

**ARCHITECTURAL DESIGN DECISION KNOWLEDGE
MANAGEMENT SYSTEM**

Shashi Lakshan Chandrasinghe

(179311P)

M.Sc. in Computer Science

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

May 2019

ARCHITECTURAL DESIGN DECISION KNOWLEDGE MANAGEMENT SYSTEM

Shashi Lakshan Chandrasinghe

(179311P)

This dissertation submitted in partial fulfillment of the requirements for the Degree of
MSc in Computer Science specializing in Software Architecture

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

May 2019

DECLARATION

I declare that this is my own work and this MSc Thesis Project Report does not incorporate without acknowledgement of any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works.

.....

Shashi Lakshan Chandrasinghe

.....

Date

I certify that the declaration above by the candidate is true to the best of my knowledge and that this project report is acceptable for evaluation for the MSc Thesis (CS6997).

.....

Dr. Indika Perera

.....

Date

ACKNOWLEDGEMENT

My sincere appreciation is dedicated to my research supervisor Dr. Indika Perera for his valuable support and guidance to complete the MSc Thesis Report successfully and also I would like to express my gratitude to my friends and colleagues in MSc' 17 who supported to complete the task.

Finally, I would like to thank the academic and non-academic staff of Department of Computer Science and Engineering of the University of Moratuwa.

ABSTRACT

The software systems typically fail and deviate from its prescriptive architecture due to various reasons such as incorrect architectural design, lack of experience and lack of domain knowledge. After precise software requirements are gathered from customers, those requirements should be converted into an appropriate design. Suppose if any inappropriate design is constructed from these requirements, it may cause to reconstruct the system implementation. So, a set of good architectural design decisions form a good system architecture and those architectural design decisions should be documented or stored as knowledge bases to use further.

Various methodologies exist to store architectural design decisions and trace them. Most of them have some drawbacks such as lack of time to gather and store and additional cost to maintain such knowledge bases. As the key objective, this report proposes an ontological knowledge management system to solve above mentioned problems in software engineering industry for avoiding the extra costs to redevelop or refine the software system implementation.

Though the implemented solution is ontology-based knowledge management system, it seems to be a simple web application to the end user. User-friendly web interfaces are implemented to store and retrieve the architectural design decisions, based on completed or already initiated software projects. Those design decisions would be useful for the professionals who design the effective software architecture designs.

Finally, empirical and Likert questionnaires were conducted to prove that the implemented solution works perfectly as a solution for the stated problems and this report ends mentioning some limitations and future work with relevant to ontological knowledge management systems and its technologies.

Keywords: knowledge bases, ontology, architectural design decisions

TABLE OF CONTENTS

DECLARATION	i
ACKNOWLEDGEMENT	ii
ABSTRACT	iii
TABLE OF CONTENTS	iv
LIST OF FIGURES	vii
LIST OF TABLES	ix
LIST OF ABBREVIATIONS	x
CHAPTER 1	1
INTRODUCTION	1
1.1 Background	2
1.2 Research Question	3
1.3 Motivations to Solve the Problem	4
1.4 Research Scope	5
1.5 Research Objectives	5
1.6 Overview of the Document	5
CHAPTER 2	7
LITERATURE REVIEW	7
2.1 Importance of Architectural Design Decisions	8
2.2 Challenges in Architectural Design Decisions	9
2.3 Some Knowledge Bases for Architectural Knowledge	11
2.4 Why Ontology Driven Software Engineering is Needed.	13

2.5 Strengths and Weaknesses of Having Ontology Driven Software Engineering	16
2.6 Designing Ontology-based Knowledge Management System.	16
2.7 Problems in Ontology Development	17
2.8 Ontology Implementation	17
2.9 Ontology Validation	18
2.10 Basic Introduction to Ontology and Its Languages	19
2.10.1 RDF	20
2.10.2 RDFS	21
2.10.3 OWL	22
2.11 Ontology Generation	25
2.12 Ontology Supporting Tools	29
CHAPTER 3	31
METHODOLOGY	31
3.1 Qualities of a Software Architecture	32
3.2 Proposed Knowledge Management System	32
3.3 Automate the Proposed Knowledge Management System	33
CHAPTER 4	35
SOLUTION ARCHITECTURE AND IMPLEMENTATION	35
4.1 Steps to Model the Ontology	36
4.1.1 Step 1 - Determine the Domain and the Scope	37
4.1.2 Step 2 - Define the Classes and Taxonomy	37
4.1.3 Step 3 - Define the Properties of Classes	38
4.1.4 Step 4 - Define the Relationships Between Classes	40
4.2 Implementing the Ontology System.	42
4.2.1 Technology Stack Used to Implement the Ontology System.	44

4.2.1 OWL File Generation	46
4.2.1 Implementation of Data Retrieval Functionality	47
CHAPTER 5	51
EVALUATION	51
5.1 Structural Evaluation	52
5.1.1 Ontology Classes	52
5.1.2 Ontology Data Properties	53
5.1.3 Individuals in Ontology	54
5.2 Evaluate the system performance	56
5.3 System Evaluation	59
5.3.1 Functionality Evaluation	59
5.3.2 User Interface Evaluation.	62
CHAPTER 6	65
CONCLUSTION	65
6.1 Research Contribution	66
6.2 Research Limitation	67
6.2 Future Work	67
REFERENCES	69
Appendix A How Ontology Concepts Affect to the User	74
Appendix B User Interface Evaluation	76
Appendix C Sample Java Classes for Proof of Implementation	78

LIST OF FIGURES

Figure 2-1 Ontology Categorization in SDLC Phases	14
Figure 2-2 Design Architecture of Knowledge Management Portal	15
Figure 2-3 Ontology Development Process Using Apache Jena API	18
Figure 2-4 Ontology Evaluation Dimensions	19
Figure 2-5 Sample RDF Format	21
Figure 2-6 Sample RDFS Format	22
Figure 2-7 Sample OWL Content	24
Figure 2-8 Ontology Generation Process	25
Figure 2-9 Data Extraction Process Automation Using Apache Jena	26
Figure 2-10 Class Level Hierarchy in Family Ontology	27
Figure 2-11 Sample Visualization for People Ontology Using ‘Jambalaya’ Tab in Protégé	30
Figure 3-1 : flow of developing the ontology-based knowledge management system.	34
Figure 4-1 Properties of Decision Class	39
Figure 4-2 Class Diagram of Ontology System	41
Figure 4-3 System Architecture for Ontology System	42
Figure 4-4 Web Interface to Insert Architectural Design Decisions	43
Figure 4-5 Web Interface to Insert New Project Details	44
Figure 4-6 Package Structure of Source Code	45
Figure 4-7 Defining Ontology Classes Through Jena API	46
Figure 4-8 Defining the Data Property for Project class	46
Figure 4-9 Source Code for 'Project' individual	47
Figure 4-10 User Interface for Basic Search	48
Figure 4-11 User Interface for Advanced Search	49
Figure 5-1 Ontology Class in Protégé	53
Figure 5-2 Data Properties with Domains and Ranges	54
Figure 5-3 Individual Values for Project Ontology Class	55
Figure 5-4 Individual Property Values for Project Class	55
Figure 5-5 SPARQL Search Query	56
Figure 5-6 Calculate Execution Time Programmatically	57

Figure 5-7 Consecutive Execution Times on Search Functionality	57
Figure 5-8 A Sample Results of a SPARQL Query	58
Figure 5-9 Execution Times Over Size of OWL Files	59
Figure 5-10 Functionality Evaluation Based on Questionnaire	61

LIST OF TABLES

Table 2-1 Sample Data of Architectural Knowledge	12
Table 5-1 OWL File Sizes Over Execution Times	58
Table 5-2 Appendix A - User Feedbacks	60
Table 5-3 Appendix C - User Feedbacks	63

LIST OF ABBREVIATIONS

Abbreviation	Description
SDLC	Software Development Life Cycle
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
OWL	Web Ontology Language
JESS	Java Expert System Shell
RDQL	RDF Data Query Language
OWL-QL	OWL Query Language
W3C	World Web Consortium
SPARQL	Simple Protocol and RDF Query Language
OOP	Object Oriented Programming
RAM	Random Access Memory