปตามที่ผู้พูดคิด)	
1.3 อื่น ๆ (ถ้ามี)	
(สำหรับผู้วิจัย) ค่าเฉลี่ยของ Anchoring & Adjustment	

ส่วนที่ 2.2.2: แบบสอบถามอคติประเภทที่ 2 (Availability) ในการรวบรวมและกำหนดความต้องการ

ประเภทของอคติที่เกิดขึ้น/คำอธิบาย/ตัวอย่างเหตุการณ์	จำนวนครั้งที่เกิดเหตุการณ์ตัวอย่างขึ้นในการรวบรวมและกำหนดความต้องการ*
2. Availability *การที่มนุษย์พยายามนึกถึงกรณีเฉพาะเจาะจงที่เกี่ยวข้องกับเรื่องที่เราจะตัดสินใจ เช่น ในการนึกถึงสถานการณ์หนึ่งแล้วมีความคิดนึกถึงเกิดขึ้นมาได้ง่าย เราจะมองว่าสถานการณ์นั้นเกิดขึ้นเป็นปกติ ถ้าเรานึกถึงได้ยาก เราจะมองว่าสถานการณ์นั้นเกิดขึ้นได้น้อยมาก โดยมิได้คำนึงถึงความเหมาะสม หรือความเป็นจริง เช่น การนึกถึงเหตุการณ์ปัจจุบันเป็นหลัก เรื่องที่กำลังเป็นกระแส หรือเรื่องที่คุ้นชิน เป็นต้น*	* ระบุค่าระดับตามตารางที่ 1 ของการเกิดเหตุการณ์ตัวอย่างขึ้นในการปฏิบัติงานด้านการรวบรวมและกำหนดความต้องการซอฟต์แวร์
(ตัวอย่าง) เนื่องจากความนิยมในเรื่อง ... (เช่น Could Computing, Mobile Application เป็นต้น) ที่มีอยู่ในปัจจุบัน ระบบซอฟต์แวร์ควรจะให้มีความสามารถเหมือนกับความนิยมนั้น	3
2.1 เนื่องจากความนิยมในเรื่อง ... (เช่น Could Computing, Mobile Application เป็นต้น) ที่มีอยู่ในปัจจุบัน ระบบซอฟต์แวร์ควรจะให้มีความสามารถเหมือนกับความนิยมนั้น	
2.2 ควรจะเปลี่ยนแปลงหรือปรับปรุงฟังก์ชันการทำงานให้รองรับกับกระแสนิยมในปัจจุบัน เทคโนโลยีล่าสุด หรือเทคโนโลยีในอนาคต	
2.3 สำหรับระบบซอฟต์แวร์นี้ ด้วยเทคโนโลยีแล้วสามารถ (หรือไม่สามารถ) ทำให้เหมือนกับความต้องการตามกระแสนิยมในปัจจุบันได้	
2.4 การเปลี่ยนแปลงหรือปรับปรุงระบบซอฟต์แวร์นี้ จากเทคโนโลยีที่มีอยู่ตอนนี้แล้วสามารถ (หรือไม่สามารถ) ทำให้เหมือนกับ ความต้องการตามกระแสนิยมในปัจจุบันได้	
2.5 คิดว่าด้วยเทคโนโลยีในตอนนี้ ระบบซอฟต์แวร์นี้ สามารถ (หรือไม่สามารถ) ทำให้เหมือนกับความต้องการตามกระแสนิยมในปัจจุบันได้	
2.6 อื่น ๆ (ถ้ามี)	
(สำหรับผู้วิจัย) ค่าเฉลี่ยของ Availability	

ส่วนที่ 2.2.3: แบบสอบถามอคติประเภทที่ 3 (Confirmation) ในการรวบรวมและกำหนดความต้องการ

ประเภทของอคติที่เกิดขึ้น/คำอธิบาย/ตัวอย่างเหตุการณ์	จำนวนครั้งที่เกิดเหตุการณ์ตัวอย่างขึ้นในการรวบรวมและกำหนดความต้องการ*
3. Confirmation *การที่มนุษย์มักจะมีการยืนยันความคิด หรือความเชื่อของตน ซึ่งสามารถเกิดขึ้นได้ในหลายรูปแบบ เช่น เมื่อมนุษย์มีโอกาสที่จะหาข้อมูลใหม่ มนุษย์ก็จะมักที่จะหาข้อมูลที่ยืนยันความเชื่อของตนเองมากกว่าจะหาข้อมูลที่ค้านกับความเชื่อของตนเอง*	* ระบุค่าระดับตามตารางที่ 1 ของการเกิดเหตุการณ์ตัวอย่างขึ้นในการปฏิบัติงานด้านการรวบรวมและกำหนดความต้องการซอฟต์แวร์
(ตัวอย่าง) ผม/ดิฉัน มั่นใจว่าระบบซอฟต์แวร์ต้องแสดงผลหรือให้ผลลัพธ์เป็นอย่างนี้	4
3.1 ผม/ดิฉัน มั่นใจว่าระบบซอฟต์แวร์ต้องแสดงผลหรือให้ผลลัพธ์เป็นอย่างนี้	
3.2 ทำไมระบบซอฟต์แวร์นี้ จึงไม่เป็นไปในลักษณะนั้น (ลักษณะอื่น ๆ ที่ผู้กล่าวคิดว่าดีกว่าลักษณะเดิม) ซึ่งถูกต้องและเหมาะสมมากกว่า	
3.3 ต้องเปลี่ยนแปลงหรือปรับปรุง ระบบซอฟต์แวร์ในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน) นั่นจะทำให้ระบบซอฟต์แวร์เดิมนั้นดีขึ้นหรือมีประสิทธิภาพมากขึ้น	
3.4 จากประสบการณ์คิดว่าระบบซอฟต์แวร์ควรจะเป็นอย่างนี้หรือเป็นแบบนี้ (ตามที่ผู้พูดคิด)	
3.5 คิดว่าควรจะเปลี่ยนแปลงหรือปรับปรุง ระบบซอฟต์แวร์ ในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน)	
3.6 คิดว่าน่าจะเปลี่ยนแปลงหรือปรับปรุงระบบซอฟต์แวร์ในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน) นั่นจะทำให้ระบบซอฟต์แวร์เดิมนั้นดีขึ้นหรือมีประสิทธิภาพมากขึ้น	
3.7 อื่น ๆ (ถ้ามี)	
(สำหรับผู้วิจัย) ผลรวมของอคติทั้ง 3 ประเภท	

ส่วนที่ 2.2.4: แบบสอบถามการรายงานข้อผิดพลาดใน<u>การรวบรวมและกำหนดความต้องการ</u>

โดยกำหนดให้ ในงานด้านการรวบรวมและกำหนดความต้องการที่ท่านรับผิดชอบ มีเหตุการณ์ดังกล่าวข้างต้นเกิดขึ้น และเมื่อทำการส่งมอบซอฟต์แวร์ไปแล้ว เป็นระยะเวลา 1 เดือน มีจำนวนของข้อผิดพลาดที่เกิดขึ้นกับซอฟต์แวร์ที่มีการรายงานกลับมา **<u>คิดเป็นจำนวนกี่ครั้ง</u>**	_____ ครั้ง
โดย**<u>จำแนกจำนวนของข้อผิดพลาด</u>**จากข้อคำถามด้านบน ออกเป็นแต่ละระดับความรุนแรงดังนี้*	วิกฤต _____ มาก _____ ปานกลาง _____ ครั้ง น้อย _____ ละเว้นได้ _____
*คำนิยามระดับความรุนแรง	
1. ระดับวิกฤต (Critical) คือ ข้อผิดพลาดที่ส่งผลต่อความล้มเหลวของระบบซอฟต์แวร์ ระบบย่อย โปรแกรมหรือโมดูล โดยสมบูรณ์ 2. ระดับมาก (Major) คือ ข้อผิดพลาดที่ส่งผลต่อความล้มเหลวของตัวระบบซอฟต์แวร์ ระบบย่อย โปรแกรมหรือโมดูลเอง แต่ไม่ส่งผลต่อของส่วนงานอื่น ๆ ข้อผิดพลาดนี้สามารถยอมรับได้หากมีการประมวลผลด้วยวิธีอื่นที่จะทำให้ได้ผลลัพธ์ตามที่ต้องการ แต่ยังคงต้องแก้ไขข้อผิดพลาดนั้น 3. ระดับปานกลาง (Average) คือ ข้อผิดพลาดที่ไม่ได้ส่งผลต่อความล้มเหลว แต่ทำให้ระบบทำงานได้ไม่ถูกต้องสมบูรณ์ ผลลัพธ์ไม่สอดคล้องกัน หรือข้อผิดพลาดลดทอนการใช้งานระบบ 4. ระดับน้อย (Minor) คือ ข้อผิดพลาดที่ไม่ได้ก่อให้เกิดความล้มเหลว ไม่ได้ลดทอนการใช้งานระบบ และยังสามารถให้ผลลัพธ์ที่ต้องการจากการประมวลผลได้ด้วยการแก้ไขข้อผิดพลาดที่ไม่ยุ่งยาก 5. ระดับละเว้นได้ (Exception) คือ ข้อผิดพลาดที่เป็นผลมาจากความไม่สอดคล้องกันกับมาตรฐาน มีความเกี่ยวเนื่องกับคุณลักษณะภายนอกของระบบ เช่น ความสวยงาม สีสัน หรือเป็นความต้องการเพื่อปรับปรุงให้ดูดีขึ้น ข้อผิดพลาดในระดับนี้อาจจะผ่อนผันหรือแม้แต่ละเว้นการแก้ไขไปได้	

แบบสอบถามส่วนที่ 3

สำหรับการพบอคติในกระบวนการออกแบบซอฟต์แวร์

<u>แบบสอบถามส่วนที่ 3: แบบสอบถามสำหรับการพบอคติในกระบวนการออกแบบซอฟต์แวร์</u>

ส่วนที่ 3.1: แบบสอบถามข้อมูลเกี่ยวกับโครงการที่ปฏิบัติงาน

ข้อแนะนำ: กรอกรายละเอียดของโครงการที่ได้ปฏิบัติงานท่านมีส่วนร่วมในกระบวนการด้านการออกแบบมากที่สุด

1) ชื่อโครงการที่ท่านปฏิบัติงานด้านการออกแบบ* _____
   (* หากเป็นความลับขององค์กร ให้ตอบแค่พอเข้าใจว่าเป็นโครงการอะไร โดยไม่ต้องระบุชื่อชัดเจน)
2) ลักษณะของโครงการ
   _____
   _____
   _____
   _____
3) ระยะเวลาในการทำโครงการ _____ ปี _____ เดือน _____ วัน
4) จำนวนบุคลากรในโครงการ _____ คน

ส่วนที่ 3.2: แบบสอบถามเกี่ยวกับอคติในการออกแบบ

ข้อแนะนำ: ศึกษาประเภทอคติ คำอธิบาย และตัวอย่างเหตุการณ์ที่เกิดอคติที่กำหนดให้ หรือหากมีเหตุการณ์ที่พบจริงในการปฏิบัติงานสามารถกรอกเพิ่มเติมได้ในช่องอื่น ๆ แล้วกรอกตัวเลขเป็นจำนวนเปอร์เซ็นต์ของการทำงานที่เกิดเหตุการณ์เหล่านั้นใน<u>โครงการที่ท่านมีส่วนร่วมในกระบวนการด้านออกแบบซอฟต์แวร์มากที่สุด</u> (ที่กรอกไว้ในส่วนที่ 3.1) พร้อมทั้งระบุการส่งผลต่อกระบวนการด้านออกแบบซอฟต์แวร์ของเหตุการณ์ที่กำหนดให้ ว่าเป็นแง่บวก แง่ลบ หรือไม่บวกไม่ลบ (ต่อหน้าถัดไป...)

ส่วนที่ 3.2.1: แบบสอบถามอคติประเภทที่ 1 (Anchoring & Adjustment) ในกระบวนการออกแบบ

ประเภทของอคติที่เกิดขึ้น/คำอธิบาย/ตัวอย่างเหตุการณ์	จำนวนครั้งที่เกิดเหตุการณ์ตัวอย่างขึ้นในการออกแบบซอฟต์แวร์*
1. Anchoring & Adjustment *การที่มนุษย์พยายามที่จะสร้างการประมาณการเกี่ยวกับปัญหาขึ้นมา โดยมีแนวโน้มที่ยึดติดอยู่กับความรู้เดิมที่เคยรู้มาก่อน เพื่อหาวิธีสำหรับการแก้ปัญหาต่าง ๆ โดยปัญหาในการประมาณการเบื้องต้นที่อาจจะมีข้อผิดพลาดจากการประมาณจากประสบการณ์ของตน ทำให้ในการประมาณการเบื้องต้นทุกครั้งจะต้องทำการปรับแต่งให้เหมาะสมกับสถานการณ์ปัจจุบัน หรือความเป็นจริง*	* ระบุค่าระดับตามตารางที่ 1 ของการเกิดเหตุการณ์ตัวอย่างขึ้นในการปฏิบัติงานด้านการออกแบบซอฟต์แวร์
(ตัวอย่าง) ให้ทำการเปลี่ยนแปลงหรือปรับปรุง ระบบซอฟต์แวร์ในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน) สักหน่อย เพื่อให้มีความสมบูรณ์มากขึ้น	3
1.1 ให้ทำการเปลี่ยนแปลงหรือปรับปรุง ระบบซอฟต์แวร์ในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน) สักหน่อย เพื่อให้มีความสมบูรณ์มากขึ้น	
1.2 ผม/ดิฉัน ต้องการให้ระบบซอฟต์แวร์ถูกออกแบบมาเป็นไปในลักษณะประมาณนี้ (เป็นไปตามที่ผู้พูดคิด)	
1.3 คิดว่าการออกแบบ ระบบซอฟต์แวร์ ที่จะให้มีความสามารถประมาณนี้ นั้นเพียงพอแล้วหรือยัง	
1.4 จากประสบการณ์ คิดว่าระบบซอฟต์แวร์ ควรที่จะเปลี่ยนแปลงหรือปรับปรุงการออกแบบไปเป็นลักษณะประมาณนี้	
1.5 ควรจะเปลี่ยนแปลงและปรับปรุงในส่วนตรงนี้ (ส่วนที่ผู้พูดต้องการให้เปลี่ยน) สักหน่อย เพื่อให้มีความสมบูรณ์มากขึ้น	
1.6 อื่น ๆ (ถ้ามี)	
(สำหรับผู้วิจัย) ค่าเฉลี่ยของ Anchoring & Adjustment	

ส่วนที่ 3.2.2: แบบสอบถามอคติประเภทที่ 2 (Availability) ในกระบวนการออกแบบ

ประเภทของอคติที่เกิดขึ้น/คำอธิบาย/ตัวอย่างเหตุการณ์	จำนวนครั้งที่เกิดเหตุการณ์ตัวอย่างขึ้นในการออกแบบซอฟต์แวร์*
2. Availability *การที่มนุษย์พยายามนึกถึงกรณีเฉพาะเจาะจงที่เกี่ยวข้องกับเรื่องที่เราจะตัดสินใจ เช่น ในการนึกถึงสถานการณ์หนึ่งแล้วมีความคิดนึกถึงเกิดขึ้นมาได้ง่าย เราจะมองว่าสถานการณ์นั้นเกิดขึ้นเป็นปกติ ถ้าเรานึกถึงได้ยาก เราจะมองว่าสถานการณ์นั้นเกิดขึ้นได้น้อยมาก โดยมิได้คำนึงถึงความเหมาะสม หรือความเป็นจริง เช่น การนึกถึงเหตุการณ์ปัจจุบันเป็นหลัก เรื่องที่กำลังเป็นกระแส หรือเรื่องที่คุ้นชิน เป็นต้น*	* ระบุค่าระดับตามตารางที่ 1 ของการเกิดเหตุการณ์ตัวอย่างขึ้นในการปฏิบัติงานด้านการออกแบบซอฟต์แวร์
(ตัวอย่าง) ด้วยเทคโนโลยีและกระแสนิยมในปัจจุบัน คิดว่าระบบซอฟต์แวร์นี้ควรจะออกแบบให้มีความสามารถเหมือนกับที่กำลังนิยมอยู่ในตอนนี้	2
2.1 ด้วยเทคโนโลยีและกระแสนิยมในปัจจุบัน คิดว่าระบบซอฟต์แวร์นี้ควรจะออกแบบให้มีความสามารถเหมือนกับที่กำลังนิยมอยู่ในตอนนี้	
2.2 ควรจะดูแบบอย่าง แล้วนำเอาลักษณะของเทคโนโลยีและกระแสนิยมในปัจจุบัน มาใช้ในการออกแบบระบบซอฟต์แวร์ น่าจะช่วยให้มีความสมบูรณ์มากขึ้น	
2.3 ควรจะออกแบบระบบซอฟต์แวร์ ให้รองรับการเปลี่ยนแปลงและปรับปรุงในอนาคต	
2.4 มั่นใจว่าจากความนิยมและเทคโนโลยีในตอนนี้ การออกแบบให้ระบบซอฟต์แวร์เป็นไปตามความนิยมและตามเทคโนโลยีในตอนนี้ จะทำให้ระบบซอฟต์แวร์นั้นมีประสิทธิภาพ	
2.5 ระบบซอฟต์แวร์นี้ คิดว่าสามารถ (หรือไม่สามารถ) ออกแบบให้เหมือนกับอีกระบบซอฟต์แวร์ที่เป็นแบบที่นิยมใช้กันในปัจจุบันได้	
2.6 คิดว่าการออกแบบระบบซอฟต์แวร์ให้รองรับกับเทคโนโลยีที่นิยมใช้กันมากอยู่ในปัจจุบัน จะได้ใช้ประโยชน์ในอนาคต ถึงแม้ว่าจะยังไม่ได้รับร้องขอความต้องการดังกล่าวมาก็ตาม	
2.7 อื่น ๆ (ถ้ามี)	
(สำหรับผู้วิจัย) ค่าเฉลี่ยของ Availability	

ส่วนที่ 3.2.3: แบบสอบถามอคติประเภทที่ 3 (Confirmation) ในกระบวนการออกแบบ

ประเภทของอคติที่เกิดขึ้น/คำอธิบาย/ตัวอย่างเหตุการณ์	จำนวนครั้งที่เกิดเหตุการณ์ตัวอย่างขึ้นในการออกแบบซอฟต์แวร์*
3. Confirmation *การที่มนุษย์มักจะมีการยืนยันความคิด หรือความเชื่อของตน ซึ่งสามารถเกิดขึ้นได้ในหลายรูปแบบ เช่น เมื่อมนุษย์มีโอกาสที่จะหาข้อมูลใหม่ มนุษย์ก็จะมักที่จะหาข้อมูลที่ยืนยันความเชื่อของตนเองมากกว่าจะหาข้อมูลที่ค้านกับความเชื่อของตนเอง*	* ระบุค่าระดับตามตารางที่ 1 ของการเกิดเหตุการณ์ตัวอย่างขึ้นในการปฏิบัติงานด้านการออกแบบซอฟต์แวร์
(ตัวอย่าง) มั่นใจว่าการออกแบบแบบนี้ถูกต้อง เพราะได้รับการยืนยันจากผู้ใช้งานแล้วว่า ให้ออกแบบระบบซอฟต์แวร์ ว่าต้องให้เป็นไปตามแบบนี้	1
3.1 มั่นใจว่าการออกแบบแบบนี้ถูกต้อง เพราะได้รับการยืนยันจากผู้ใช้งานแล้วว่า ให้ออกแบบระบบซอฟต์แวร์ ว่าต้องให้เป็นไปตามแบบนี้	
3.2 ต้องเปลี่ยนแปลงและปรับปรุง ระบบซอฟต์แวร์ในส่วนตรงนี้	
3.3 จากประสบการณ์ คิดว่าระบบซอฟต์แวร์ ที่ต้องออกแบบไปเป็นแบบนี้ จะสามารถทำได้ (หรือทำไม่ได้)	
3.4 ควรจะเปลี่ยนแปลงและปรับปรุง ระบบซอฟต์แวร์ จากตรงนี้ไปเป็นแบบนี้ซึ่งคิดว่าทำได้ (หรือทำไม่ได้)	
3.5 อื่น ๆ (ถ้ามี)	
(สำหรับผู้วิจัย) ผลรวมของอคติทั้ง 3 ประเภท	

ส่วนที่ 3.2.4: แบบสอบถามการรายงานข้อผิดพลาดใน<u>กระบวนการหาออกแบบ</u>

โดยกำหนดให้ ในงานด้านการออกแบบซอฟต์แวร์ที่ท่านรับผิดชอบ มีเหตุการณ์ดังกล่าวข้างต้นเกิดขึ้น และเมื่อทำการส่งมอบซอฟต์แวร์ไปแล้ว เป็นระยะเวลา 1 เดือน มีจำนวนของข้อผิดพลาดที่เกิดขึ้นกับซอฟต์แวร์ที่มีการรายงานกลับมา **<u>คิดเป็นจำนวนกี่ครั้ง</u>**	_____ ครั้ง
โดย**<u>จำแนกจำนวนของข้อผิดพลาด</u>**จากข้อคำถามด้านบน ออกเป็นแต่ละระดับความรุนแรงดังนี้*	วิกฤต _____ มาก _____ ปานกลาง _____ ครั้ง น้อย _____ ละเว้นได้ _____
*คำนิยามระดับความรุนแรง	
1. ระดับวิกฤต (Critical) คือ ข้อผิดพลาดที่ส่งผลต่อความล้มเหลวของระบบซอฟต์แวร์ ระบบย่อย โปรแกรมหรือโมดูล โดยสมบูรณ์ 2. ระดับมาก (Major) คือ ข้อผิดพลาดที่ส่งผลต่อความล้มเหลวของตัวระบบซอฟต์แวร์ ระบบย่อย โปรแกรมหรือโมดูลเอง แต่ไม่ส่งผลต่อของส่วนงานอื่น ๆ ข้อผิดพลาดนี้สามารถยอมรับได้หากมีการประมวลผลด้วยวิธีอื่นที่จะทำให้ได้ผลลัพธ์ตามที่ต้องการ แต่ยังคงต้องแก้ไขข้อผิดพลาดนั้น 3. ระดับปานกลาง (Average) คือ ข้อผิดพลาดที่ไม่ได้ส่งผลต่อความล้มเหลว แต่ทำให้ระบบทำงานได้ไม่ถูกต้องสมบูรณ์ ผลลัพธ์ไม่สอดคล้องกัน หรือข้อผิดพลาดลดทอนการใช้งานระบบ 4. ระดับน้อย (Minor) คือ ข้อผิดพลาดที่ไม่ได้ก่อให้เกิดความล้มเหลว ไม่ได้ลดทอนการใช้งานระบบ และยังสามารถให้ผลลัพธ์ที่ต้องการจากการประมวลผลได้ด้วยการแก้ไขข้อผิดพลาดที่ไม่ยุ่งยาก 5. ระดับละเว้นได้ (Exception) คือ ข้อผิดพลาดที่เป็นผลมาจากความไม่สอดคล้องกันกับมาตรฐาน มีความเกี่ยวเนื่องกับคุณลักษณะภายนอกของระบบ เช่น ความสวยงาม สีสัน หรือเป็นความต้องการเพื่อปรับปรุงให้ดูดีขึ้น ข้อผิดพลาดในระดับนี้อาจจะผ่อนผันหรือแม้แต่ละเว้นการแก้ไขไปได้	

ขอขอบพระคุณเป็นอย่างสูง สำหรับความร่วมมือ

☺☺☺☺☺

ภาคผนวก ข

ผลการทดลองจากกลุ่มตัวอย่างก่อนที่จะนำมาสร้างแบบสอบถาม

ในการสร้างแบบสอบถามที่มีลักษณะเหตุการณ์ตัวอย่างของการมีอคตินั้น สร้างมาจากการรวบรวมลักษณะประโยคในรูปแบบต่าง ๆ จากกลุ่มตัวอย่างทดสอบที่เป็นนิสิตในสาขาวิศวกรรมซอฟต์แวร์ ซึ่งผลจากการสำรวจแบบสอบถามนี้ นอกจากจะได้มาซึ่งลักษณะเหตุการณ์ตัวอย่างแล้ว ยังได้ผลในการแสดงความสัมพันธ์ของอคติกับความเสียหายอีกด้วย โดยได้ศึกษาความสัมพันธ์ระหว่างการมีอคติกับความเสียหาย ด้วยวิธีการวิเคราะห์ความถดถอย กับความเสียหาย โดยใช้วิธีการเลือกตัวแปรอิสระเข้าสมการแบบ Stepwise โดยมีผลการทดลองดังนี้

ข.1 ผลการประมาณค่าความเสียหายของซอฟต์แวร์จากการมีอคติในกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์

ตารางที่ ข.1 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหาย

Variables Entered/Removed

Model	Variables Entered	Variables Removed	Method
1	Adjustment	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).
2	Confirmation	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).
3	Availability	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).

a. Dependent Variable: Y

ตารางที่ ข.2 ผลลัพธ์ Model Summary ของการประมาณความเสียหาย

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.704	.496	.478	1.27514	2.010
2	.773	.597	.567	1.16102	
3	.862a	.744	.714	0.94342	

a. Predictors: (Constant), Adjustment, Confirmation, Availability

b. Dependent Variable: Y

ตารางที่ ข.3 ผลลัพธ์ ANOVA ของการประมาณความเสียหาย

ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	44.814	1	44.814	27.561	.000a
	Residual	45.527	28	1.626		
	Total	90.342	29			
2	Regression	53.946	2	26.973	20.010	.000b
	Residual	36.395	27	1.348		
	Total	90.342	29			
3	Regression	67.200	3	22.400	25.167	.000c
	Residual	23.141	26	.890		
	Total	90.342	29			

a. Predictors: (Constant), Adjustment

b. Predictors: (Constant), Adjustment, Confirmation

c. Predictors: (Constant), Adjustment, Confirmation, Availability

d. Dependent Variable: Y

ตารางที่ ข.4 ผลลัพธ์ Coefficients ของการประมาณความเสียหาย

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	.332	.458		.725	.475
	Adjustment	.298	.065	.530	4.571	.000
	Confirmation	.394	.084	.625	4.689	.000
	Availability	-.252	.065	-.465	-3.859	.001

a. Dependent Variable: Y

ตารางที่ ข.5 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์

Variables	b	SE_b	Beta	t	p-value
Adjustment	.298	.065	.530	4.571	.000
Confirmation	.394	.084	.625	4.689	.000
Availability	-.252	.065	-.465	-3.859	.001
Constant .332; SE_{est} = ±0.94342 R^2=.744; F=25.167; p-value<.05					

จากตาราง ข.5 จะเห็นว่าตัวแปรอิสระนั่นคืออคติประเภท Adjustmentm, Confirmation, Availability มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อยู่ในระดับปานกลางค่อนข้างมาก โดยมี R^2 เป็น .744 นั่นสามารถร่วมกันพยากรณ์ความเสียหายของซอฟต์แวร์ได้ร้อยละ 74.4 โดยมีความคลาดเคลื่อนมาตรฐานในการพยากรณ์เท่ากับ ±0.94342 และเมื่อพิจารณาค่าสัมประสิทธิ์การถดถอยของตัวพยากรณ์ พบว่าตัวแปรอิสระสามารถพยากรณ์ความความเสียหายของซอฟต์แวร์ได้ (p-value < .05) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05

ตรวจสอบเงื่อนไขจากค่าสถิติทดสอบ Durbin-Watson = 2.010 อยู่ในช่วง 1.5-2.5 สรุปได้ว่าค่าคลาดเคลื่อนเป็นอิสระต่อกัน เป็นไปตามเงื่อนไขของการวิเคราะห์ถดถอย ดังนั้นจึงสามารถสรุปได้ว่า

มีอคติอย่างน้อยหนึ่งประเภทที่มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์ และสำหรับสมการพยากรณ์ความเสียหายของซอฟต์แวร์เมื่อนำตัวแปรอิสระทุกตัวเข้าสมการในรูปคะแนนดิบ จะเป็นดังนี้

$$\hat{Y}_{req} = 0.332 + 0.298adj + 0.394con - 0.252ava \quad (4.2)$$

โดยผลจากการทดลองนี้พบว่า อคติประเภท Anchoring และ Adjustment นั้น จัดเป็นอคติประเภทเดียวกันซึ่งมีความต่อเนื่องกันอย่างมีนัยสำคัญทางสถิติ .05 โดยจะเห็นได้ว่าจุดที่ตัดกันของอคติทั้งสองประเภทนั้น เป็น .000 ซึ่งหมายถึงมีความสัมพันธ์กันอย่างมาก โดยสามารถแสดงได้ดังตาราง ข.6

ตารางที่ ข.6 ผลลัพธ์ Correlations ของอคติแต่ละประเภท

Correlations

		Y	Anc	Adj	Ava	Con
Sig. (1-tailed)	Y	.	.001	.000	.378	.000
	Anc	.001	.	.000	.056	.013
	Adj	.000	.000	.	.042	.002
	Ava	.378	.056	.042	.	.001
	Con	.000	.013	.002	.001	.

ซึ่งตรงกับคำนิยามอคติและความหมายของอคติประเภทในเชิงจิตวิทยาที่รวมอคติสองประเภทนี้เข้าไว้ด้วยกันนี้ จึงทำให้การทดลองใหม่ในงานวิจัยนี้ทำการรวมอคติประเภท Anchoring และ Adjustment ไว้ด้วยกัน แล้วเรียกแทนด้วย Anchoring and adjustment ซึ่งได้ทำการทดลองยืนยันแนวความคิดนี้อีกครั้งในการวิเคราะห์ในกระบวนการออกแบบซอฟต์แวร์อีกด้วย ซึ่งให้ผลออกมาเป็นไปทิศทางนี้เช่นกัน

ข.2 ผลการประมาณค่าความเสียหายของซอฟต์แวร์จากการมีอคติในกระบวนการออกแบบซอฟต์แวร์

ตารางที่ ข.7 ผลลัพธ์ Varialbes Entered/Removed ของการประมาณความเสียหายของซอฟต์แวร์

Variables Entered/Removed

Model	Variables Entered	Variables Removed	Method
1	Confirmation	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).
2	Adjustment	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).

a. Dependent Variable: Y

ตารางที่ ข.8 ผลลัพธ์ Model Summary ของการประมาณความเสียหายของซอฟต์แวร์

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.795	.633	.619	1.49279	1.639
2	.830a	.688	.665	1.39987	

a. Predictors: (Constant), Confirmation, Adjustment

b. Dependent Variable: Y

ตารางที่ ข.9 ผลลัพธ์ ANOVA ของการประมาณความเสียหายของซอฟต์แวร์

ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	107.404	1	107.404	48.197	.000a
	Residual	62.396	28	2.228		
	Total	169.800	29			
2	Regression	116.890	2	58.445	29.825	.000b
	Residual	52.910	27	1.960		
	Total	169.800	29			

a. Predictors: (Constant), Confirmation

b. Predictors: (Constant), Confirmation, Adjustment

c. Dependent Variable: Y

ตารางที่ ข.10 ผลลัพธ์ Coefficients ของการประมาณความเสียหายของซอฟต์แวร์

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	-1.833	.839		-2.185	.038
	Confirmation	.454	.092	.638	4.942	.000
	Adjustment	.271	.123	.284	2.200	.037

a. Dependent Variable: Y

ตารางที่ ข.11 ผลการวิเคราะห์การถดถอยเพื่อการพยากรณ์

Variables	b	SEb	Beta	t	p-value
Confirmation	.454	.092	.638	4.942	.000
Adjustment	.271	.123	.284	2.200	.037
Constant -1.833; SEest = ±1.39987 R2=.688; F=29.825; p-value<.05					

จากตาราง ข.11 จะเห็นว่าตัวแปรอิสระนั่นคืออคติประเภท Confirmation, Adjustmentm มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์อยู่ในระดับปานกลาง โดยมี R^2 เป็น .688 นั่นสามารถร่วมกันพยากรณ์ความเสียหายของซอฟต์แวร์ได้ร้อยละ 68.8 โดยมีความคลาดเคลื่อนมาตรฐานในการพยากรณ์เท่ากับ ±1.39987 และเมื่อพิจารณาค่าสัมประสิทธิ์การถดถอยของตัวพยากรณ์ พบว่าตัวแปรอิสระสามารถพยากรณ์ความความเสียหายของซอฟต์แวร์ได้ (p-value < .05) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05

ตรวจสอบเงื่อนไขจากค่าสถิติทดสอบ Durbin-Watson = 1.639 อยู่ในช่วง 1.5-2.5 สรุปได้ว่าค่าคลาดเคลื่อนเป็นอิสระต่อกัน เป็นไปตามเงื่อนไขของการวิเคราะห์ถดถอย ดังนั้นจึงสามารถสรุปได้ว่ามีอคติอย่างน้อยหนึ่งประเภทที่มีความสัมพันธ์กับความเสียหายของซอฟต์แวร์ และสำหรับสมการพยากรณ์ความเสียหายของซอฟต์แวร์เมื่อนำตัวแปรอิสระทุกตัวเข้าสมการในรูปคะแนนดิบ จะเป็นดังนี้

$$\hat{Y}_{des} = -1.833 + 0.454con + 0.271adj \quad (4.2)$$

โดยผลจากการทดลองนี้ก็พบว่า อคติประเภท Anchoring และ Adjustment นั้น จัดเป็นอคติประเภทเดียวกันซึ่งมีความต่อเนื่องกันอย่างมีนัยสำคัญทางสถิติ .05 เช่นกันกับกระบวนการรวบรวมและกำหนดความต้องการซอฟต์แวร์ โดยจะเห็นได้ว่าจุดที่ตัดกันของอคติทั้งสองประเภทนั้นเป็น .000 ซึ่งหมายถึงมีความสัมพันธ์กันอย่างมาก โดยสามารถแสดงได้ดังตาราง ข.12

ตารางที่ ข.12 ผลลัพธ์ Correlations ของอคติแต่ละประเภท

Correlations

		Y	Anc	Adj	Ava	Con
Sig. (1-tailed)	Y	.	.000	.000	.000	.000
	Anc	.000	.	.000	.001	.000
	Adj	.000	.000	.	.000	.001
	Ava	.000	.001	.000	.	.000
	Con	.000	.000	.001	.000	.

ซึ่งตรงกับคำนิยามอคติและความหมายของอคติประเภทในเชิงจิตวิทยาที่รวมอคติสองประเภทนี้เข้าไว้ด้วยกันนี้ จึงทำให้การทดลองใหม่ในงานวิจัยนี้ทำการรวมอคติประเภท Anchoring และ Adjustment ไว้ด้วยกัน แล้วเรียกแทนด้วย Anchoring and adjustment แต่สำหรับอคติประเภทอื่นที่มีความสัมพันธ์กันสูงเช่นกัน แต่ด้วยเงื่อนไขของวิธีการ Stepwise ก็ยังสามารถนำเข้าวิเคราะห์ความถดถอยได้

ประวัติผู้เขียนวิทยานิพนธ์

นายนพธนิษฐ์ โชติสาร เกิดเมื่อวันอังคารที่ 12 สิงหาคม พุทธศักราช 2529 ที่จังหวัดสระบุรี ในปีการศึกษา 2551 สำเร็จการศึกษาระดับปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาศาสตร์คอมพิวเตอร์ ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์ ในปีการศึกษา 2553 สำเร็จการศึกษาระดับปริญญารัฐศาสตรบัณฑิต สาขาวิชารัฐศาสตร์ กลุ่มวิชาความสัมพันธ์ระหว่างประเทศ คณะรัฐศาสตร์ มหาวิทยาลัยรามคำแหง และในปีการศึกษา 2554 เข้าศึกษาต่อในระดับปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย