UNIVERSITI PUTRA MALAYSIA

IMPLEMENTATION FRAMEWORK BASED ON DESIGN REFINEMENTS FOR SYSTEMATIC MIGRATION OF LEGACY APPLICATIONS TO CLOUD

MUHAMMAD EHSAN RANA

FSKTM 2020 23

IMPLEMENTATION FRAMEWORK BASED ON DESIGN REFINEMENTS FOR SYSTEMATIC MIGRATION OF LEGACY APPLICATIONS TO CLOUD

By

MUHAMMAD EHSAN RANA

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfilment of the Requirements for the Degree of Doctor of Philosophy

June 2020

COPYRIGHT

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfilment of the requirement for the degree of Doctor of Philosophy

IMPLEMENTATION FRAMEWORK BASED ON DESIGN REFINEMENTS FOR SYSTEMATIC MIGRATION OF LEGACY APPLICATIONS TO CLOUD

By

MUHAMMAD EHSAN RANA

June 2020

Chairman : Associate Professor Wan Nurhayati Wan Ab. Rahman, PhD
Faculty : Computer Science and Information Technology

Legacy applications refer to the applications that have been in use for a long period of time and developed using obsolete technologies. They are often business-critical systems; therefore, any changes here inevitably will affect the other parts of the system. Legacy applications need to be compatible and up to date with the contemporary business requirements and modern infrastructure, however their outdated architecture and stern design constraints proved to be a major hurdle towards achieving this goal. Cloud computing offers numerous benefits over traditional on-premises infrastructure. It provides a new platform for organizations, which promises flexible scalability, business agility, high availability and reduction in cost. Considering these benefits, migration of legacy applications to cloud is a lucrative option for many organizations. However, the architecture of these legacy applications requires a tested, fool-proof and risk-free approach for migration. The existing models or frameworks proposed for the migration of legacy applications does not address all major issues of migration. These models either lack the software re-architecting phase or mentions it in an abstract and brief way without involving the specifics and intricacies of its application. This research explores and evaluates the coverage of design refinement phase in existing studies for migration of legacy applications to cloud and proposes a design refinement-based implementation framework for systematic migration of legacy applications to cloud. For the purpose of designing the framework, structured interviews are employed to seek the most up-to-date feedback regarding current industry trends. The information obtained through the interviews is used to gain decisive inputs for the designing of the artefact. As part of this research, it is justified with the help of empirical evidence that the use of good practices through design patterns help in improving the three key quality factors of software namely efficiency, flexibility and maintainability. The information gained is used to incorporate legacy application's design refinements as one of the core components of the proposed

implementation framework for systematic migration of legacy applications to cloud. Expert review is used to validate the proposed implementation framework. Moreover, two applications were chosen and migrated to cloud by applying necessary design refinements as one of the core steps of the proposed framework. Expert review confirms that the framework is easy to implement and is applicable to all typical legacy applications which allows the architects to follow the steps sequentially for a systematic migration. After successful migration, the performance of the application is compared to its on-premise deployment to see the overall performance improvements as well as the improvements in terms of the stated quality factors (an increase in 51% efficiency, 66% flexibility and 7% maintainability is recorded in the migrated application). The main contribution of this research is to systematically embed and justify with evidence the significance of architectural and design refinement process that improves efficiency, flexibility and maintainability of the legacy application as part of its migration to cloud. This systematic design refinement process ensures improving the overall architecture suitable for cloud infrastructure.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk ijazah Doktor Falsafah

KERANGKA PELAKSANAAN BERDASARKAN PENAMBAHBAIKAN REKA BENTUK UNTUK MIGRASI SISTEMATIK PERMOHONAN UNDANG-UNDANG UNTUK MENGUMPUL

Oleh

MUHAMMAD EHSAN RANA

Jun 2020

Pengerusi : Profesor Madya Wan Nurhayati Wan Ab. Rahman, PhD
Fakulti : Sains Komputer dan Teknologi Maklumat

Sistem warisan merujuk pada aplikasi yang telah digunakan untuk jangka waktu yang lama dan dikembangkan menggunakan teknologi usang. Mereka selalunya sistem kritikal perniagaan; oleh itu, sebarang perubahan di sini pasti akan mempengaruhi bahagian sistem yang lain. Aplikasi warisan harus serasi dan terkini dengan keperluan perniagaan kontemporari dan infrastruktur moden, namun seni bina mereka yang ketinggalan zaman dan kekangan reka bentuk yang tegas terbukti menjadi halangan utama untuk mencapai tujuan ini. Pengkomputeran awan menawarkan banyak faedah berbanding infrastruktur tradisional di tempat. Ini menyediakan platform baru untuk organisasi, yang menjanjikan skalabilitas fleksibel, kelincahan perniagaan, ketersediaan tinggi dan pengurangan kos. Memandangkan faedah ini, penghijrahan sistem warisan ke awan adalah pilihan yang menguntungkan bagi banyak organisasi. Walau bagaimanapun, seni bina aplikasi warisan ini memerlukan pendekatan bebas, tanpa bukti dan bebas risiko untuk migrasi. Model atau kerangka kerja yang ada yang dicadangkan untuk cloudifikasi sistem warisan tidak menangani semua masalah utama dalam migrasi. Model-model ini tidak mempunyai fasa pengarsipan semula perisian atau menyebutnya secara abstrak dan ringkas tanpa melibatkan spesifik dan selok-belok penerapannya. Penyelidikan ini meneroka dan menilai liputan fasa penyempurnaan reka bentuk dalam kajian yang ada untuk penghijrahan aplikasi warisan ke awan dan mencadangkan kerangka pelaksanaan berasaskan penyempurnaan reka bentuk untuk penghijrahan aplikasi warisan ke cloud secara sistematik. Untuk tujuan merancang kerangka kerja, wawancara berstruktur digunakan untuk mendapatkan maklum balas terkini mengenai trend industri semasa. Maklumat yang diperoleh melalui temu ramah digunakan untuk mendapatkan input yang menentukan untuk merancang artefak. Sebagai sebahagian daripada penyelidikan ini, dibenarkan dengan bantuan bukti empirikal bahawa

penggunaan amalan baik melalui corak reka bentuk membantu dalam meningkatkan tiga faktor kualiti utama perisian iaitu kecekapan, fleksibiliti dan kesenggaraan. Maklumat yang diperoleh digunakan untuk menggabungkan penyempurnaan reka bentuk sistem warisan sebagai salah satu komponen teras kerangka pelaksanaan yang dicadangkan untuk penghijrahan aplikasi warisan sistematik ke cloud. Untuk mengesahkan kerangka pelaksanaan yang dicadangkan, kajian pakar digunakan. Selain itu, dua aplikasi dipilih dan dimigrasikan ke cloud dengan menerapkan penyempurnaan reka bentuk yang diperlukan sebagai salah satu langkah utama kerangka kerja yang dicadangkan. Penghijrahan aplikasi yang berjaya mengesahkan bahawa kerangka kerja mudah dilaksanakan dan dapat diterapkan pada semua aplikasi warisan khas yang memungkinkan para arkitek mengikuti langkah-langkah secara berurutan untuk migrasi yang lancar. Setelah penghijrahan berjaya, prestasi aplikasi dibandingkan dengan penggunaan di premis untuk melihat peningkatan prestasi keseluruhan dan juga peningkatan dari segi faktor kualiti yang dinyatakan. Penyelidikan ini adalah penambahan yang berharga bagi kedua-dua penyelidik dan pengamal industri yang bersangkutan dalam memindahkan aplikasi warisan ke awan dengan menyediakan proses penyempurnaan reka bentuk sistematik untuk aplikasi warisan yang memastikan peningkatan keseluruhan arsitektur yang sesuai untuk infrastruktur awan.

ACKNOWLEDGEMENTS

First and foremost, I would like to praise Allah Almighty for giving me courage and strength to do and complete this thesis. Without His blessings and guidance, this thesis would not have materialized.

Secondly, I would like to thank my supervisor Associate Professor Dr Wan Nurhayati Wan Ab. Rahman of the Faculty of Computer Science and Information Technology. She has been extremely considerate and flexible in terms of providing guidance and allocating time to listen and solve my issues. She always directed me in the right track and provided me the required support. Her guidance in forming the thesis into its present shape is matchless.

I would also like to thank my supervisory committee members, Associate Professor Dr Masrah Azrifah Azmi Murad and Associate Professor Dr Rodziah Atan, who were always available to assist me. Their feedback during the progress presentations and review process proved to be extremely helpful in refining and improving the research work.

I must express my very profound gratitude to my parents, wife and kids for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without their cooperation.

I would also like to thank the industry experts who were involved in the interview process for this research. Without their passionate participation and input, the proposed implementation framework could not have been properly formulated. Finally, special thanks to my colleagues and friends who motivated and assisted me throughout this journey.

This thesis was submitted to the Senate of the Universiti Putra Malaysia and has been accepted as fulfilment of the requirement for the degree of Doctor of Philosophy. The members of the Supervisory Committee were as follows:

Wan Nurhayati Wan Ab. Rahman, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

Masrah Azrifah Azmi Murad, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

Rodziah Atan, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

ZALILAH MOHD SHARIFF, PhD
Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia

Date: 10 December 2020

Declaration by graduate student

I hereby confirm that:

- this thesis is my original work;
- quotations, illustrations and citations have been duly referenced;
- this thesis has not been submitted previously or concurrently for any other degree at any institutions;
- intellectual property from the thesis and copyright of thesis are fully-owned by Universiti Putra Malaysia, as according to the Universiti Putra Malaysia (Research) Rules 2012;
- written permission must be obtained from supervisor and the office of Deputy Vice-Chancellor (Research and innovation) before thesis is published (in the form of written, printed or in electronic form) including books, journals, modules, proceedings, popular writings, seminar papers, manuscripts, posters, reports, lecture notes, learning modules or any other materials as stated in the Universiti Putra Malaysia (Research) Rules 2012;
- there is no plagiarism or data falsification/fabrication in the thesis, and scholarly integrity is upheld as according to the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) and the Universiti Putra Malaysia (Research) Rules 2012. The thesis has undergone plagiarism detection software

Signature: _____ Date: _____

Name and Matric No: Muhammad Ehsan Rana GS36562

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AMI	Amazon Machine Image
API	Application Programming Interface
AWS	Amazon Web Services
CAM	Cohesion Among Methods of Class
CKJM	Chidamber and Kemerer Java Metrics
CPU	Central Processing Unit
CRM	Customer Relationship Management
DAM	Data Access Metrics
DB	Database
DCC	Direct Class Coupling
DFD	Data Flow Diagram
EC2	Elastic Compute Cloud
ERP	Enterprise Resource Planning
GoF	Gang of Four
IaaS	Infrastructure as a Service
IDE	Integrated Development Environment
IIS	Internet Information Services
IT	Information Technology
MDA	Model-Driven Architecture
MDE	Model-Driven Engineering
MELIS	Migration Environment for Legacy Information Systems
MFA	Measure of Functional Abstraction
MI	Maintainability Index
MKV	Matroska Video (file extension)
MTS	Movie Ticketing System
NGO	Non-Governmental Organization
NIST	National Institute of Standards and Technology

OLTP	Online Transaction Processing
OO	Object-Oriented
ORM	Object-Relational Mapping
OS	Operating System
PaaS	Platform as a Service
POC	Proof of Concept
RAM	Random Access Memory
S3	Simple Storage Service
SaaS	Software as a Service
SLA	Service Level Agreement
SME	Small and Medium-sized Enterprises
SOLID	S - Single-responsibility principle
	O - Open-closed principle
	L - Liskov substitution principle
	I - Interface segregation principle
	D - Dependency Inversion Principle
SOA	Service Oriented Architecture
SQL	Structured Query Language
UAL	UML Action Language
UML	Unified Modeling Language
VCR	Video Cassette Recorder
VM	Virtual Machine
WCF	Windows Communication Foundation

CHAPTER 1

INTRODUCTION

1.1 Background

Legacy applications refer to the applications that have been in use for a long period of time and developed using obsolete technologies. They are often business-critical systems; therefore, any changes here inevitably will affect the other parts of the system. Although these systems are considered to be outdated but are too costly and risky for an organization to replace it (Kumar & Jain, 2017; Sommerville, 2011, p. 38).

Cloud computing provides a new platform for organization that promises flexible scalability, business agility, high availability and reduction in costs (Mell & Grance, 2011). Considering these benefits, migration of legacy applications to cloud is a lucrative option for many organizations (Colosimo et al., 2009). However the architecture of these legacy applications require a tested, fool-proof and risk free approach for migration (Beserra et al., 2012).

Virtualization is the first step towards moving an application to cloud, however legacy applications are typically not architected even for a virtualized infrastructure (Frey & Hasselbring, 2011). Understandably, when designed, the legacy applications were not intended to be ported on cloud. A legacy application built using the on-premise infrastructure in mind will not have the capabilities of integrating services and hence is not easily deployable on cloud. At times, legacy application relies on hardware dependent modules that are not cloud conversant. These modules might represent some specific business requirements of a business critical application, therefore need to be dealt with proper planning and execution (Frey & Hasselbring, 2010; Illa et al., 2012).

Shifting and migrating legacy application onto the cloud is a complex task which sometimes becomes more challenging than simply developing an application from scratch with the cloud infrastructure in mind. However replacing these legacy applications by newly developed solutions involve high risk, cost and time which is never a priority of any enterprise (Colosimo et al., 2009). One of the major tasks involved in this migration process is to re-architecture the application in such a way that it can exploit the available benefits that cloud computing is offering to its clients (Frey & Hasselbring, 2011). However the absence of implementation frameworks to smoothly and efficiently carrying out this process is a major hurdle towards achieving this target (Rimal et al., 2011).

1.2 Research Motivation

Even though the use of cloud computing has significantly increased in the last decade, but there exist several domains of cloud adoption especially the area related to legacy application migration that has not been fully explored yet. In recent years, to execute migration of these legacy applications, more studies have been performed and some models and methodologies have been proposed. However, most of these models and methodologies did not adequately explain the phenomenon; the approach is problematic or contradict with the existing updated knowledge. One such territory is the migrated application design and architecture that should fit well as per the cloud infrastructural requirements.

The available migration models and frameworks proposed for legacy applications' migration process emphasize merely on the generic phases and procedures on migrating the applications and data to cloud. Some of these models of cloud migration are based on advantages of deploying existing applications to cloud which deals with issues like cost, time, security, organizational culture and market maturity etc. Only few researchers have identified application re-architecting as one of the constituent phases of migration process but seemed to provide minimal or no explanation on how this modification of design can contribute towards improving the missing elements of legacy design. The existing research in this domain is limited to conceptual artefacts and lacks the empirical validation of the redesigning approach.

1.3 Problem Statement

Fahmideh et al. (2017) state that existing methods for legacy application's migration to cloud do not provide a holistic solution considering the key challenges from re-engineering perspective. In their research work, Fahmideh et al. (2017) attempt to provide a way of applying design refinements as an essential part of legacy applications migration to cloud by using design principles, however this research is restricted to the application of four pre-fixed design principles. It is hard to scope the refinement of legacy application design using these four pre-fixed design principles. Shadrin (2005) concludes that the design refinements should be applied sequentially by incorporating appropriate design principles, heuristics and design patterns, and if these elements of object orientation are not followed sequentially, it increases the chance of inducing flaws and defects in the design. There are various other design principles and patterns that can be applied to fix many other design issues, hence specifying four design principles for all migration scenarios might not be appropriate for all such cases. Design issues like application rigidity, high coupling, exposing of creational logic, program to an implementation rather abstraction in conventional legacy systems require a combination of design principles and patterns to be applied to fix these issues. Hence, the implementation details of re-architecting

phase for the migration process need to be systematically executed in order to get the benefits of cloud flexibility. Legacy applications usually follow a monolithic and self-contained design which is highly coupled and inter-dependent as opposed to a modular design which is much more efficient and reusable (Guillén et al., 2013). Frey and Hasselbring (2011) have indicated that most of the legacy applications are not architected with the application flexibility factor in mind. Therefore, an attempt to forklift a legacy application to cloud would incorporate rigid structures and undesirable results. This intensifies the need for re-architecting the legacy application before migration (Borges et al., 2018). Feldman (2013) has pointed out that many organizations take legacy spaghetti and converts it into a cloud spaghetti without making it elegant for cloud infrastructure. Frey and Hasselbring's research (2011) establishes that most of the legacy applications are not architected with the application flexibility factor in mind. Some researches specify inefficiency (a delay in transmitting or processing data), inflexibility and poor maintainability as major performance issues that a traditional legacy application suffers from (Li et al., 2007; Rai et al., 2015). Based on these three major problems associated with legacy applications, design refinements should be applied to enhance efficiency, flexibility and maintainability of the application before migrating to cloud for optimal performance. These signify the need of a new design refinement-based implementation framework for systematic migration of legacy applications to cloud that can target the improvement of the stated quality factors.

1.4 Research Questions

This research was factored further into manageable research questions to thoroughly examine this domain. Following research questions were articulated to carry out research.

Question 1. What is the coverage of design refinement phase in existing legacy application frameworks for cloud migration?

Question 2. What is the significance of design refinement phase in an implementation framework for migrating legacy applications?

Question 3. What methods and techniques can be employed for improving efficiency, flexibility and maintainability of an application by refining application design?

Question 4. How does the implementation framework be validated for systematic migration and effectiveness?

1.5 Research Objectives

The main objective of this research is to propose an implementation framework for systematic migration of legacy applications to cloud by applying suitable design refinements.

The main objective is supported by the following specific objectives:

1. To investigate existing legacy application frameworks for cloud migration in terms of their coverage of design refinements.
2. To empirically evaluate the use of design refinements in cloud migration by applying design principles and patterns for improvement in software quality.
3. To develop an implementation framework for systematic migration of legacy applications to cloud by applying design refinements to optimize the key quality factors.
4. To conduct validation of the proposed implementation framework to ensure optimization of the key quality factors.

1.6 Research Scope

The scope of this research is limited to following major segments of legacy applications' cloud migration process:

- The motivation for legacy application's migration to cloud considering the advancement of cloud computing.
- The techniques, options and potential concerns involved in legacy applications' migration process.
- The legacy application architectural constraints in migrating to cloud and the problems associated with it.
- The constraints and issues involved in available frameworks and models in terms of their coverage of design refinements.
- The systematic phases required in implementation framework for migration of legacy applications to cloud.

The research related to the design refinements is pretty much oriented towards the use of design principles and design patterns. The emphasis is given on re-designing the application with cloud infrastructure in mind, re-developing its architecture to improve the quality factors and eventually the modification of application code subject to the changes in design. This research even though generally touches application's data migration concerns, does not specifically cater for the issues in moving application data to cloud. Therefore, the

associated issues related to data conversion rules, audit trails, checksums and cleaning data defects etc. are beyond the scope of this research.

There are potential risks related to cloud security, privacy and trust at various levels of application deployment. However, this doesn't come under the scope of this research as the author is confined with the refinement of application architecture to contribute for three key quality factors namely efficiency, flexibility and maintainability. Moreover, several studies have already proved to improve the application security by applying various strategies on the cloud. Similarly, issues related to multi-tenancy and data leakage are not part of this investigation. Guidelines have been developed for better governance and vulnerability management that can be taken in account for such issues. Exploring the shortcomings in governance related matters of cloud is a different territory of research.

1.7 Thesis Structure

The thesis is organised to present the research process and outcomes in a sequential flow.

Chapter 1 provides a basis for the overall research process and sets forth a measurable criterion through research questions and objectives. Problem statement concisely details the identified gaps and states the focus of investigation in the research undertaken. The chapter also includes research motivation and scope.

Chapter 2 encompasses literature review which focuses on associated problems, considerations and potential concerns related to legacy application migration techniques based on existing models and frameworks. It particularly reviews the existing frameworks that have been proposed for legacy applications with regards to their coverage of design refinement phase. This chapter also helps author to uncover emerging trends and success strategies in the stated research domain. Detailed discussions on object-oriented elements are undertaken to identify techniques of enhancing quality through design modifications. This part compliments the need of design and architectural refinements in the migration framework. The last section of this chapter formulates the conceptual migration framework.

Chapter 3 outlines the research methodology with detailed justifications. This chapter presents the applied conceptual framework of research that illustrates the key factors involved, inputs, outputs and relationships among them. The conducted research has been divided into four discrete and sequential phases which are explained thoroughly in distinct sections.

Chapter 4 provides analysis of the data gathered from structured interviews through a systematic coding process. It examines the responses in a methodical way to reach a valuable set of recommendations for the research in consideration. It also presents in detail the findings of empirical evaluation of essential quality factors for cloud migration that includes software efficiency, flexibility and maintainability.

Chapter 5 presents the proposed solution in the form of an implementation framework to be used for systematic migration of legacy applications to cloud. The framework is constructed by incorporating good practices based on research and industry feedback. The core components of this framework are structured in a way that signifies it from the existing researches. The existing literature is used to discover and understand the concepts and issues involved in this development.

Chapter 6 evaluates the validity of the proposed implementation framework by migrating two different applications to two heterogeneous cloud platforms. All steps recommended by the framework were implemented to carry out the migration process. The performance of the application before and after deployment is compared. The correctness of the findings is validated through expert review. The experts reviewed the implementation framework and thoroughly examined the results of porting the applications to cloud.

Chapter 7 summarizes the research outcomes and provides conclusions. It provides a detailed account of research contributions in the existing body of knowledge. This chapter also includes the limitations of the research undertaken as well as the recommendations for future work.

REFERENCES

Abdullah, Khan, M. H., & Srivastava, R. (2015). Flexibility: A Key Factor To Testability. *International Journal of Software Engineering & Applications (IJSEA)*, *6*(1), 89–99.

Aggarwal, V., Mathur, M., & Saraswat, N. (2013). Comprehensive cloud incremental data application migration – A proposed model for cloud migration. *International Journal of Computer Applications in Engineering Sciences*, *3*(1), 20–25.

Ahmed, H. A. S., Ali, M. H., Kadhum, L. M., Fadli, M., Zolkipli, B., & Alsariera, Y. A. (2017). A Review of Challenges and Security Risks of Cloud Computing. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, *9*, 87–91.

Ahn, J. (RightScale). (2015). *Cloud Workload Migration Overview*.

Ampatzoglou, A., Frantzeskou, G., & Stamelos, I. (2012). A methodology to assess the impact of design patterns on software quality. *Information and Software Technology*, *54*(4), 331–346. https://doi.org/10.1016/j.infsof.2011.10.006

Arcelli, D., & Di Pompeo, D. (2017). Applying Design Patterns to Remove Software Performance Antipatterns: A Preliminary Approach. *Procedia Computer Science*, *109*(2016), 521–528. https://doi.org/10.1016/j.procs.2017.05.330

Banerjee, J. (2012). Moving to the cloud: Workload migration techniques and approaches. *2012 19th International Conference on High Performance Computing, HiPC 2012*. https://doi.org/10.1109/HiPC.2012.6507519

Barnes, L., Santos, L., & Bergsma, R. (2016). A Repeatable Cloud-First Deployment Process Model. In *Cloud Security Alliance*.

Beserra, P. V, Camara, A., Ximenes, R., Albuquerque, A. B., Mendonça, N. C., Pós-graduação, P. De, & Aplicada, I. (2012). Cloudstep : A Step-by-Step Decision Process to Support Legacy Application Migration to the Cloud. *6th International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA)*, 7–16.

Bhopale, S. D. (2008). Cloud Migration Benefits and Its Challenges Issue. *IOSR Journal of Computer Engineering (IOSR-JCE)*, 40–45.

Borges, M., Barros, E., & Maia, P. H. (2018). Cloud restriction solver : A refactoring-based approach to migrate applications to the cloud. *Information and Software Technology*, *95*(March 2017), 346–365. https://doi.org/10.1016/j.infsof.2017.11.014

Chan, M. (2017). *Cloud migration - 5 strategies for migrating your applications to the cloud*. https://www.thorntech.com/2017/01/5-strategies-migrating-applications-cloud/

Chen, T.-H., Shang, W., Jiang, Z. M., Hassan, A. E., Nasser, M., & Flora, P. (2014). Detecting performance anti-patterns for applications developed using object-relational mapping. *Proceedings of the 36th International Conference on Software Engineering - ICSE 2014*, 1001–1012. https://doi.org/10.1145/2568225.2568259

CISCO. (2010). *Planning the Migration of Enterprise Applications to the Cloud*. CISCO. https://www.cisco.com/en/US/services/ps2961/ps10364/ps10370/ps11104/Migration_of_Enterprise_Apps_to_Cloud_White_Paper.pdf

Cloud Security Alliance. (2011). Security Guidance for Critical Areas of Focus in Cloud Computing V3.0. In *Cloud Security Alliance*.

Coleman, D., Ash, D., Lowther, B., & Oman, P. (1994). Using metrics to evaluate software system maintainability. *Computer*, *27*(8), 44–49.

Colosimo, M., Lucia, A. De, Scanniello, G., & Tortora, G. (2009). Evaluating legacy system migration technologies through empirical studies. *Information and Software Technology*, *51*(2), 433–447. https://doi.org/10.1016/j.infsof.2008.05.012

Fahmideh, M., Daneshgar, F., Beydoun, G., & Rabhi, F. (2017). Challenges in migrating legacy software systems to the cloud — an empirical study. *Information Systems*, *67*, 100–113. https://doi.org/10.1016/j.is.2017.03.008

Feldman, J. (2013). *Moving Legacy Apps To The Cloud*. InformationWeek. https://www.informationweek.com/software/moving-legacy-apps-to-the-cloud/d/d-id/1108964

Fergus, O., Kingsley, O., & Chioma, U. (2015). Effects of Object-Oriented Programming on Modern Software Development. *International Journal of Computer Applications Technology and Research*, *4*(11), 829–837.

Frey, S., & Hasselbring, W. (2011). An Extensible Architecture for Detecting Violations of a Cloud Environment ' s Constraints During Legacy Software System Migration. *Proceedings of the European Conference on Software Maintenance and Reengineering, CSMR*, 269–278. https://doi.org/10.1109/CSMR.2011.33

Frey, S., & Hasselbring, W. (2010). Model-Based Migration of Legacy Software Systems to Scalable and Resource-Efficient Cloud-Based Applications : The CloudMIG Approach. *CLOUD COMPUTING 2010 : The First International Conference on Cloud Computing, GRIDs, and Virtualization*, *c*, 155–158.

Gamma, E., Helm, R., Ralph, J., & Vlissides, J. (1995). Design Patterns - Elements of Reusable Object-Oriented Software. In *Addison-Wesley Professional*. Addison-Wesley Professional. https://doi.org/10.1093/carcin/bgs084

Gartner. (2011). *Gartner Identifies Five Ways to Migrate Applications to the Cloud*. Gartner. https://www.gartner.com/technology/pressRoom.do?id=1684114

Groves, R. M., Floyd J. Fowler, J., Couper, M. P., Lepkowski, J. M., Singer, E., & Tourangeau, R. (2009). *Survey Methodology* (2nd ed.). John Wiley & Sons, Inc.

Guillén, J., Miranda, J., Manuel, J., & Canal, C. (2013). A service-oriented framework for developing cross cloud migratable software. *The Journal of Systems & Software*, *86*(9), 2294–2308. https://doi.org/10.1016/j.jss.2012.12.033

Gunka, A., Seycek, S., & Kühn, H. (2013). Moving an application to the cloud: an evolutionary approach. *MultiCloud '13 Proceedings of the 2013 International Workshop on Multi-Cloud Applications and Federated Clouds*, 35. https://doi.org/10.1145/2462326.2462334

Hogan, M., Liu, F., Sokol, A., & Tong, J. (2011). *NIST Cloud Computing Standards Roadmap*.

Howarth, F. (2015). *What Are the Risks of Legacy Infrastructure?* Security Intelligence. https://securityintelligence.com/what-are-the-risks-of-legacy-infrastructure/

Hsueh, N., Kuo, J., & Ching-Chiuan Lin. (2009). Object-oriented design : A goal-driven and pattern-based approach. *Softw Syst Model*, *8*(1), 67–84. https://doi.org/10.1007/s10270-007-0063-y

Illa, P. K., Nathani, S., & Veluri, N. (2012). Enterprise Cloud Migration: A Phase Driven Step-by-Step-Strategy intended for Scalability, Elasticity, Agility and Reliability. *International Journal of Computer Application*, *2*(2), 6.

Jaafar, F., Gueheneuc, Y.-G., Hamel, S., & Khomh, F. (2013). Analysing Anti-patterns Static Relationships with Design Patterns. *Electronic Communications of the EASST First Workshop on Patterns Promotion and Anti-Patterns Prevention (PPAP 2013)*, *59*(August 2018), 1–26. https://doi.org/10.14279/tuj.eceasst.59.930.916

Kumar, M., & Jain, A. (2017). Legacy Systems and their evolution through Cloud Services. *Journal of Advanced Computing and Communication Technologies*, *5*(3), 72–75.

Kvale, S., & Brinkmann, S. (2014). *InterViews: Learning the craft of qualitative research interviewing* (Third edit). Sage Publications, Inc.

Lehrig, S., & Becker, S. (2015). The CloudScale Method for Software Scalability , Elasticity , and Efficiency Engineering : a Tutorial. *ACM*, 329–331. https://doi.org/http://dx.doi.org/10.1145/2668930.2688818

Lehrig, S., Eikerling, H., & Becker, S. (2015). Scalability , Elasticity , and Efficiency in Cloud Computing : A Systematic Literature Review of Definitions and Metrics. *ACM*, *1*, 83–92. https://doi.org/http://dx.doi.org/10.1145/2737182.2737185

Li, S.-H., Huang, S.-M., Yen, D. C., & Chang, C.-C. (2007). Migrating Legacy Information Systems to Web Services Architecture. *Journal of Database Management*, *18*(4), 1–25. https://doi.org/10.4018/jdm.2007100101

Lindlof, T. R. (2010). *Qualitative Communication Research Methods* (Third Edit). SAGE Publications, Inc;

Luckow, C. (Zumasys). (2013). *Three Ways to Get to the Cloud*. https://www.zumasys.com/corey-luckow/three-ways-to-get-to-the-cloud/

Martin, R. C. (2017). *Clean Architecture: A Craftsman's Guide to Software Structure and Design*. Pearson Education.

Mell, P. (NIST), & Grance, T. (NIST). (2011). The NIST Definition of Cloud Computing. In *National Institute of Standards and Technology (NIST)*.

Menychtas, A., Santzaridou, C., Kousiouris, G., Varvarigou, T., Orue-Echevarria, L., Alonso, J., Gorronogoitia, J., Bruneliere, H., Strauss, O., Senkova, T., Pellens, B., & Stuer, P. (2014). ARTIST methodology and framework: A novel approach for the migration of legacy software on the cloud. *Proceedings - 15th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2013*, 424–431. https://doi.org/10.1109/SYNASC.2013.62

MI and MINC - Maintainability Index. (2020). Virtual Machinery. http://www.virtualmachinery.com/sidebar4.htm

Mittermeir, R. (1995). Object-Oriented Re-Architecturing. *Lecture Notes in Computer Science*, *989*(September 1995). https://doi.org/10.1007/3-540-60406-5

Perepa, S. (2016). *4 simple steps to cloud adoption*. IBM. https://www.ibm.com/blogs/cloud-computing/2016/08/18/4-simple-steps-to-cloud-adoption/

Pizka, M., & Deißenböck, F. (2007). *How to effectively define and measure maintainability*.

Pree, W., & Koskimies, K. (1999). Rearchitecting Legacy Systems-Concepts and Case Study. In *IFIP — The International Federation for Information Processing*. Springer, Boston, MA. https://doi.org/https://doi.org/10.1007/978-0-387-35563-4_4

Rai, R., Sahoo, G., & Mehfuz, S. (2015). Exploring the factors influencing the cloud computing adoption: a systematic study on cloud migration. *SpringerPlus*, *4*(1), 1–12. https://doi.org/10.1186/s40064-015-0962-2

Rashmi, Mehfuz, S., & Sahoo, G. (2012). A five-phased approach for the cloud migration. *International Journal of Emerging Technology and Advanced Engineering*, *2*(4), 286–291. https://doi.org/10.1007/978-3-642-13645-0

Rimal, B. P., Jukan, A., Katsaros, D., & Goeleven, Y. (2011). Architectural Requirements for Cloud Computing Systems: An Enterprise Cloud Approach. *Journal of Grid Computing*, *9*(1), 3–26. https://doi.org/10.1007/s10723-010-9171-y

RiverMeadow. (2013). Automated Cloud Onboarding & Workload Mobility. In *RiverMeadow Software, Inc.*

Sahay, A. A. (2016). *Peeling Saunder's Research Onion*. https://www.researchgate.net/publication/309488459_Peeling_Saunder's_Research_Onion

Saunders, M., Lewis, P., & Thornhill, A. (2016). *Research Methods for Business Students* (7th ed.). Pearson Education Ltd.

Shadrin, G. (2005). *Three Sources of a Solid Object-Oriented Design*. SYS-CON Media, Inc. http://ima.udg.edu/~sellares/EINF-ES1/OOPrinciplesGShadrin.pdf

Sharma, V. S., & Anwer, S. (2013). Detecting performance antipatterns before migrating to the Cloud. *Proceedings of the International Conference on Cloud Computing Technology and Science, CloudCom*, *1*, 148–151. https://doi.org/10.1109/CloudCom.2013.166

Singh, A. K., Mishra, R., Ahmad, F., Sagar, R. K., & Chaudhary, A. K. (2012). A review of cloud computing open architecture and its security issues. *International Journal of Scientific & Technology Research*, *1*(6), 65–67.

Singh, H., & Hassan, S. I. (2015). Effect of SOLID Design Principles on Quality of Software: An Empirical Assessment. *International Journal of Scientific & Engineering Research*, *6*(4), 1321–1324.

Smith, C. U., & Williams, L. G. (2002). Software Performance AntiPatterns: Common Performance Problems and their Solutions. *Performance Engineering Services and Software Engineering Research*, *2*, 797–806.

Smith, C. U., & Williams, L. G. (2003). More New Software Performance Antipatterns : Even More Ways to Shoot Yourself in the Foot. *Performance Engineering Services and Software Engineering Research*.

Sommerville, I. (2007a). *Software Engineering 8* (8th ed.).

Sommerville, I. (2007b). *Software Engineering Eighth Edition* (8th ed.). Pearson Education Limited.

Sommerville, I. (2011). *Software Engineering Ninth Edition* (9th ed.). Pearson Education Limited.

Strom, D. (2015). *The Impact of the End of Windows XP on Your Enterprise*. Security Intelligence. https://securityintelligence.com/the-impact-of-the-end-of-windows-xp-on-your-enterprise/

Subburaj, R., Jekese, G., & Hwata, C. (2015). Impact of Object Oriented Design Patterns on Software Development. *International Journal of Scientific and Engineering Research*, *6*(2), 961–967.

Varia, J. (2010). Migrating your existing applications to the AWS cloud. In *A Phase Driven Approach to Cloud Migration*.

Vizteams. (2014). *Issues and Challenges Facing Legacy Systems*. Vizteams. https://www.vizteams.com/blog/issues-and-challenges-facing-legacy-systems/

Walliman, N. (2011). *Research Methods - The Basics*. Routledge Taylor & Francis Group.

Wielki, J. (2015). *An analysis of the opportunities and challenges connected with utilization of the cloud computing model and the most important aspects of the migration strategy*. *5*, 1569–1574. https://doi.org/10.15439/2015F93

Xiao, Z., Chen, Q., & Luo, H. (2014). Automatic scaling of internet applications for cloud computing services. *IEEE Transactions on Computers*, *63*(5), 1111–1123. https://doi.org/10.1109/TC.2012.284

Zoufaly, F. (2002). *Issues and Challenges Facing Legacy Systems*. Developer.Com. https://www.developer.com/mgmt/article.php/11085_1492531_2/Issues-and-Challenges-Facing-Legacy-Systems.htm

BIODATA OF STUDENT

Muhammad Ehsan Rana is a PhD candidate in Software Engineering at Universiti Putra Malaysia. His research focuses on creating an implementation framework based on design refinements for systematic migration of legacy applications to cloud. Muhammad Ehsan Rana possesses vast experience in academic management, lecturing and quality assurance. He holds Masters in Computer Science and Masters in Mathematics degrees and has a vast project supervision experience. He is associated with Asia Pacific University of Technology & Innovation (APU) Malaysia since December 2008. His research interest includes software architecture, cloud computing, design patterns and privacy for electronic health records. He has authored and co-authored many journal papers and conference publications in the areas specified. Prior to lecturing, he worked as Software Quality Manager for an international software development company which specializes in ERP solutions. As a software quality manager, he managed and implemented organization wide quality policies, quality assurance plan and risk management. Currently, Ehsan's research explores Internet of Things (IoT) and Blockchain.

LIST OF PUBLICATIONS

Published Articles

Rana, M. E., & Rahman, W. N. W. A. (2017). The Effect of Applying Software Design Patterns on Real Time Software Efficiency. Proceedings of the Future Technologies Conference (FTC) 2017 (pp. 691-697). Vancouver, Canada.

Rana, M. E., & Rahman, W. N. W. A. (2018). A Review of Cloud Migration Techniques and Models for Legacy Applications: Key Considerations and Potential Concerns. Advanced Science Letters, 24(3), 1708-1711.

Rana, M. E., Rahman, W. N. W. A., & Ahmed M. (2018). Evaluating Design Pattern Based Solutions with their Equivalent Simpler Solutions to Promote Maintainability in Software. Advanced Science Letters, 24(3), 1702-1707.

Rana, M. E., Rahman, Ahsan, M., & W. N. W. A. (2019). A Detailed Analysis on the Use of Gamification in Cloud Based Learning Management Systems. Journal of Theoretical and Applied Information Technology, 97(8), 2179-2194.

Rana, M. E., Rahman, W. N. W. A., Murad, M. A. A., & Atan, R. B. (2019). The Impact of Flyweight and Proxy Design Patterns on Software Efficiency: An Empirical Evaluation. International Journal of Advanced Computer Science and Applications, 24(3), 161-170.

Rana, M. E., Farooq U., & Rahman, W. N. W. A. (2019). A Study of Scalability Enhancement for Cloud-based Applications using Software Oriented Methods. International Journal of Engineering and Advanced Technology, 8(6), 4208-4213.

Articles Submitted For Publication

The Impact of Mediator and Observer Design Patterns on Software Reliability: An Empirical Evaluation

UNIVERSITI PUTRA MALAYSIA

STATUS CONFIRMATION FOR THESIS / PROJECT REPORT AND COPYRIGHT

ACADEMIC SESSION : First Semester 2020/2021

TITLE OF THESIS / PROJECT REPORT :

IMPLEMENTATION FRAMEWORK BASED ON DESIGN REFINEMENTS FOR SYSTEMATIC MIGRATION OF LEGACY APPLICATIONS TO CLOUD

NAME OF STUDENT: MUHAMMAD EHSAN RANA

I acknowledge that the copyright and other intellectual property in the thesis/project report belonged to Universiti Putra Malaysia and I agree to allow this thesis/project report to be placed at the library under the following terms:

1. This thesis/project report is the property of Universiti Putra Malaysia.
2. The library of Universiti Putra Malaysia has the right to make copies for educational purposes only.
3. The library of Universiti Putra Malaysia is allowed to make copies of this thesis for academic exchange.

I declare that this thesis is classified as :

*Please tick (√)

☐	**CONFIDENTIAL**	(Contain confidential information under Official Secret Act 1972).
☐	**RESTRICTED**	(Contains restricted information as specified by the organization/institution where research was done).
☐	**OPEN ACCESS**	I agree that my thesis/project report to be published as hard copy or online open access.

This thesis is submitted for :

☐ **PATENT** Embargo from _____ (date) until _____ (date)

Approved by:

(Signature of Student)
New IC No/ Passport No.:

Date :

(Signature of Chairman of Supervisory Committee)
Name: Associate Professor
Dr. Wan Nurhayati Wan Ab. Rahman

Date :

[Note : If the thesis is CONFIDENTIAL or RESTRICTED, please attach with the letter from the organization/institution with period and reasons for confidentially or restricted.]