

หัวข้อวิทยานิพนธ์

แบบจำลองการจัดลำดับความต้องการก่อนหลังสำหรับความต้องการขนาดใหญ่

ผู้เขียน

นางสาวเจนจิรา ใจมั่ง

ปริญญา

วิทยาศาสตรมหาบัณฑิต (วิศวกรรมซอฟต์แวร์)

อาจารย์ที่ปรึกษา

อาจารย์ ดร.ภราดร สุริย์พงษ์

บทคัดย่อ

ในโครงการพัฒนาซอฟต์แวร์ การจัดลำดับความสำคัญของความต้องการก่อนนำไปสู่ขั้นตอนอื่นๆ ของการพัฒนาซอฟต์แวร์เป็นสิ่งที่ท้าทายสำหรับนักพัฒนาซอฟต์แวร์ เนื่องจากความต้องการเป็นวัตถุดิบพื้นฐานของกระบวนการที่เหลือของการพัฒนาซอฟต์แวร์ ดังนั้นแล้วหากกระบวนการความต้องการหลักผิด จะทำให้การพัฒนาซอฟต์แวร์ผิดทิศทาง ไม่สามารถตอบสนองการใช้งานของผู้ใช้งาน ทั้งนี้เนื่องจากความพึงพอใจสูงสุดของผู้ใช้งานคือตัวชี้วัดคุณภาพของซอฟต์แวร์ ดังนั้นซอฟต์แวร์ที่ผลิตขึ้นมาจึงมีคุณภาพต่ำ ดังนั้นเพื่อสร้างความพึงพอใจแก่ลูกค้าสูงสุด ภายใต้เงื่อนไขด้านเวลางบประมาณ และทรัพยากรที่จำกัด นักพัฒนาซอฟต์แวร์จึงต้องทำการจัดลำดับความต้องการก่อนหลัง โดยการวิเคราะห์หาความต้องการหลักเพื่อใช้ในการลำดับการพัฒนาความต้องการต่อไป

ถึงแม้ว่างานวิจัยหลายชิ้นได้ทำการศึกษาเกี่ยวกับการจัดลำดับความต้องการก่อนหลัง แต่สำหรับความต้องการขนาดใหญ่ ซึ่งหมายถึงความต้องการมากกว่า 100 ความต้องการขึ้นไป ยังเป็นประเด็นสำคัญที่ถูกอภิปรายถึงความยุ่งยากซับซ้อนในปัจจุบัน ดังนั้นวิทยานิพนธ์ฉบับนี้จึงทำการศึกษาการออกแบบแบบจำลองการจัดลำดับความต้องการก่อนหลังที่สามารถจัดการกับความต้องการขนาดใหญ่ได้ โดยที่แบบจำลองดังกล่าวสามารถลดค่าความพยายามและเพิ่มความถูกต้องของการเรียงลำดับความต้องการได้ โดยประกอบด้วยกระบวนการศึกษาทั้งหมด 4 กระบวนการได้แก่ (1) วิเคราะห์และนำเสนอแบบจำลองการจัดลำดับก่อนหลังสำหรับความต้องการขนาดใหญ่ (2) การศึกษาขั้นตอนการจัดลำดับความต้องการก่อนหลังแบบเดิม (3) ทดสอบเพื่อเปรียบเทียบการนำแบบจำลองไปใช้ และ (4) ประเมินผลหลังจากการนำแบบจำลองไปใช้

ในกระบวนการที่ 1 ทำการทบทวนวรรณกรรมเพื่อวิเคราะห์และนำเสนอแบบจำลอง โดยที่แบบจำลองที่ได้ประกอบด้วย 5 ขั้นตอนหลักๆ ได้แก่ (1) นำความต้องการทั้งหมดมาแบ่งกลุ่มความต้องการด้วยเทคนิค Priority Grouping (2) ทำการให้นำนักปัจจัยที่มีผลต่อการจัดลำดับความต้องการ ก่อนหลังจากนั้น (3) นำความต้องการมาให้คะแนนตามปัจจัยต่างๆตามหลักการของ Weigers's Method หลังจากนั้น (4) นำค่าคะแนนที่ได้มาคำนวณค่าความสำคัญของความต้องการ (5) นำผลคะแนนที่ได้มาทำการเรียงลำดับโดยใช้เทคนิค Binary Search Tree

ในกระบวนการที่ 2 ทำการสัมภาษณ์เพื่อคัดเลือกโครงการที่ประกอบด้วยความต้องการขนาดใหญ่ เพื่อศึกษาการจัดลำดับก่อนหลังแบบเดิมและทำรายการความต้องการ

ในกระบวนการที่ 3 ทำการเปรียบเทียบแบบจำลองที่นำเสนอกับวิธีการเดิม โดยการนำแบบจำลองไปทดลองจัดลำดับความต้องการกับโครงการทั้งหมด 2 โครงการ ได้แก่ โครงการที่ 1 คือโครงการพัฒนาซอฟต์แวร์สำหรับการดูแลธุรกิจบนโทรศัพท์ที่ประกอบด้วยความต้องการทั้งหมด 118 ความต้องการ และโครงการที่ 2 คือโครงการพัฒนาซอฟต์แวร์สำหรับประเมินพนักงานบนเว็บไซต์ ซึ่งประกอบด้วยความต้องการทั้งหมด 102 ความต้องการ โดยขั้นตอนนี้มีวัตถุประสงค์เพื่อวัดค่าความพยายามในการจัดเรียงลำดับความต้องการ

กระบวนการสุดท้ายทำการประเมินค่าความถูกต้องของการจัดลำดับความต้องการหลังจากการทดลองนำแบบจำลองไปใช้ประมาณ 1 สัปดาห์ ผู้ถูกสัมภาษณ์ทั้งหมดจะต้องเลือกลำดับที่ถูกต้องที่สุดของความต้องการโดยไม่ทราบว่าที่ลำดับนั้นได้จากวิธีการใดหรือที่เรียกว่าหลักการ Blind Test

ผลจากการศึกษาพบว่าแบบจำลองที่นำเสนอสามารถช่วยลดค่าความพยายามและเพิ่มความถูกต้องในการจัดลำดับก่อนหลัง นอกจากนั้นแบบจำลองนี้ยังสามารถจัดการกับความต้องการขนาดใหญ่ได้อีกด้วย จากผลการทดลองพบว่าค่าความพยายามในทุกโครงการลดลงมากกว่าร้อยละ 50 โดยในโครงการแรกไม่เพียงค่าความพยายามลดลงถึงร้อยละ 53.6 แต่ร้อยละ 63.33 ของผู้ที่ทำการทดลองในโครงการแรกเชื่อว่าผลลัพธ์จากแบบจำลองมีความถูกต้องมากกว่าวิธีการเดิม ในส่วนโครงการที่สองนั้น ร้อยละ 33.33 ของผู้ที่ทำการทดลองเชื่อว่าผลลัพธ์จากแบบจำลองมีความถูกต้องมากกว่าวิธีการแบบเดิม อีกทั้งค่าความพยายามในการเรียงลำดับความต้องการของโครงการที่สองนั้นลดลงถึงร้อยละ 59.16 อีกด้วย

Thesis Title	Requirements Prioritization Model for Large-Scale Requirements
Author	Ms. Jenjira Jaimunk
Degree	Master of Science (Software Engineering)
Advisor	Dr. Pradorn Sureephong

ABSTRACT

In a software development project, rearranging order of requirements before moving forward to the implementation phase is a challenging for software developers. Because requirement is the basic material for the rest of the software development process, therefore, if the core requirements are mistakenly identified, the software is developed in the wrong direction. It cannot meet the user expectation. Due to the user satisfaction is a measure of the quality of the software, so the software quality is dropped. In order to maximize the user satisfaction under constrain of budget, time and resources, the developers need to prioritize the requirement to identify the core requirements used in project planning.

Although, there are many researchers studied on requirements prioritization, however the large scale of requirement- at least 100 requirements or more is one of the most significant current discussions in software engineering. This thesis investigates on designing a requirement prioritizing model to handle the large scale of requirements. This model aims to minimize effort and maximize the accuracy of priority sequence. There are 4 processes to study, including (1) To analyze and propose model, (2) To elicit a traditional prioritizing method, (3) To compare proposed method with traditional method and (4) To evaluate the result.

In the first process, the literature review was conducted, and then designed the model. The proposed model composes of 5 steps. They are Priority grouping, Criteria weighting, Estimate requirement by criteria using Weigers's method, calculate priority and sorting by Binary Search Tree.

The second process aims to investigate on the traditional method and gathering the requirement list by interviewing the case study.

The third process was implemented to compare the proposed model with the traditional method by designed experiment. The experiment is conducted in 2 software development projects. The first project is mobile application development with 118 requirements and the second project is web application development with 102 requirements. The expected result is effort evaluation.

In the last process, One week after executed the experiment, the accuracy evaluation will be conducted. This experiment focuses on assessing the accuracy of priority sequence of requirements. All interviewee have to choose the most accurate sequence of requirements without realizing on which sequence is from which method (Blind Test).

These findings further support the idea that the proposed model can minimize the consumed effort and maximize accuracy of the requirement priority sequence. Moreover, this integrated model can handle the large number of requirements. Based on experiment result, it is interesting to note that effort in all cases have been reduced more than 50%. Regarding the first project, not only the effort is reduced by 53.6%, 63.33% of interviewees from the first project believe that the proposed model provides more accuracy than the traditional one. According to the second project, 33.33% of interviewees from the second project believe that the proposed model provides more accuracy than the traditional one. Moreover, 59.16% of consumed effort for the second project is reduced dramatically.

Copyright© by Chiang Mai University
All rights reserved