Develop Framework for Selecting Best Software Development Methodology

Jyoti Verma, Sunita Bansal, Himanshu Pandey

Abstract- There is many software development methodologies exist that is used to control the process of developing a software system. No exact system was found which could help the software engineers for selection of best software development methodology. This paper present the framework of expert system combined with Likert scale. With the help of Likert scale we create a rule based model and assign some weighted score to each methodology and develop a tool named as ModSet which will help the software engineers to select an appropriate development methodology that may increase the probability of system success.

Keywords:, Expert System, Likert Scale, Mod Set, Rule Based System, Software Development Methodology.

1. Introduction

The problem of selecting a suitable software development methodology for the project has been addressed in different way by many researchers but there is no proper framework has developed. This paper extends prior work by considering the expert system with the Likert scale to select best software development methodology.

The goal of developing a rule based expert system is to come up with the result that would help to establish which software develement methodology is particularly suitable for a particular project. In other words obtaining one clear anewer was not the aim, but rather obtaining guidance in choosing from a known set of methodologies. The model selected should suit the characteristics of the given project should correspond to the criteria set out in the selection of methodology, the criteria should be stored in the knowledge base of the system. Then to obtain information about a suitable methodology, the user of such a program would have to determine the characteristics of their project by answering the set of questions asked by the system.

1.1 Software Development Methodologies

A software development methodology describes a environment that is used to organize, plan and direct the process of developing a software system. There are many software development methodologies and these methodologies contain some basic stages of software development life cycle. These stages are planning, analysis, design, implementation and maintenance. In this paper we compare 3 software development methodologies. These are waterfall, spiral and prototype methodology

1.1.1 Waterfall Development Methodology

The waterfall methodology is the linear approach in which development is seen as flowing continuously downwards with the phases named as requirement analysis, design, implementation, testing, integration and maintenance. Each phase of this model has well defined starting and ending criteria.

Waterfall methodology is most appropriate when-

- Requirements are well documented,clear and fixed from the star ting stage.
- End product is stable.
- Technology is understood and static.
- The project is short.
- Not good methodology for complex and object oriented project.
- Difficulty in accommodating changes after project development.

1.1.2 Spiral Development Methodology

The spiral methodology is a risk-driven process model generator for software projects.

Spiral methodology is most appropriate when-

- Risk evaluation is important.
- Customer is not sure of their requirements.
- Requirements are complex and need clarity.
- It can work well with the changing user requirements.
- This methodology is mainly used for large projects.
- Budget is high.
- Little documentation is required as compared to waterfall methodology.

1.1.3 Prototype Development Methodology

In Prototype development methodology firstly the working prototype is developed instead of developing the actual software. In this methodology to understand the requirement a prototype is built before design and coding is begin.

Prototype methodology is most appropriate when-

- Prototyping is an basic idea for complicated and large systems for which there is no manual process or existing system to help determining the requirements.
- This methodology is used when it is difficult to collect all the requirements of customer.

1.2 Expert system

Expert system can be used in various fields [8]. In this paper we extend the prior work by considering the expert system in the field of software development model. The expert system is a computer program that imitates the decision making ability of the human experts. There are three components of expert system i.e. user interface, knowledge base and inference engine.

The knowledge base contains all the knowledge and rules about a particular problem and provides it to the inference engine when they needed.

The inference engine is the program that analyzes the problem and derive conclusion by applying logical processing.

The user interface works as an interface between a user and the expert systems.

2. Literature Study

Literature study illustrate that many work has been taken on different aspects of software development methodologies but there is no proper framework and tool being developed for the selection of best software development methodologies. In 2005 M. AYMAN AL AHMAR[1] present a prototype expert system supported with object oriented modeling that helps in software development methodology selection process. Abdur Rashid khan, Zia Ur Rehman and Hafeez Ullah Amin[2] focused some technologies like fuzzy logic, certainty factors and analytical hierarchy factors to develop expert system and the developed expert system named as ESPMS (expert system process model selection). This system was developed using expert system for text animation as a development tool. Much work has been done in the area of software development methodologies but most of these works are the comparative study between these methodologies[3][10]. In 2010 Nabil Mohammed Ali Munauar and A Govardhan[4] made a comparison between five different methodologies and show the features and defects of each methodology. When the software engineers select a development model, risk is an important factor which affects the selection process of software development methodology. Haneen Hijazi and Thair Khdour[5] investigate the risk that exist in the development methodologies and define risk management. In this paper we use rule based expert system as a major component to differentiate between methodologies. Literature study reveals that expert system can be used in various fields and knowledge base is a component of expert system which stores knowledge about a particular problem area. Manuj Darbari, N. Dhanda[7] presents the model driven knowledge representation framework. Sunita Bansal and Manuj Darbari[6,14] defines a knowledge base expert system for handling business dynsmics. In 2012 Sudhakar Tripathi, Arvind kumar Tiwari and R.B.Mishra[11] create a rule base model for clustering gene expression data. In 2009 M.S.Josephine, Dr. K.Sankara Subramanian[13] used an expert system for software error detection and correction. In this paper a rule based expert system is created to combine with Likert scale measurement that guides in the selection process of the software development[9] methodology.

3. Proposed framework

The proposed framework is shown below:

Expert system model for selecting suitable software development methodology for project-

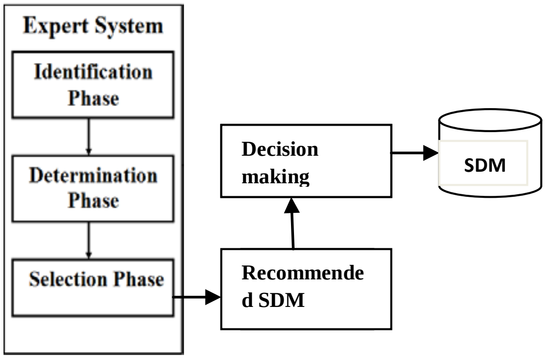

Figure1.Proposed Framework

4. Likert Scale

Likert scale is a type of rating scales which is used to measure the performance directly. Likert Scale is a five (or seven) point rating scale which allows the user to express their thoughts means how much they agree or disagree with a problem statement. User gives their responses by choosing a Likert items. A Likert item is a word or statement which the user is asked to evaluate according to the given criteria. Likert items are used to measure the degree of agreement or disagreement. We use 5 Likert items i.e. poor, fair, average, good and excellent.

Features	Poor	Fair	Average	Good	excellent
Req. specification	1	2	3	4	5
Complexity of system	1	2	3	4	5
Time schedule	1	2	3	4	5
Cost	1	2	3	4	5
Documentation	1	2	3	4	5
Project size	1	2	3	4	5
Change incorporated	1	2	3	4	5

Table1: Likert scale measurement scale

4.1 Selection Parameters for Selecting Best Development Methodology

No one methodology is ideal so we design a framework for selecting a methodology which depends on many factors and project features and selection parameters. These selection parameters[11] are:

requirement specification, complexity of system, time schedule, cost, documentation, project size, and change incorporated.

4.2 Comparison among Waterfall, Spiral and Prototype Model

Features	Waterfall	Spiral	Prototype
Req. specification	Beginning	Frequently changed	Beginning
Complexity of system	Simple	Moderate	Complex
Time schedule	Very low	Long	Long
Cost	Low or almost as estimated	High or above budget	Expensive
Documentation	Necessary	Yes but not much	Yes
Project size	Large scale	Low to medium scale	Large scale and complex
Change incorporate	Difficult	Easy	Easy

Table2: Comparison Table

5. Rule Based Model for Selecting Software Development Methodology

Rule based model have been generated with the help of Likert scale measurement.

Rule1:

IF Requirement specification <= 1 and

Complexity of system <= 2 and

Time schedule <= 5 and

Cost <= 2 and

Documentation <= 5 and

Project size <= 4 and

Change incorporated <= 1 and

THEN Waterfall Model (Score <= 20)

Rule2:

IF Requirement specification <= 4 and

Complexity of system <= 3 and

Time schedule <= 4 and

Cost <= 4 and

Documentation <= 3 and

Project size <= 3 and

Change incorporated <= 4 and

THEN Spiral Model (Score <= 25)

Rule3:

IF Requirement specification <= 2 and

Complexity of system <= 5 and

Time schedule <= 4 and

Cost <= 5 and

Documentation <= 4 and

Project size <= 5 and

Change incorporated <= 4 and

THEN Prototype Model (Score <= 29)

5.1 Decision Making Tool

The decision for selecting particular software development methodology depends on the selection parameters and final score assigning to the methodologies. The prioritization and policy for selecting the development methodology might be refined; the important consideration is that the decision is made explicitly as shown below:

Constraints priority

C1: If documentation is necessary and requirements are stable or unchanging, then it should automatically select the waterfall methodology.

C2: If project risk is high then it should automatically select the spiral methodology.

C3: If requirement are undefined then it should automatically select the prototype methodology.

Constraints	Priority
C1: If documentation is necessary and requirements are stable or unchanging then it should automatically select the waterfall methodology.	5
C2: If project risk is high then it should automatically select the spiral methodology.	7
C3: If the requirement are undefined then it should automatically select the prototype methodology	8

Table3: Constraints and priorities for SDM

5.2 Methodology Selection Tool: Modset

In order to stimulate the methodology selection rule we have implemented a tool provide automated support for decision of development methodology selection. We have implemented a tool called "**ModSe**t" selector shown in figure. A tool consists of 5 basic selections with check boxes and conditions.

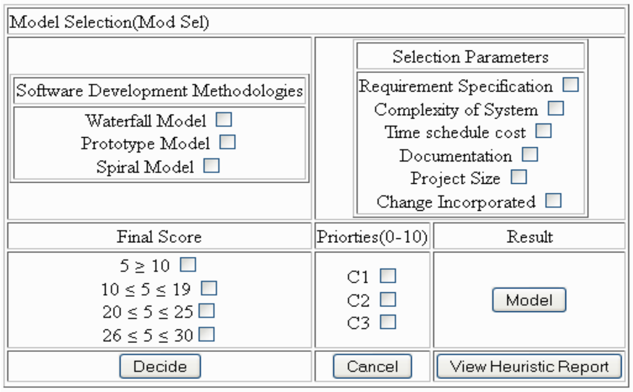

Figure 2. Model Selection Tool

The first block represents the software development methodologies. The next block represents selection criteria, final score, dynamic priority allocation and results. Expect the result block all the blocks have check boxes and the priorities can be assigned between 0 and 10. After making the entire selections "Decide" button is pressed and the result is displayed in the result text box giving the model selected. "View Heuristic Report" button gives the entire report with situational analysis.

6. Conclusion and Future Work

This study proposed a rule base expert system combined with Likert scale measurement that will become a base for the software engineers in the selection of best software development methodology for the project. In this paper, we provide a tool that will help in the selection process of development methodology. The future work will include more software development methodologies and more selection parameters.

REFERENCES

1. M.Ayman Al Ahmar, "Rule based expert system for selecting software development methodology", Journal of theoretical and Applied Science, 2005-2010.

2. Abdur Rashid khan, Zia Ur Rehman and Hafeez Ullah Amin, "Knowledge based system's modeling for software process model selection", IJACSA, Vol.2, No.2, Feb2011.

3. Vishwas Massey, K.J.Satao "Comparing Various SDLC Models And The New Proposed Model On The Basis Of Available Methodology", IJARCSSE, Vol.2, Issue4, April2012.

4. Nabil Mohammed Ali Munassar and A. Govardhan, "A Comparison Between Five Models Of Software Engineering", IJCSI, Vol.7, Issue5, September2010.

5. Haneen Hijazi, Tthair Khdour and abdulsalam alarabeyyat, "A Review of Risk Management in Different Software Development Methodologies"International Journal of Computer applications, Vol.45-No.7, May2012.

6. Dr. Sunita Bansal and M Darbari, "Designing and Knowledge Based Expert System for Handling Business Dynamics", International Journal of Scientific & Engineering Research Volume2 , Issue 11 , March -2011.

7. M Darbari, N Dhanda, "Applying Constraints in Model Driven Knowledge Representation Framework" International Journal of Hybrid Information Technology 3 (3),4, 2010.

8. Folorunso, I. O, Abikoye, O. C.,Jimoh, R. G. and Raji, K.S, "A Rule-Based Expert System for Mineral Identification", Journal of Emerging Trends in Computing and Information Sciences, Vol.3, No.2, February 2012.

9. M Darbari, S Medhavi, AK Srivastava "Development of effective Urban Road Traffic Management using workflow techniques for upcoming metro cities like Lucknow (India)", (2)4, 2008.

10. Dinesh Ch. Jain and Shikha Maheshwari, "A Comparative Analysis of Different types of Models in Software Development Life Cycle", IJARCSSE, Vol.2, Issue5, May 2012.

11. IA Siddiqui, M Darbari, "Application of Use Case for Identification of Root Cause of the Dependencies and Mutual Understanding and Cooperation Difficulties in Software Systems, International Journal of Applied Software Engineering, 4, 10-20.

12. Sudhakar Tripathi, Arvind kumar Tiwari and R.B.Mishra, "Rule based mosel for clustering gene expression data", International conference on Artificial Intelligence & Soft Computing, December-2012.

13. M.S.Josephine, Dr. K.Sankara Subramanian, "Software Error Detection And Correction Using Layer Base Approach In Expert System", International Journal Of Reviews In Computing, Vol 4, 2012.

14. Dr. Sunita Bansal, M Darbari, "Application of Multi Objective Optimization in Prioritizing and Machine Scheduling: a Mobile Scheduler Toolkit", International Journal of Advance Information Systems, Vol 3, No. 2, 2012.