الجامعة الإسلامية العالمية ماليزيا
INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA
(Company No. 101067-P)
Garden of Knowledge and Virtue

TAWHIDIC EPISTEMOLOGY LEADING THE WAY
UMMATIC EXCELLENCE LEADING THE WORLD
KHALĪFAH • AMĀNAH • IQRA' • RAḤMATAN LIL-ĀLAMĪN

KICT
KULLIYYAH OF INFORMATION AND COMMUNICATION TECHNOLOGY

Advancement in ICT: Exploring Innovative Solutions (AdICT) Series 3/2025

Editors
Elin Eliana Abdul Rahim
Noor Azura Zakaria
Dini Oktarina Dwi Handayani
Ahmad Fatzilah Misman

KICT Publishing

ADVANCEMENT IN ICT: EXPLORING INNOVATIVE SOLUTIONS (AdICT) Series 3/2025

Editors
Noor Azura Zakaria
Dini Oktarina Dwi Handayani
Elin Eliana Abdul Rahim
Ahmad Fatzilah Misman

KICT Publishing

ADVANCEMENT IN ICT: EXPLORING INNOVATIVE SOLUTIONS
(AdICT) Series 3/2025

First published 2024
Third edition 2025

Published by

KICT Publishing
International Islamic University Malaysia
53100 Kuala Lumpur, Selangor, Malaysia

e ISBN 978-629-99388-4-2

9 786299 938842

Cataloguing-in-Publication Data

Perpustakaan Negara Malaysia

A catalogue record for this book is available from the National Library of Malaysia

eISBN 978-629-99388-4-2

Kulliyyah of Information and Communication Technology (KICT)

Preface

It is our great pleasure to present Advancement in ICT: Exploring Innovative Solutions (AdICT), Series 3/2025. This publication brings together diverse and forward-looking ideas in ICT, showcasing innovative approaches that not only address current challenges but also shape future opportunities. The contributions in this volume reflect the dedication, creativity and perseverance of KICT lecturers and students, whose hard work continues to push the boundaries of knowledge and application. Their commitment highlights the spirit of innovation that drives both academic inquiry and practical solutions in the ever-evolving field of ICT.

We would like to extend our sincere appreciation to all authors, reviewers, editors and the organising team for their invaluable efforts in making this series possible. It is our hope that this volume will serve as both an inspiration and a useful reference for scholars, students and professionals. By sharing these innovative perspectives and solutions, we aim to contribute meaningfully to the ongoing discourse in ICT, offering fresh insights and encouraging further exploration in advancing the discipline for the benefit of society.

Editors
Elin Eliana Abdul Rahim
Noor Azura Zakaria
Dini Oktarina Dwi Handayani
Ahmad Fatzilah Misman

TABLE OF CONTENTS

A Web-based Solution for Value-Based Software Process Tailoring

Muhammad Amirul Ashraaf Pakasa
Department of Computer Science
International Islamic University Malaysia
Kuala Lumpur, Malaysia
4ashraaf@gmail.com

Mohd Izzat Ameir Mat Zain
Department of Computer Science
International Islamic University Malaysia
Kuala Lumpur, Malaysia
zamzameir@gmail.com

Noor Azura Zakaria
Department of Computer Science
International Islamic University Malaysia
Kuala Lumpur, Malaysia
azurazakaria@iium.edu.my

***Abstract*—Developing high-quality software products is crucial for the software industry to remain competitive. The success of a software development project depends significantly on a well-structured software process, which improves productivity. Software process tailoring is the process of modifying and specifying the terms of a general framework to create a version suited to a specific, less general environment. Software process tailoring enables software teams to customize and optimize development processes to meet the unique requirements of each project. This study introduces a web-based solution for value-driven software process tailoring, incorporating a value-based approach that assesses cost, benefits, and overall impact on the organization. By integrating value considerations, this tool aids in strategic decision-making, ensuring efficient allocation of time, cost, effort, and resources. The proposed solution aims to enhance productivity, improve process adaptability, and support organizations in achieving optimal software development outcomes.**

Keywords— Software Process, Value-Based Software Process Tailoring Framework, Web-Based Tool

I. INTRODUCTION

Software process tailoring requires prior understanding to decide the tailoring operation. However, the current practices use an ad hoc approach and lack of guidelines to conduct the tailoring process. The software process tailoring tools also do not offer the assistant on decision-making process since there are many factors involved in a software project. The objectives are to investigate requirements, design, develop and evaluate the web-based tool for Value-Based Software Process Tailoring. The value-based software process tailoring provides clear understanding to conduct tailoring process since it aligns the value with the decision-making process and avoid inconsistency and insufficiency of quality information.

II. LITERATURE REVIEW

A. Software Development Model

Software process tailoring is a part of software development which is the process of conceiving, specifying, designing, programming, documenting, testing, and bug fixing involved in creating and maintaining applications, frameworks, or other software components. The process of writing and maintaining the source code are parts of software development. Software development covers all that is involved between the desired software conception until the final manifestation of the software, sometimes even in a planned and structured process. Thus, research, new development, prototyping, modification, reuse, re-engineering, maintenance, or any other activities that result in software products may be included in software development.

To meet the needs of a specific client or business and to meet the needs of some potential users or for personal use are the common purposes of developing software product. The development process needs to be integrated with the development of the controlled physical product for the development of embedded software that is used for controlling consumer products. Often developed separately, system software underlies applications and the programming process itself. Software development life cycle models, methodologies, processes, or models are the approaches to software project management. Among the models are Waterfall model, V-model, Spiral model and the most used model in the market, Agile model. It is crucial to choose the right model for developing the software product or application because the development and testing processes are carried out based on the decided model.

Testing starts only after the development is completed based on the Waterfall model - a very old model. Because there are many defects and failures reported at the end, fixing these issues is a must. That is why Agile model is preferred because after every sprint, the customer has a demo-able feature for them to decide whether the features are satisfying their needs or not. Apart from that, many software companies used V-model in their product. Verification and validation made up V-model. The developer's life cycle and tester's life cycle are mapped to each other in V-model as the testing is done alongside the development.

Talking about software process phase, there are six phases in software which are requirements gathering and analysis, design, implementation or coding, testing, deployment and lastly maintenance. In requirements gathering, the main focus are the managers and stake holders. They are basically being asked with some general question about the project. After all requirements are validated, next comes the design phase. This phase helps in specifying and defining overall system architecture. Next, the longest phase of software development which is the implementation phase. After the software is developed, it will be tested against the requirements to make sure the product is actually satisfying the needs. After successful tested, the product is delivered to the customer for their use. The actual problems come up when the customer starts using the developed system and that needs to be solved periodically. It is the process of maintenance where the care is taken for the developed product from time to time.

B. *Software Process Tailoring*

Software process tailoring refers to the activity of tuning a standardized project to meet the needs of specific project. Zakaria et al [1] stated that "One size could not fit all" is synonym with software development process since every project is unique and different with one and another according to its project context. Therefore, tailoring process is a need in software development.

Software Process Tailoring Tools: There are various tools used in software process tailoring with different input factors, steps, operations included and methods. Lorenz et. al. [2] proposed a software process tailoring tool using meta-model and AHP technique where it uses meta-model and software process line approach, this tool systematically tailors the software process according to the project context and risk of the project. The input factor used are context of the project and the project risks. The steps taken starts with the definition of the project context, followed by the tailoring requirements selection and lastly, process architecture selection. However, the operations included in this tool was not stated. Another approach by Hurtado et. al. [3] which is somewhat similar to the previous tool as it also uses meta-model as the technique, in addition to Atlas Transformation Language (ATL). The rules defined from the meta-models are transformed by using this technique in order to tailor the software process and it only include the project context for the input factor. Developing the context model is the first step taken for this tool to work before it proceeds to the software process model development, defining rules, transforming rules in ATL and adapted software process model suggestion. The tailoring operation of this tool was not stated.

Rui et. al. [4] objective is to preserve the dependency of the software process elements during tailoring, thus a tool that apply unified lifecycle template is used. The input factor is the project characteristics and the steps for the tailoring process were not stated. The tailoring operation includes add, delete, drop, remove, replace, modify and merge. To ensure the consistency during the tailoring activity, Pereira et. al. [5] used a meta-model and well-formed rules. The input factor for this tool was not stated. The tailoring steps for this tool are to design extended RUP meta model, define relationship and dependencies, determine tailoring operation, and develop tailoring well-formed rules. Add and delete are the 2 tailoring operations. Next, Mikulenas and Kapocius [6] developed a tool for determining appropriate agile practices for adaptation techniques with input factors agile practices, value for each practice and the cost for each practice. The steps taken include identifying agile practices using RDP, value and cost for each practice using AHP prioritization and cost and value mapping. The tailoring operations are not applicable for this tool and the tool developed by Gonzalez-Perez et. al. [7]. This tool selects an appropriate set of method fragments for method construction technique. The product and project characteristics and organization are the input factors of the tool and the initial process towards the tailoring activity which is the selection of the appropriate method fragments is the focus of this work.

Apart from that, a software process tailoring tool that uses an algorithm to tailor the RUP based on software reliability characteristics were proposed by Li and Mao [8] where it uses software reliability as the input factor. The technique used is fuzzy comprehensive evaluation where it develops RUP extension and then scores the relevancy for the RUP elements. The tailoring operations were not stated. Tool that uses neural network technique as proposed by Park et. al. [9] is a tool that applies tailoring decisions that learn using a neural network from historical data or previous project. The project environment becomes the input factor of the tool and the steps involve process filtering, process reconfigurations and process feedback with adoption and rejection tailoring operations. Park and Bae [10] developed a tool based on the similarity between the tailoring case and the current process tailoring, this tool assists the practitioner to decide the elements that need to be tailored. The technique used is case-based reasoning and process slicing with environmental factors as the input factors. The tailoring steps are to construct the tailoring cases followed by the calculation of the similarity. The three tailoring operations are basically add, modify and delete. Another tool that uses case-based reasoning technique was developed by Kang et. al. [11] where it is able to find the structural similarity of the software process with the previous projects. The input factor is the project context and the steps taken include the process and tailoring rule structure. Add, remove and replace are the tailoring operations for this tool.

Henninger and Baumgarten [12] applied the same case-based reasoning technique for their tool. It stores knowledge of the employee in the software process tailoring using a knowledge-based tool as a knowledge repository with project characteristics being the input factor. Eliciting project characteristics and matching project characteristics with software development methodology requirements are the steps taken in this tool together with a single operation which is add. He et. al. [13] proposed a tool that systematically tailors the software process based on knowledge reuse by using decision model and algorithm of this technique. The technique is namely meta-model and conflict resolution. It uses the project characteristics as the input factor. The steps included are to identify the project characteristics, selecting process tailoring packages, applying process tailoring packages to process framework, conflict check, consistency check and process instance creation. Contribute, suppress and replace are the tool's three tailoring operations. By capturing experience and knowledge of the experienced employee in performing the tailoring using a knowledge-based tool, Xu and Ramesh [14] proposed a tool that capture the project characteristics and organization characteristics as the input factors. The steps for the tailoring and the technique are, however, not stated. Add, delete and modify are the three tailoring operations of this tool. Lastly, Yoon et. al. [15] focused on addressing tailoring on the selected standard and verify the tailored process with correctness and consistency with the standard, this tool combines case-based reasoning and formal method. The technique used is Activity Artefact Graph (AAG) technique by having process module as the input factor. It focused more on the dependency of the process module. There are 4 tailoring operations for this tool namely add, delete, split and merge.

Although different techniques are used for the development of software process tailoring tools, they all focused on the same goal. By having these existing tools as reference, a value-based software process tailoring tool is the aim of this project as it has the advantages over the existing tools. The implementation of this tool will be further explained in the next chapter.

III. METHODOLOGY

A. Design

a) Value-Based Software Process Tailoring Framework (VBSPTF): Reducing decision making subjectivity is one of many advantages of using Value-Based Software Process Tailoring Framework (VBSPTF). The decision-making process includes three main elements – 'adopt', 'modify' and 'delete', which can help the user to conduct the tailoring process. A website which will assist users to enhance their software process was developed that will be structured based on the value-based factors classification. Using the VBSPTF as shown in Figure 1.

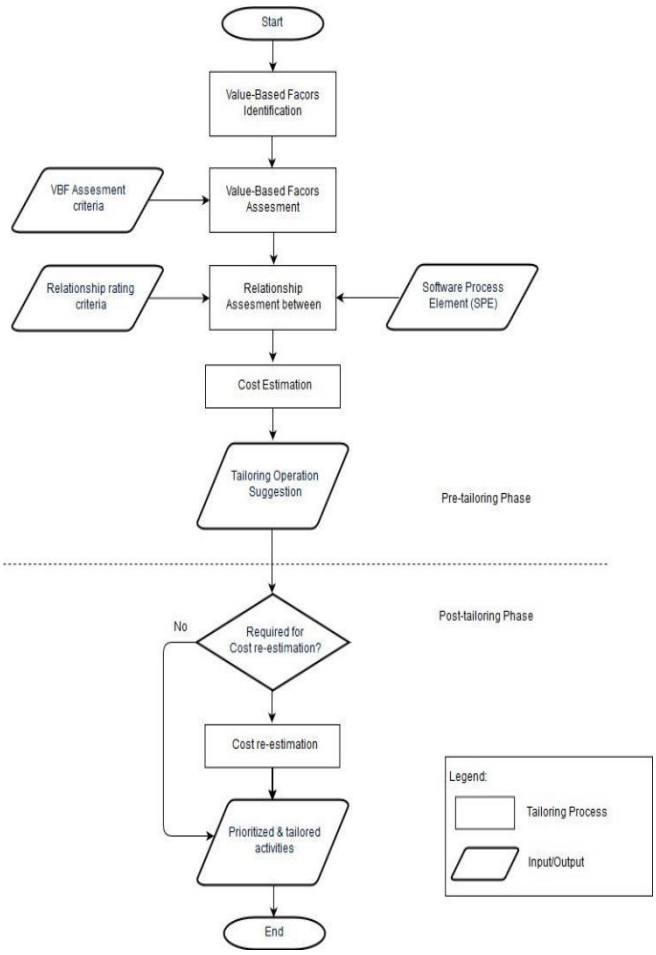

Fig. 1. Process for Value-Based Software Process Tailoring (VBSPTF)

The project manager needs to identify the value-based factors first. There is a total of 28 value-based factors (VBF) that need to be considered during VBF identification although this guideline was not meant to be a limited list [10]. Next, the value-based factors need to be prioritized before establishing relationship assessment, performing cost estimation for each activity and identifying tailoring operation. Lastly, if the software process element was included in the modify region, cost re-estimation need to be done. Figure 2 shows the architecture design for the development.

B. Architecture Design

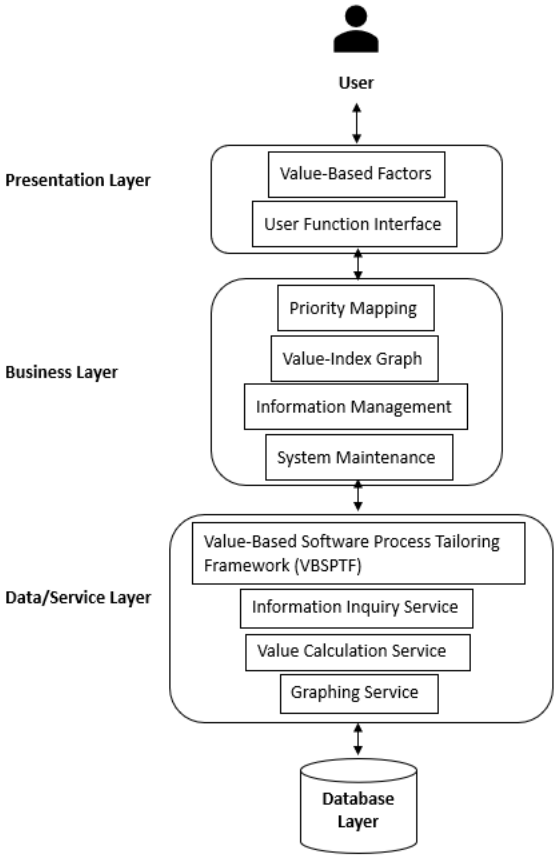

Fig. 2. Architecture Design

C. Design of User Interface

The user interface design for the main module of the web-based tool is based on the design by Lorenz et. al. [8] as shown in Figure 3. This design allows the sets of steps to be organized. Figure 4 shows the design of the interface implemented in the web-based tool. A menu showing the module for tailoring process will be shown on the top side of the page that will include the options for Value-Based Factors Classification together with its factors, Prioritization, Relationship Assessment and Cost Estimation.

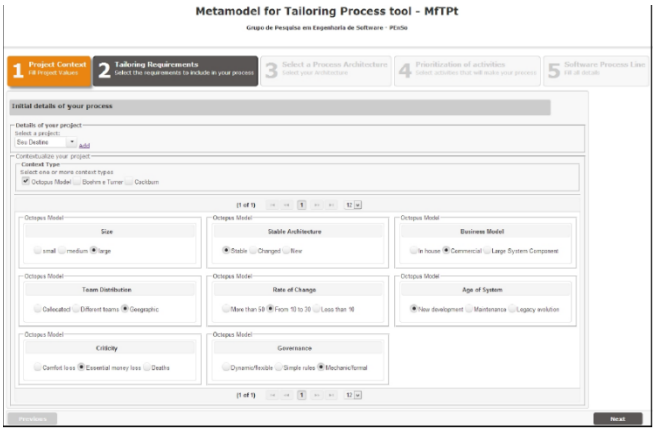

Fig. 3. Main interface by [8]

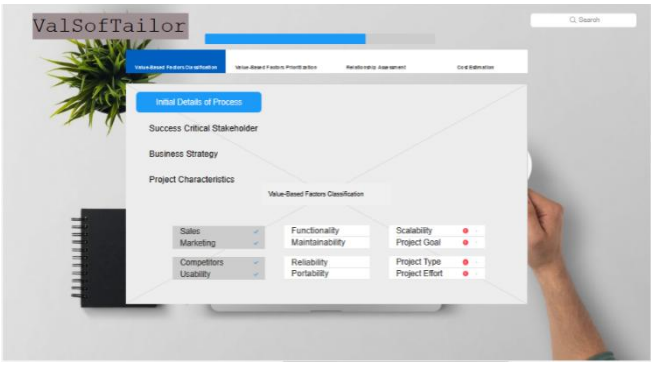

Fig. 4. Modules in Value-Based Software Process Tailoring

IV. RESULT

A prototype has been developed that consists of the main page, login page, sign up page and the tailoring process module page. The prototype was developed using HTML, CSS, JavaScript together with the templates by Bootstrap. Figure 5 shows the main web page. Apart from that, a simple database was established using MySQL that stores the information of users signed up. Figure 6 shows the sign-up web page and Figure 7 shows the member login web page. For the tailoring process module page, simple table displaying the important elements in the tailoring process with the function to change the prioritization of the Value-Based Factors. Figure 8 shows the table for the tailoring process module which later will be improved with the design shown in Figure 4. For the purpose of visualization, Google Charts was used to display the Priority Mapping Graph which is shown in Figure 9. This mapping graph will show which testing activity will be adopted, modified or dropped depending on the strength of relationship and cost.

Fig. 5. Main Web Page

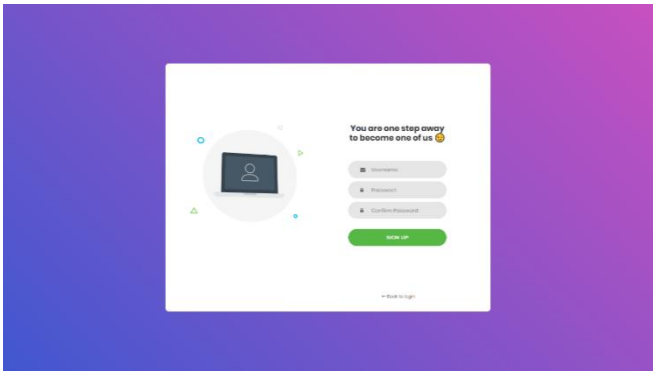

Fig. 6. Sign Up Page

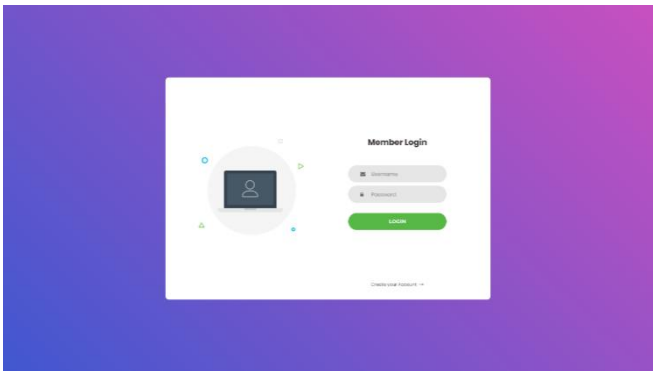

Fig. 7. Member Login Page

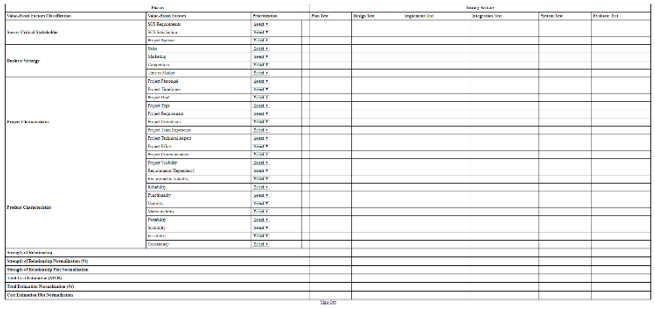

Fig. 8. Tailoring Process Module

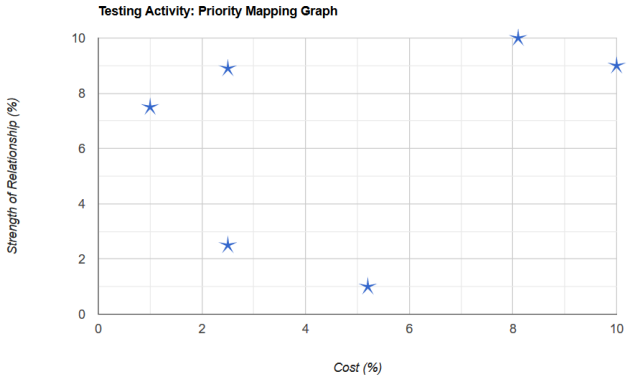

Fig. 9. Priority Mapping Graph

V. CONCLUSION METHODOLOGY

Value-Based Factors Identification, Value-Based Factors Assessment, Relating the Assessment between Value-Based Factors and Software Process Element, Cost Estimation, Tailoring Operation Suggestion and Cost Re-estimation are the steps in implementing Value-Based Software Process Tailoring Framework. In other words, to make sure the testing activities have high strength of relationship and low cost, the software process is tailored based on the value assessment.

The advantage of having values as the main elements of software process tailoring is that it helps in decision making – the metrics can be easily compared and quantified. The framework proposed was successfully implemented in the web-based tool developed and it is meant to ease the task of project managers, software development teams and practitioners while performing tailoring activities and help them to achieve a good quality software product. The web-

based tool developed will be easily accessed with a created account. The tool can help to suggest suitable value-based factors classification with prioritization and calculate the strength of relationship and the cost. This will enable the users to decide on what activity needs to be added, modified or deleted according to the result produced by the tool.

REFERENCES

[1] N. A. Zakaria, S. Ibrahim and M. N. Mahrin, "A proposed value-based software process tailoring framework," *2015 9th Malaysian Software Engineering Conference (MySEC)*, Kuala Lumpur, 2015, pp. 149-153.

[2] Lorenz, W. G., Brasil, M. B., Fontoura, L. M., and Vaz Pereira, G. (2014). Activity-Based Software Process Lines Tailoring. *International Journal of Software Engineering and Knowledge Engineering, 24*(09), 1357-1381

[3] Hurtado, J. A., Bastarrica, M. C., Ochoa, S. F., and Simmonds, J. (2013). MDE software process lines in small companies. *Journal of Systems and Software*, 86(5), 1153-1171.

[4] Rui, H., Hao, W., and Zhiqing, L. (2009). A Software Process Tailoring Approach Using a Unified Lifecycle Template. Paper presented at the International Conference on Computational Intelligence and Software Engineering, 2009. CiSE 2009. , December 11-13, Wuhan, China.

[5] Pereira, E. B., Bastos, R. M., and Oliveira, T. C. (2007). *A systematic approach to process tailoring*. Paper presented at the International Conference on Systems Engineering and Modeling, 2007. ICSEM'07., March 20-23, Herzliyya-Heifa, Israel.

[6] Mikulenas, G., and Kapocius, K. (2011a). An approach for prioritizing agile practices for adaptation. In W. W. Song, S. Xu, C. Wan, Y. Zhong, W. Wojtkowski, G. Wojtkowski & H. Linger (Eds.), *Information Systems Development* (pp. 485-498). New York: Springer.

[7] Gonzalez-Perez, C., Giorgini, P., and Henderson-Sellers, B. (2009). Method construction by goal analysis. In C. Barry, K. Conboy, M. Lang, G. Wojtowski & W. Wojtowski (Eds.), Information Systems Development: Challenges in Practice, Theory and Education (Vol. 1, pp. 79-91). US: Springer.

[8] Li, J., and Mao, M. (2009). *A Case Study on Tailoring Software Process for Characteristics Based on RUP*. Paper presented at the International Conference on Computational Intelligence and Software Engineering, 2009. CiSE 2009. , December 11-13, Wuhan, China.

[9] Park, S., Na, H., Park, S., and Sugumaran, V. (2006). A semi-automated filtering technique for software process tailoring using neural network. *Expert Systems with Applications, 30*(2), 179-189.

[10] Park, S.-H., and Bae, D.-H. (2013). Tailoring a large-sized software process using process slicing and case-based reasoning technique. *IET Software*, 7(1), 47-55.

[11] Kang, D., Song, I.-G., Park, S., Bae, D.-H., Kim, H.-K., and Lee, N. (2008). *A case retrieval method for knowledge-based software process tailoring using structural similarity*. Paper presented at the 15th Asia-Pacific Software Engineering Conference, 2008. APSEC'08. , December 3-5, Beijing, China.

[12] Henninger, S., and Baumgarten, K. (2001). A case-based approach to tailoring software processes. In D. W. Aha & I. Watson (Eds.), *Case-Based Reasoning Research and Development* (pp. 249-262). Berlin Heidelberg: Springer.

[13] He, X. Y., Wang, Y. S., Teng, Y. X., and Guo, J. G. (2008). *A Systematic Method for Process Tailoring Based on Knowledge Reuse*. Paper presented at the International Conference on Software Engineering and Knowledge Engineering 2008 (SEKE 2008), 1-3 July 2008, San Francisco, USA.

[14] Xu, P., and Ramesh, B. (2003). *A tool for the capture and use of process knowledge in process tailoring*. Paper presented at the Proceedings of the 36th Annual Hawaii International Conference on System Sciences, 2003., Jan 6-9, Waikoloa, Hawaii.

[15] Yoon, I.-C., Min, S.-Y., and Bae, D.-H. (2001). *Tailoring and verifying software process*. Paper presented at the Eight Asia-Pacific Software Engineering Conference, 2001. APSEC 2001. , 4-7 December 2001, Macau, China.

e ISBN 978-629-99388-4-2

KICT
KULLIYYAH OF INFORMATION AND COMMUNICATION TECHNOLOGY

KICT Publishing