Evaluation of Software Architecture by Weight Metric for an Internet Banking System

[1] A. Meiappane, [2] Dr. V. Prasanna Venkatesan, [3] N. Roshini, [4] S. Nivedha, [5] R. Maheswari

Abstract— Software architecture has a measurable part which exemplifies the early design decisions covering several perspectives and also it impacts the quality attributes of a system. Therefore the evaluation of software architecture plays a prominent role in determining the degree to which architecture satisfies its quality requirement and also to expose architecture risk that potentially inhibit the achievement of an organization's business goals.In this paper , we present some identification of pattern in an internet banking system for efficient access. The pattern for an internet banking system will be allowing the system to be load free , the transactions will be speed and faster processing .We have also implemented some patterns for an internet banking system for server load free functions. Then the rest of the paper deals with software architecture evaluation for an internet banking system using some evaluation method. Finally, the evaluation results are represented in a weight metric form. And our next work is to propose an refined internet banking architecture.

Index Terms—Software architecture, pattern, evaluation, quality functions, SAAM, ATAM, Performance, add beneficiary, money transfer

1. INTRODUCTION

One of the challenging issues in developing a software system is maintaining an appropriate level of quality. Thus, the prediction of the quality of a software system from a high level design description has evolved since the past three decades [1].

This has resulted in the emergence of an important quality assurance technique known as software architecture (SA) evaluation. The need for this number of formal and systematic measures are implied to ensure the quality issues (such as performance, security, maintainability, usability) in a software intensive system. The key aspect of evaluating software architecture is to assess whether the architecture chosen is capable of fulfilling the required quality factors and to identify potential risks [2]. The various existing methods in SA evaluation process are Software Architecture Analysis.

- A. Meiappane, Research Scholar and Dr. V. Prasanna Venkatesan, Associate Professor are from Pondicherry University, Puducherry, India. *E-mail:* [1] *auromei@yahoo.com*
- [3,4,5] *Students, Department of Information Technology, Sri Manakula Vinayagar Engineering College, Puducherry, India.*

Method (SAAM) [3], Architecture Trade off Analysis Method (ATAM)[4], Architecture-Level Maintainability Analysis(ALMA) [5], Cost Benefit Analysis Method(CBAM), Active Recursive for Intermediaries Design(ARID). The evaluation methods are based on various techniques such as scenario elicitation, mathematical model, and architecture centric model.

Among the above evaluation techniques scenario based evaluation is a prominent and majorly adopted so far among the architects. The SA evaluation methods mostly resemble each other structurally but incorporate various differences in their activities. This paper focuses on evaluation of a pattern oriented software architecture using scenario-based methods. The modern SA is often composed form architectural styles and design patterns in order to handle different quality attributes. The existing SA evaluation method doesn't incorporate any experimental study to assess the strength and weakness of design patterns and architectural styles which comprises of the building blocks of a software architecture .They assume that the strength and weakness of SA building blocks are already known.

This paper deals with evaluation of pattern oriented software architecture of an e-banking system. First phase of this paper inculcates about the patterns identified in the e-banking system after undergoing proper survey. These patterns hold their base from the GOF design patterns. They are meant to facilitate reusability which is the major concern of applying patterns and also other quality

attributes such as performance, scalability and so on. Second phase of this paper focuses on the evaluation of the patterns by using scenario-based evaluation technique. Scenario is elicited based on the identified patterns and a weight age is assigned based on the interaction among the scenarios. A final conclusion is made by comparing the assigned weight age of an architecture in which the patterns are applied and the existing architecture.

2. BACKGROUND AND MOTIVATION

2.1 Quality attributes

In software engineering, a software system deals with both functional and non-functional quality requirements. The functional quality requirement defines a function of the software system or its components. They may be any data manipulation, processing, calculation and other specific functionality which defines what a system is supposed to achieve. Functional requirements are supported by non-functional requirements, which are also known as quality requirements. The non-functional requirements impose constraints on the design or implementation such as performance requirements, security or reliability. In general, functional requirement represents "what the system must do", whereas non-functional requirements are "how the system shall be". The implementation of functional requirements is detailed in the system design, while that of non-functional requirement is expressed in the software architecture [7]. This paper deals with the evaluation of the non-functional attributes of an e-banking system's software architecture.

2.2 Patterns

Christopher Alexander says, "*Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.*" Though, Christopher described patterns in terms of buildings it was true even in the case of object-oriented design patterns. A pattern in software engineering has four essential elements such as pattern name, problem, solution, consequences. In software engineering design patterns are classified into creational, structural and behavioural patterns also known as Gang of Four pattern (GoF). And then further their scope is sub-divided into classes or objects. Creational pattern deals with the process of object creation. Structural pattern deals with the composition of classes or objects. Behavioural pattern is regarding interaction among the classes or objects and their responsibilities [12]. The above categorization of patterns lays the foundation for other patterns.

3. PATTERNS IN INTERNET BANKING

Pattern mining has been undertaken in internet banking system using BPM, which is used to analyze the processes in e-banking and then the cross-cutting concerns are applied. This paper deals with the evaluation of the following patterns in internet banking system. The patterns mined have their base from the GoF (Gang of Four) design patterns. The following are the patterns are assessed using scenario-based evaluation techniques:

(a)Enquiry Pattern,
(b) Money Transfer Pattern,
(c) Payment Pattern
(d)Add Beneficiary Pattern
(e)Request Pattern

(a) Enquiry pattern is mined from the visitor pattern which is a type of behavioral pattern. Behavioral pattern deals with the interaction and responsibility between the objects. It's concerned with the algorithms and pattern of communication among the objects. Visitor pattern eases the process of defining new operations. It represents an operation performed on the elements of an object structure. New operations can be added without changing the class to which an element belongs. In e-banking, enquiry pattern deals with enquiry of various statements like mini-statement enquiry, credit-card statement enquiry and so on. Various kinds of enquirers can be handled by a single visitor class.

(b) Money transfer pattern is mined from strategy pattern, which is a type of behavioral pattern. Strategy pattern in meant for implementing interchange algorithms depending upon the client. It defines a family of algorithms and make them interchangeable. In e-banking, money transfer deals with implementing different types of money transfer. The transfer of money can be made to an account which belongs to the same bank or to accounts in other banks or it may be a visa payment.

(c) Payment Pattern is mined from the strategy pattern, which is a type of behavioral pattern. The scenarios covered by the payment pattern in an e-banking system are credit card bill payment and various other commercial bill payments can be performed.

(d) Add Beneficiary Pattern reflects the factory pattern. The factory pattern is based on the creational patterns. It eases the instantiation process by abstracting them. The instantiation process consists of creation,

composition and representation of the classes and objects. Add beneficiary pattern in e-banking focuses on the following scenarios such as adding payee for various types of payments, mobile recharge and so on.

(e) Request Pattern is mined based on the command pattern which is a type of behavioral pattern. It deals with messaging patterns among the classes by using objects. The request between the elements are encapsulated as an object and then processed. It also deals with the different requests

4. EVALUATION METHODS

The architecture is the basis for any software system because it will be focusing on the major elements and their interaction with each other in a system. The most important thing for a system is the architecture so it must be builds on a solid foundation. So, the software architecture must be designed in an efficient manner to meet all the requirements of users in business process [8].

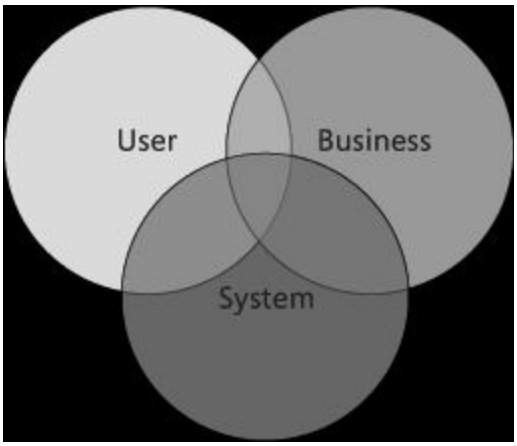

Figure 1

The evaluation of software architecture has been a common practice which is used to deliver a quality product to the stakeholders. Architectural evaluation will greatly reduce the development cost and effort and identify the potential risks for the system .The software architecture evaluation process will provide assurance to the developers that they had given a quality product to the users which will meet both the functional and nonfunctional attributes .So there are several software architecture analysis method which is again classified as scenario based model, attribute based approach and quantitative model. We have given the overall idea of the evaluation methods in a tabular format as follows:

4.1 Evaluation of software architecture for an internet banking system

Nowadays all the customers prefer online banking because of their convenience, cost saving and control purposes. Before itself the architecture patterns are used in internet banking system architecture and we are going to evaluate it using some evaluation method.

The evaluation methods will be having the following steps:

1. The architectural scenarios are developed
2. Scenarios are classified as direct / indirect
3. Assign weight age to the modules
4. Evaluate the architecture
5. Generate the result

The evaluation of software architecture of an internet banking system is calculated using the formula

Module Weight age = Weight age of scenario / No of functionality

Table 1- comparison of SA evaluation methods

METHODS	SAAM	ATAM	CBAM	ALMA	FAAM
QUALITY ATTRIBUTES ASSESSED	Modifiability	Trade-off between attributes	Costs, Benefits, and Schedule Implications	Modifiability	Interoperability And Extensibility
PROCESS DESCRIPTION	Rational	Good	Reasonable	Reasonable	Very good Exhaustive process flow
METRIC AND TOOL SUPPORT	Scenario arrangement (direct vs. indirect ones)	Identifying the areas of high Impending difficulty Open for any architectural explanation	Time and Costs	Impact estimation, Modifiability prediction Model	Various focused tables and Diagrams
STRENGTH	Identifying the areas of high Impending difficulty Open for any architectural explanation	Scenarios generation based on Requirements Applicable for static and dynamic properties Quality utility tree	Provide business measures for Particular system changes. Make clear the uncertainty linked with the estimates	Scenario creation stopping norm	Various specialized tables and Diagrams
WEAKNESS	Not a clear worth metric Not supported by techniques for performing the steps	Requires detailed technical awareness	Identifying and trading costs and benefits can be done by the participants in an open manner	Various specialized tables and Diagrams	Only partially proven in one Particular atmosphere. Concentrates o n static properties

CONCLUSION

This paper deduces that the patterns indentified in an internet banking system can be assessed effectively using scenario based evaluation techniques. Final result is projected as a comparison between the existing internet banking system and the pattern implemented system by using weight metrics. This paper also provides a comparison among various existing software architecture evaluation techniques. Future work is to evaluate the pattern applied internet banking system in terms of quality attributes using unit operation.

Reference

[1] L. Bass, P. Clements and R. K. Kaman. Software Architecture in Practice. SEI Series in Software Engineering. Addison-Wesley, 1998. ISBN 0-201-19930-0.

[2] Buchman, F.; Meaner, R.; Rohnert, H.; Summerland, P.; & Stall, M. *Pattern-Oriented Software Architecture*. Chic ester, UK: Wiley, 1996.

[3] Banani Roy and T.C. Nicholas Graham;"Methods for evaluating software architecture: A survey" Technical Report no.

[4] Muhammad Ali Babar, Ian Gorton;"Comparison of Scenario-Based Architecture Evaluation Methods" National ICT Australia Ltd. And University of New South Wales, Australia.

[5] Mugurel T.Ionita, Dieter K.Hammer, Henk Obbink;"Scenario-Based Software Architecture Evaluation

Methods: An overview" university at Kingston Ontario, Canada; April 2008.

[6] Fadhel.S.AlAbdullah, Fahad H.Alshammari, Rami Alnaqeib, Hamid A.Jalab, A.A.Zaidan, B.B.Zaidan "Analytical Study on Internet Banking System ",June 2010.

[7] Hamdan.O.Alanazi , Rami Alnaqeib, Ali K.Hmood, M.A.Zaidan , Yahya Al-Nabhani " On the Module of Internet Banking System" ,May 2010.

[8] Rick Kazman, Gregory Abowd,Mike Webb " SAAM: A Method for Analyzing the Properties of Software Architectures " .

[9] Software Architecture By Zheng Qin Jiankuan Xing Xiang Zheng.

[10] Software Architecture Analysis Tool By Johan Muskens.

[11] Varinder Pabbi ," Software Architecture and its Various Tools" , 2012

[12] A. Meiappane, B. Chithra, Dr. V. Prasanna Venkatesan, "Identification and Implementation of Observer and State Pattern in Internet Banking Application"**IEEE** - ICRCC 2012, SKP Engineering College, India – 606 611. Pp- 305-307. December 21-22, 2012. IEEE Catalog No : CFP1265T - ART, ISBN No : 978-1-4673-2758-9.