

Design and Implementation of Project Time Management Risk Assessment Tool for SME Projects using Oracle Application Express

Abdullahi Mohamud Sharif and Mohd. Zaidi Abd. Rozan

Abstract—Risk Assessment Tool (RAT) is an expert system that assesses, monitors, and gives preliminary treatments automatically based on the project plan. In this paper, a review was taken out for the current project time management risk assessment tools for SME software development projects, analyze risk assessment parameters, conditions, scenarios, and finally propose risk assessment tool (RAT) model to assess, treat, and monitor risks. An implementation prototype system is developed to validate the model.

Keywords—Project Time Management, Risk Assessment Tool (RAT), Small and Medium Enterprises (SME).

I. INTRODUCTION

PROJECT Risk Management and Project Time Management are part of the nine knowledge areas of project management life cycles (PMBOK *fourth edition*, 2008). Risks are being managed and mitigated throughout the project and through every phase of the project. According to [7], Risk management is defined as "A *systematic application of policies, procedures, methods and practices to the tasks of identifying, analyzing, evaluating, treating and monitoring risks.*"

On the other hand the definition for SME Software Development Projects (SMESDP) varies, and the size of software development projects are measured based on the number of Lines of Code (LOC) the project contains, the number of programmers doing the project, and the duration of the project. During the writing of this paper, the term SME Software Development Projects (SMESDP) is defined as any projects that have 50,000-100,000 LOC [2], 6-12 months, and ten or fewer programmers [5].

Most of the big enterprises have Risk Assessment Tool (RAT) and risk mitigation plan that act in accordance to project management standard templates like Project Management Body of Knowledge (PMBOK) and Project IN Controlled Environment (PRINCE2) to control risks of the projects at the beginning and during projects implementation. On the other hand, SMEs does not have RATs because those tools are usually designed for big enterprises and are not used for small and medium software projects for their few Lines of

Code (LOC), project staff, and time interval. Unfortunately, the parameters of the frameworks and templates of risk management in projects that big enterprises use are not applicable to SME projects. This has caused most of project managers to handle those SMESDPs time management risk assessments manually, and measure project time simply and superficially based on their experience. Hence, if big enterprises fail on single SME project, the impact will be like millipede does not stand for a broken leg. While doing more than one SMESDP at the same time, the failure of those projects induces higher impact to the company.

After going through processes such as interview, observation, finding information from Internet, referring books and journals, etc. Most of the Small and Medium-size Enterprise (SME) companies does not have risk registry. So by using web-based system can enhance the risk monitoring process, also able to be uploaded development project risks to the system, and easily can be fetched. Therefore, the important question that the paper focuses is: *How the risks in project time management for small and medium software development projects can be assessed, treated, and monitored systematically?*

To assess, treat, and monitor time risks in SMESDPs, research was done to categorize potential negative risk conditions associated with time management in a SMESDPs, the root causes of errors in estimating project time, and to analyze the parameters for risk assessment as the requirements for system design work. On the other hand, the project will focus only on at risk factors in a SMESDPs and risks associated with time management, such as errors in estimating time.

In this paper, research was done to gather potential negative risk conditions (shown in section II-A), risk assessment parameters (shown in section II-B), and review of the existing tools. After that, section III proposes approach for assessing risks by proposing a risk assessment tool (RAT) model in section III-A, and risk rules and processes in sections III-B and III-C respectively. In section IV, an implementation case study was carried out, and then a conclusion of the project and further study was summarized on section V and VI respectively.

II. LITERATURE REVIEW

A. Potential Negative Risk Conditions

In general, there are a lot of potential negative risk conditions for software development projects. But According to [8], [6], and [1], the potential negative risk conditions for project

Abdullahi is a Research Officer (RO) in the Department of Information Systems, Faculty of Computer Science and Information System, University Technology Malaysia (UTM), 81310 Skudai, Johor, Malaysia (Mobile: +60 173060508 /149106627, Email: saakuut@gmail.com).

Mohd. Zaidi is the Head of Information Systems Department, Faculty of Computer Science and Information System, University Technology Malaysia (UTM), 81310 Skudai, Johor, Malaysia (Mobile: +6017-9114820/+60197287573, Phone: +607-5532421, Email: mdzaidi@utm.my).

time management that have regular occurrence in software projects are listed in table I. These risks are considered to be issues that need to be addressed and thereafter controlled.

TABLE I
PROJECT TIME MANAGEMENT NEGATIVE RISK

Risk No.	Risk name
1	Errors in estimating time
2	Errors in determining the critical path
3	Poor allocation of activity
4	Early release of competitive products
5	Management float
6	Errors in resource availability
7	Creeping Requirements
8	Requirements Gold Plating
9	Developer Gold Plating
10	Shortchanged quality
11	Overly optimistic schedules
12	Inadequate design
13	Research Oriented Development
14	Weak personnel
15	Contractor Failure
16	Friction between developers and customer
17	Scheduling and timing

B. Risk Assessment Parameters

Generally, risks can be identified through interviews, reporting, brainstorming sessions, Delphi technique, decomposition, assumption analysis, critical path analysis, and utilization of risk taxonomies techniques [8], but identify risks in a project time management automatically using RAT needs beyond those techniques - risk parameters and scenarios to identify risks and then to register, store, monitor them, and give appropriate mitigation.

[11], [3], [10], and [8] provided a general sample of the risk registry parameters or format or the type of information or items which can be stored in a risk register parameters. But specifically, the risk assessment parameters for project time management risks are summarized in Table II.

C. Review of Existing Tools

In general, risk assessment software's or risk management software's are becoming increased in the market; they include software's related to the generic projects like constructions and software projects. Additionally, the tools that assess risks in software projects are mostly covering all risks in that area or the triple constraints of projects time, cost, and quality. Besides that, dividing project risks into sub constraints time, cost, and quality may increase the efficient of getting the right problem and the appropriate solutions for it. After an exclusive research on project time management RATs, there is no system specifically deals with risks related to the project time management for SMESDPs.

III. PROPOSED APPROACH

A. RAT Model

Risk Assessment Tool (RAT) is an expert system that assesses, monitors, and gives preliminary treatments automatically based on the project plan. In Risk Assessment Tool

TABLE II
SUMMARY OF RISK ASSESSMENT PARAMETERS

Risk	Description
Risk ID:	Risk number
Risk:	Short-form of Risk name
Rank:	Number
Description:	Risk in detail
Category:	1-Unknown, 2-Low, 3- Medium, 4- High, 5- Fatal
Occurrence:	Estimated likelihood of risk occurrence
Root cause:	The root cause of the risk
Dependence:	Nature of any significant inter-dependencies with other risks (e.g. the Risk ID of the dependent risk will be entered)
Triggers:	The symptoms of the actual risk event
Potential Responses:	Potential Response to each risk
Response effect:	Likely effect of responses on the risk
Risk Owner:	The person who will own or take responsibility for the risk
Probability:	1 -Low, 2- Medium, 3- High
Project Impact:	1 -Low, 2- Medium, 3- High
Objective Impact:	Project objectives on which the risk impacts (e.g. scheduling, cost, etc)
Impact estimate:	Risk impact estimate
Status:	1 -Found, 2- Analyzed, 3- Response present, 4- Response selected, 5- Response implemented, 6- Closed
Residual:	Residual risk after effective response

(RAT) Model, there are 5 important tasks that each one has a connection to other one through specified process; users, project plan input, risk rules which includes risk ranking matrix, risk conditions, and risk scenarios, risk fetching processes, and risk reports (as shown in fig. 1).

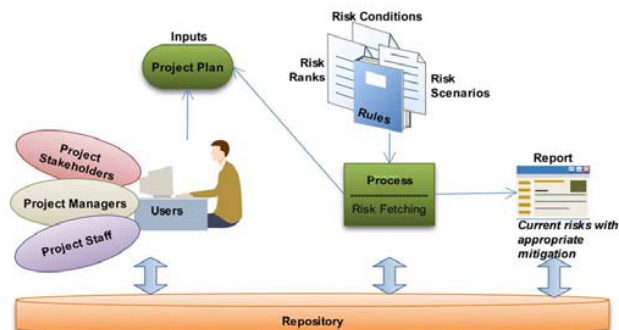


Fig. 1. RAT Model

Users of the project are the project members that have direct access for risk assessing and monitoring including project stakeholders, project managers, and project team members. According to [9], stakeholders are people have direct or indirect influence to the project activities like project sponsors. Those users are firstly uploads the project plan to the repository for assessing and fetching the risks automatically. Hence, project plan is the detailed tasks of a project like task name, duration of the task, task owner, and so on.

After the project plan is being uploaded, risk fetching process will generate potential risks for the recently uploaded plan as shown in fig. 5 on page 4 and explained more in section

III-C. Additionally, while risks are being assessed through risk fetching process, a certain rules will be used to identify risks, ranking it based on the available risk scenarios as discussed in section III-B. Finally, a detailed list of current risks of the project is being displayed to the user to take the final decision to select among displayed risks which will be taken as a risk and being set to monitoring or under control state, and being saved to the repository.

B. Risk Rules

Risk ranks, risk conditions, and risk scenarios are the daemon rules that allow risk process fetching to be able to fetch risks from the uploaded project plan correctly and systematically

1) *Risk Ranks*: Project managers manage rank risks manually based on the risk category, impact, and probability of occurrence. A few researchers like [4] has shown a general ranking table for risk registry. In RAT model, while the risk is automatically fetched from the project plan, a risk rank matrix based on brainstorming and interviewing techniques that follows the same scenario will be ranked for every risk in the project automatically as shown in table III.

2) *Risk Conditions*: Risk conditions are normally Structured Query Language (SQL) statements that examine the uploaded project plan and assign the identified risks for their appropriate rank and scenario. In the RAT prototype, a risk conditions was developed as discussed below.

Risk Condition 1:: This condition fires when the finish date of a specific task is today and the state of the task is open (as shown if fig. 2). If this condition is true then it calls risk scenario 1 (as shown in table IV on page 4), and assigns the appropriate risk rank based on table III.

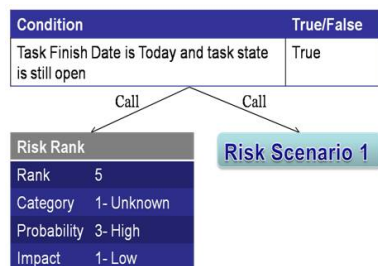


Fig. 2. Risk Condition 1

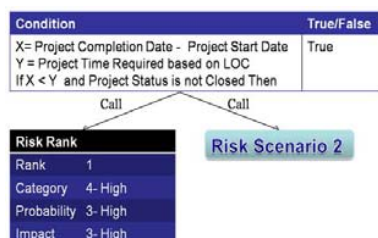


Fig. 3. Risk Condition 2

TABLE III
RISK RANK MATRIX

Rank	Category	Probability	Project Impact
1	5	3	3
1	5	3	1
1	5	3	2
1	5	2	2
1	5	2	3
1	5	1	3
1	4	3	3
2	5	1	2
2	4	3	1
2	4	3	2
2	4	2	3
2	4	2	2
2	3	3	2
2	3	3	3
3	5	2	1
3	4	2	1
3	4	1	3
3	3	3	1
3	3	2	3
3	3	2	2
4	5	1	1
4	4	1	1
4	4	1	2
4	3	2	1
4	3	1	3
4	2	3	2
4	2	3	3
4	2	3	1
4	2	2	3
4	2	2	2
4	1	3	3
4	1	1	2
4	1	1	3
5	3	1	1
5	3	1	2
5	2	2	1
5	2	1	1
5	2	1	2
5	2	1	3
5	1	3	1
5	1	3	2
5	1	2	1
5	1	2	2
5	1	2	3
5	1	1	1

Risk Condition 2:: Likewise, risk condition 2 calls risk scenario 2 (as shown in table V on page 4), and assigns appropriate risk rank based on table III, when the difference of project completion and project start date are less than the allocated time of the project based on LOC (as shown in fig. 3), and the project status is open. The project time requirement based on LOC is calculated by following this formula:

$$Efforts(person_months) = 1.4 * LOC^{1000}$$

$$Project\ Time\ Requirements = 3.0 * person_month^{1/3}$$

Risk Condition 3:: Lastly, as shown in fig. 4, the risk condition fires when project assigned staff is less then project staff required based on LOC and the status of the project is open by calling risk scenario 3 (refer table VI on page 4), and risk ranking matrix. The project staff allocated for the project is calculated based on below formula, while project efforts and

project time requirements formulas are being defined in above risk scenario 2 paragraph.

$$\text{Project Staff} = \frac{\text{Project Efforts}(\text{person_months})}{\text{Project Time Requirement}}$$

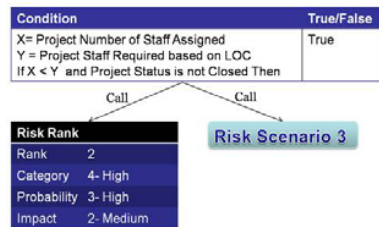


Fig. 4. Risk Condition 3

3) *Risk Scenarios*: Risk scenarios are also brainstorming scenarios that are based on project time management risk conditions and its normally stored in a table and fetched based on the risk condition fired. In the prototype, three sample risk scenarios are being created based on previous risk conditions, and each scenario is called when their specific risk condition is fired. Risk scenario 1, 2, and 3 that shown in tables IV, V, and VI are called after firing risk condition 1, 2, and 3 respectively.

Each risk scenario, stores information about the fired risk starting from risk name, risk description, occurrence, impact estimate, root causes, triggers, and potential responses. And all of these information is gathered by using brainstorming technique.

TABLE IV
RISK SCENARIO 1

Risk Name	Task Time Frame
Risk Description	The finish date of the risk is today and the task is still open state.
Occurrence	Today
Impact	Estimate Can cause delay to other risks
Root Causes	The task is not completed yet and today is the last day
Triggers	The risk finish date is today
Potential Response	To put the task on eye until it being finished today

C. Risk Process

As shown in fig. 5, risk process allows users to input project plan and on the other side fetches project risks automatically through risk rules and saves the displayed report to the repository after the project user selects the real project risks. Every time the user visits the project risks page, the risks fetching process is being executed to display previous and new risks. Finally, users are also able to add risks manually.

IV. CASE STUDY

In this case study, an overview of the implementation of the prototype will be discussed. The RAT system is being

TABLE V
RISK SCENARIO 2

Risk Name	Project Implementation Time Interval.
Risk Description	The identified duration of the project implementation (based on start date and finish date) are less then the estimated time for the project.
Occurrence	Any Time...
Impact Estimate	Increasing more budgets to rush before the dateline and working extra time.
Root Causes	A poor estimation of the project time interval can cause like this problem.
Triggers	When the estimated time of the project based on the Lines Of Code and the identified times based on start and compilation dates are not merging together.
Potential Response	By redefining or re-estimating the project time based on the Lines Of Code of the project.

TABLE VI
RISK SCENARIO 3

Risk Name	Not enough Staff
Risk Description	The project assigned staff are less than the calculated staff required based on effort size (person-months) and schedule time from the Lines of Code (LOC) of the project.
Occurrence	Any Time...
Impact Estimate	It can delay of the project compilation or failure of the project.
Root Causes	Project Manager does not assign the sufficient project staffs
Triggers	The trigger is when the assigned project staff is less than the result of effort size divided by schedule time for the LOC
Potential Response	To Increase the project staff based on the LOC staff requirement will prevent project to be delayed or failure.

developed by using Oracle Application Express (Apex) 3.2 as a website developer and Oracle Database 11g for the database development. Oracle Apex does not require much coding background, it's a click based application that creates pages from the tables query and Procedural Language/Structured Query Language (PL/SQL) commands. Oracle Apex 3.2 is being selected for the capability of interactive search tool and quick development.

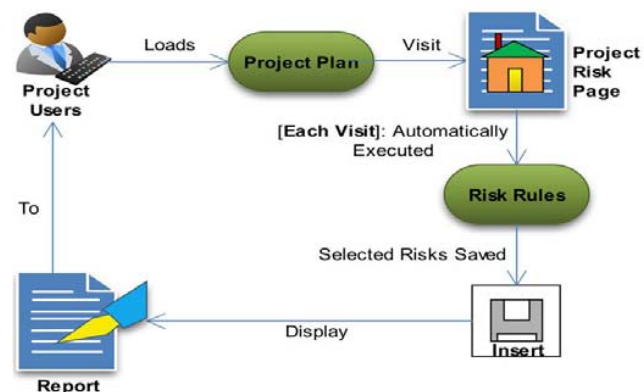
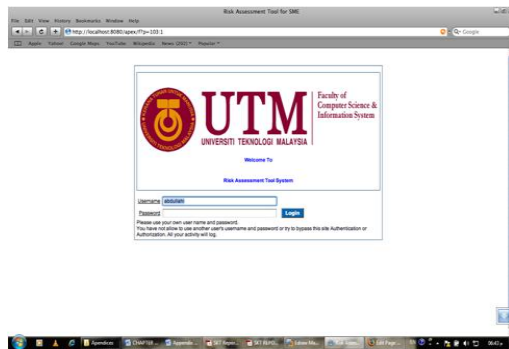
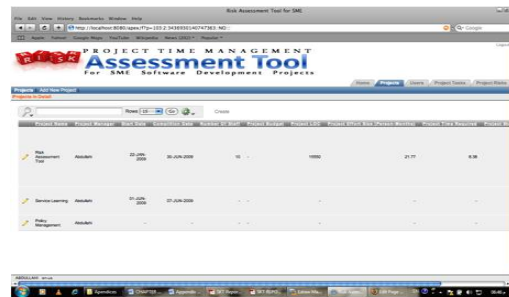
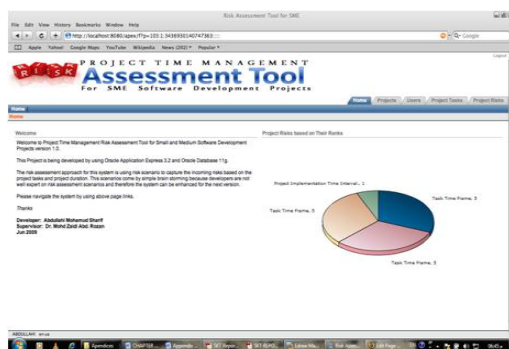
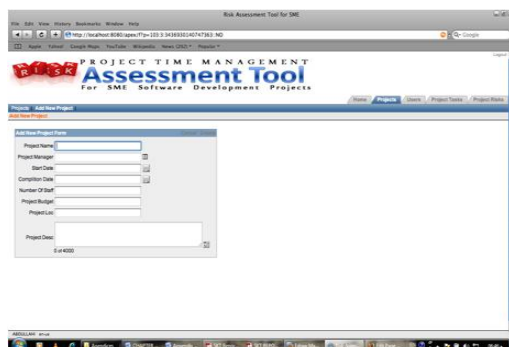


Fig. 5. Risk Fetching Process

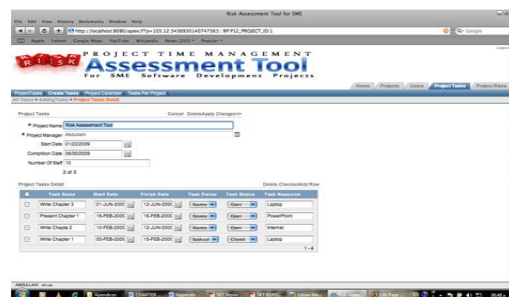
Step 1**Step 4****Step 2**

After successfully logged the system in step 1 and 2, project user creates firstly new project in step 3 by assigning project name, starting and finishing dates, providing project LOC, budget, number of staff, and sample description of the project.

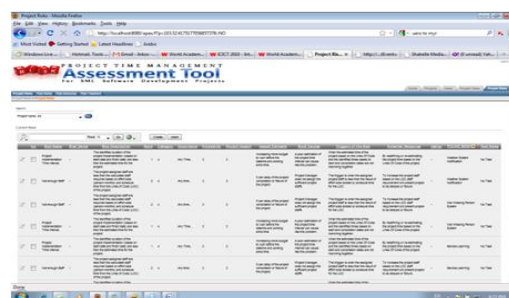
Step 3

Also project users can view existing projects as shown in step 4 after they add new project.

After completing project addition, the project user will add tasks to the created projects as shown in step 5 by filling the task name and resources, and selecting task's start and finish dates, and task owner and status.

Step 5

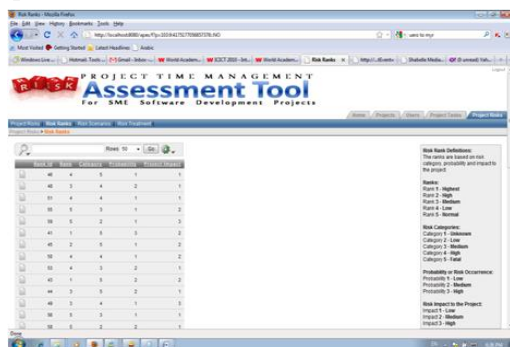
Step 5 will be repeated until the project plan is completed and all tasks are being uploaded to the system. After that, project user goes to project risks as shown in step 6 to view risks on the uploaded tasks. Hence, by displaying the project risk page, the risk fetching process is taken over and a detailed list of current risks on the project will be shown.

Step 6

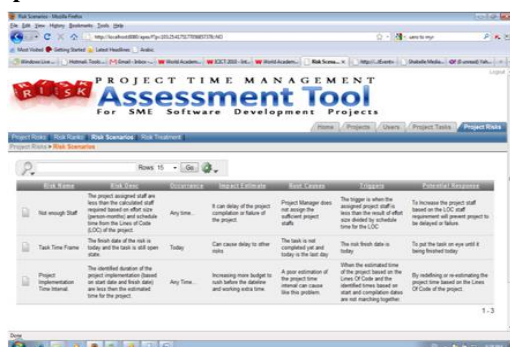
After displaying project risks, the project user identifies which risks will be selected to be saved on to the repository for future monitoring as shown in step 6 with interactive selection of the risks based on their project.

Finally, step 7 and 8 (shown in page 6) are showing risk ranks and risk scenarios respectively that the project fellows. As mentioned in previous sections, this prototype is being developed with sample risk scenarios and risk ranks.

Step 7



Step 8



V. CONCLUSION

In conclusion, the risk assessment tool is an initiative taken in purpose to help stake holders, project managers, and project team members to assess, and monitor project time management risks. By using RAT model, project managers can easily identify project time risks automatically through project plan inputs.

Finally, the expected result of this paper is hopefully could give the overall benefits to all users of the project time management risk assessment tool.

VI. FURTHER STUDY

The system can be developed in the future by using different programs rather than Oracle Application Express 3.2 and Oracle Database 11g. Also the risk scenarios and conditions that the system is being developed can be increased as much as scenarios or conditions is being declared because the existing risk scenarios and conditions are based on a non-expert brain storming information.

ACKNOWLEDGMENT

The authors would like to thank University Technology Malaysia (UTM) for their cooperation toward submitting and presenting this paper.

REFERENCES

- [1] Tarek K. Abdel-Hamid, Kishore Sengupta, and Clint Swett. The impact of goals on software project management: an experimental investigation. *MIS Q.*, 23(4):531–555, 1999.
- [2] Dennis Alan, Haley Wixom Barbara, and Tegarden David Paul. *System Analysis and Design with UML Version 2.0 an Object-Oriented Approach*. John Wiley and Sons, Inc., U.S., 2005.
- [3] Carter Bruce, Hancock Tony, Morin Jean-Marc, and Robins Ned. *Introducing RISKMAN: the European project risk management methodology*. NNC Blackwell Ltd, UK, 1994.
- [4] Patterson Fiona D. and Neailey Kevin. A risk register database system to aid the management of project risk. *International Journal of Project Management*, 20(5):365– 374, Jul 2002.
- [5] Donna L. Johnson. *Risk Management and the Small Software Project*. LOGOS International, Inc., Nashville, Tennessee, 2006.
- [6] Steve McConnell. *Rapid development: taming wild software schedules*. Microsoft Press, Redmond, W.A, 1996.
- [7] Robert R. Moeller. *COSO Enterprise Risk Management: Understanding The New Integrated ERM Framework*. John Wiley and Sons, Hoboken, New Jersey, 2007.
- [8] Kathy Schwalbe. *Information Technology Project Management*. Thomson Course Technology, USA, 2007.
- [9] Kathy Schwalbe. *Information Technology Project Management*. Cengage Learning, USA, 2009.
- [10] SC Ward. Assessing and managing important risks. *International Journal of Project Management*, 17(6):331–336, Dec 1999.
- [11] Terry M. Willams. Using a risk register to integrate risks management in project definition. *International Journal of Project Management*, 12(1):17– 22, Feb 1994.



Malaysia (UTM), Malaysia.



Abd Rozan, Mohd Zaidi (Dr.) has received his B.Sc. (Hons.) in Physics & Comp w. Ed., and M.Sc. IT from Universiti Teknologi Malaysia (UTM), Malaysia. He has received a Doctorate of Engineering (D.Eng) in Information Science & Control Engineering from Nagaoka University of Technology, Japan. He is also a PRINCE2 Certified & Registered Project Management Practitioner. Currently, he is the Head, Department of Information Systems, Faculty of Computer Science & Information Systems, Universiti Teknologi Malaysia (UTM) and an active researcher for the Information Service Systems & Innovation (ISSI) Research Group, UTM. Prior to joining UTM, he has experienced working as a Tour Coordinator with a tour company in Kota Kinabalu, Sabah. Also has involved in IT project implementation with several clients during his appointment as a Database Systems Executive in an International System Development Firm, based in Johore, Malaysia. His research interests are IT Project Management, Profiling and Data Mining utilizing Multivariate Approach, and Technopreneurship. He holds a Radio Amateur Licence, with callsign 9W2DZD.