Finite State Machine as a Designing Tool for Web Based Transactional Systems

Binayak Panda[1], Dr. Nilambar Sethi[2]

Asst. Prof, Dept. of CSE, GIET Gunupur, Odisha, India[1]

Assoc. Prof, Dept. of CSE, GIET Gunupur, Odisha, India[2]

Abstract: Web based systems are highly useful for extending business as well as information sharing. Many platforms and technologies available for development of Web based systems. For a successful Web based system the design plays a vital role where the business logic is to be realised. Many traditional design approaches available, among which choosing the appropriate one is a challenge. Many Web based systems are developed in a hit and trail manner to meet the business objective. But having a proper design helps in future enhancement and systematic understanding of the system. In this paper Finite Automaton has been chosen as a design approach for Web based transactional systems which are functionally different than Web based informative systems. Finite automaton is an oldest and efficient mechanism for modelling any mathematical problem. Software development or Web based development can be visualized as s mathematical problem hence finite automaton can be used as a design approach for developing Web based Systems. In this paper a comparison on key features has been carried out with other tradition design approaches to justify the Finite automaton is equally efficient for designing a software or Web based system.

Keywords: Finite Automaton a design approach for Web based System, Web based System development, Web based transactional System development, Web based Informative system, and Design of Web based system.

1. INTRODUCTION

Web based system development is a realization of Web engineering. Web engineering [6] is the establishment and use of sound scientific, engineering and management principles and disciplined and systematic approaches to the successful development, deployment and maintenance of high quality Web-based systems and applications.

Web Engineering as Software Engineering
Though Web engineering involves some programming and software development, and adopts some of the principles of the software engineering [7], Web-based system development is different from software development, and also Web engineering is different from software engineering.

1. Most Web-based systems, at least as of now, are document-oriented containing static or dynamic Web pages.
2. Web-based systems will continue to be focused on look and feel, favoring visual creativity and incorporation of multimedia (in varying degrees) in presentation and interface. More emphasis will be placed on visual creativity and presentation as regards to the front-end interface with which a user interacts.
3. Most Web-based systems will continue to be content-driven – often Web-based system development includes development of the content presented.
4. Multiplicity of user profiles – Most Web-based systems need to cater to users with diverse skills and capability, complicating human-computer interaction, user interface and information presentation.
5. The nature and characteristics of the medium of Web is not well understood as the software medium.
6. The Web exemplifies a greater bond between art and science than generally encountered in software development.
7. Most Web-based systems need to be developed within a short time, making it difficult to apply the same level of formal planning and testing as used in software development.
8. Also Web is different from software as related to the delivery medium.
9. Further, the type of individuals who build/develop Web-based systems are vastly varied in their background, skills, knowledge and system understanding, and as well as their perception of Web and quality Web-based system.

2. IMPACT OF WEBSITE ON ORGANIZATION

[2] Organizations recognize the importance of the Web as a conduit for acquiring knowledge about customers, competitors and partners. They provide Website that aims to facilitate Web users' acquisition of knowledge about their organizations' activities, archives, products and services (Fang and Holsapple, 2006). A number of studies found that Website or internet has a strong impact on organization's processes and performance. Avlonitis and Karayanni (2000) found that there is substantial positive effect of the use of the Internet on sales management activities, market-oriented product management activities, and sales performance and efficiency. A study done by Lichtenthal and Eliaz (2003) found that companies use the internet to alter existing industry structures and business processes to improve company information, redefine their information with clients, leverage global resources and pioneer new business models. Another study done by Tang and Huang (2008) found that there is a strongest impact on Website use and that the improvement of customer
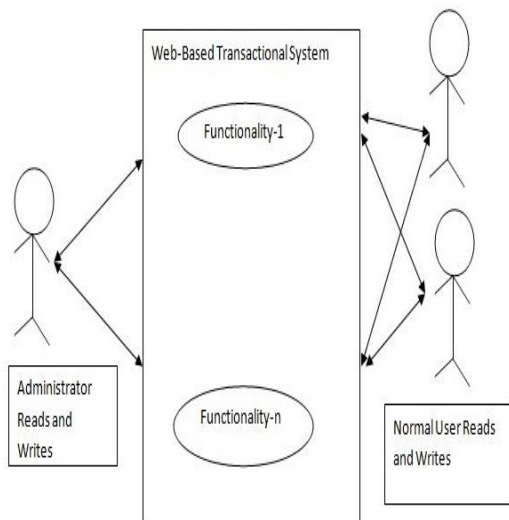
satisfaction can significantly increase organization performance. A study on website adoption and sales performance in Valais' hospitality industry by Scaglione et al. (2009) found that E-business has little effect on productivity in small enterprises, but positive effects on performance in large firms. The observed patterns suggested that Internet technologies might have a positive impact on hotel performance via improved marketing and distribution.

Classification Web-based system With Respect to use
[9] In the current era few Web-based systems play a major role in information sharing and few are used for extending business ignoring many barriers like geographical location, operational time etc. Considering the above stated criteria Web-based systems can be classified in to two categories as explained below.

Web-based transactional System-
- Transaction is a term usually means a sequence of information exchange and related work.
- For any organization having operational units at different geographic locations and different operational times information exchange is a major challenge. As a solution to this Web-based systems are widely accepted.
- Designing such Web-based systems which can be a replica of the manual operations of the organization is a challenge.
- Many designing tools or design approaches existing like Data Flow Based Design; Unified Modeling Language based design etc.
- Here Finite Automata one of the oldest Modeling tool has been used to propose a design approach for Web-based transactional systems.
- A Use-Case diagram [Figure-1] depicts the picture of overall design of Web-based Transactional System.
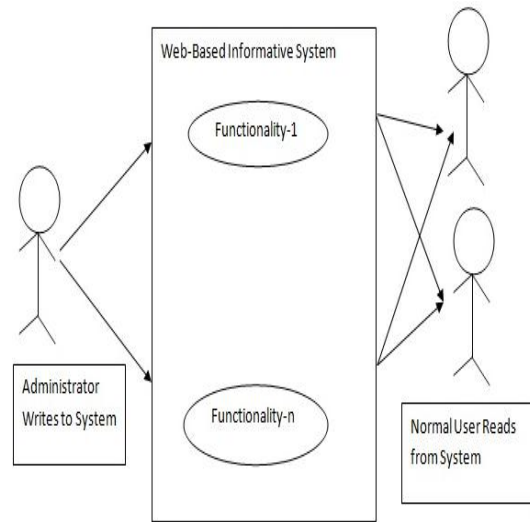

[Figure-1]

Web-base Informative System-
- These are Web-based systems only meant for displaying information.
- Traditional design approaches are well efficient for the design of such Web-based systems.

- A Use-Case diagram [Figure-2] depicts the picture of overall design of Web-based Informative System.

[Figure-2]

3. PROPOSED APPROACH FOR THE DESIGN OF WEB-BASED TRANSACTIONAL SYSTEMS

3.1 Argument about software as a Finite State Machine
[1, 8, 5, 3] Software in execution can be seen as Finite State Machine or Finite State Automata. [8, 4] Finite State Machines are very popular to formulate mathematical models expressing behaviour of complex system in terms of input, output and state. Formally, a Finite State Machine is defined as:
A tuple, A $(Q,\Sigma, q0, \delta, F, h, R)$ where:
Q: Finite set of states,
Σ : Alphabet of input symbols,
R: Output alphabet
q0: One start (initial) state where $q0 \in Q$,
F: Zero or more final (accepting) states where $F \subseteq Q$,
δ: A transition function $\Sigma \times Q \rightarrow Q$,
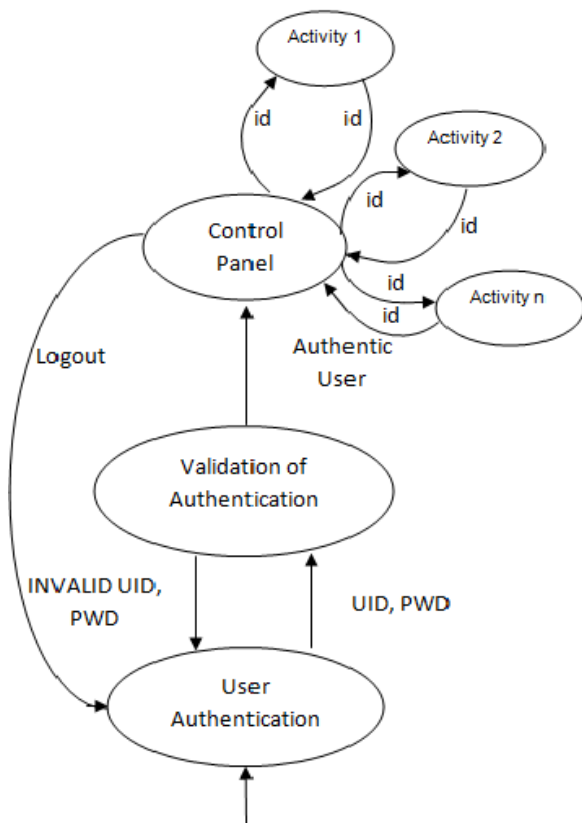h: Output function h: $Q \rightarrow R$

A major advantage of Finite State Machine approach is the study of system behaviour. A Finite State Machine based model can further allow easy identification of an error state caused due to factors like incorrect input, misrepresentation of values, incorrect bounds checking, memory overflow etc.

Various applications of FSM-Model are:
➤ Validation and Verification of web services choreographies.
➤ Testing of Interactive Systems
➤ Realistic Component Protocol Modelling
➤ And many more

3.2 A Generalized Finite Automaton Based Design For Web-based Transactional System
Finite automaton is one of oldest and efficient approach of expressing design for any mathematical model with input transition, output transition and state's of the system. An external view of Web based systems can be defined as a request form client browser (one state) to server

[Figure -3]

application (another state) where data are being sent from client browser to server application(label of transition) and data are received from server application to client browser (label of transition). It can be extended to represent all possible states the system may move around with respect to input transition and output transition. A generalized finite automaton based design for web-based systems is shown in [Figure-3].

3.3 A Comparison with Other Popular Design Approaches

In the [Table-1] 13 no. Of key features has been compared among three different design approaches.

Key Features	Methodology	Key Design Points	Prime Focus	Optimal Design	Realization of Design to Code	Reusability	Risk Factor on the approach	Adaptability Of Approach	Applicability	Reverse Engineering	Representation of Error Management	Input/output Ordering	Understandability for Stake Holders
Structured Design Approach (DFD)	SDLC models (Prototype, Incremental)	Inter Modular Dependency (Cohesion and Coupling)	Module/Functionality	No Optimality Test Possible	Difficult	YES	High	Widely used with maturity	For projects with Stable User Requirements	Difficult for Legacy Systems	NO	Not Present	Complex
Object Oriented Design Approach (UML)	USE CASE TOOLS	Inter Object Dependency (Inheritance)	OBJECT of System	No Optimality Test Possible	Difficult	YES	Low	Emerging with new concepts	For projects with Changing User Requirements	Difficult for Legacy Systems	NO	Not Present	Complex
Finite State Machine Based Design Approach (Finite Automaton)	FINITE STATE MACHINE	Inter State Dependency	States of System	Optimality Test Possible (Through Minimization of Finite Automata)	Relatively Simple (Transitions can be visualized from every state)	YES	Low	Widely used with maturity	For projects with stable as well as Changing User Requirements	Easy for Legacy Systems	YES (In Form of transition and Dead State)	Present (In the form of Label of transition)	Simple (Transitions are clearly visualized from finite automata)

[Table -1]

4. CONCLUSION

Design of a system plays a major role in realization to code and future enhancement. In the proposed design approach few important shortcomings of widely used design approaches like Structured Design Approach (DFD) and Object Oriented Design Approach (UML) has been sorted out. The proposed design approach has a proper input/output ordering specification in terms of input/output transition on states, Representation of Error Management is possible in the proposed approach in terms of dead state, Realization of Design to Code is easy, with the proposed design approach application of Reverse

Engineering on legacy products is more clear and understandability of the design by stake holders are more clear because of proper input output ordering which are the shortcomings and major challenges of other widely used approaches. As Finite automaton is one input and multi-output model, it can be extended to multi input and multi output Transition System with concepts of Mealy and Moore models for complex system design.

REFERENCES

[1] "Automata-Based Software Reliability Model: The Key to Reliable Software", International Journal of Software Engineering and Its Applications, Vol.7, No.6 (2013), pp.111-126

[2] Chai Lee Goi, "The impact of website development on organisation performance: Malaysia's perspective", African Journal of Business Management Vol. 6(7), 22 February, 2012, pp. 2435-2448.

[3] J. Li, F. Xie, T. Ball, V. Levin and C. McGarvey, "An Automaton, Int. Conf. on Fundamental Approaches to Software Engineering, Proc., (2010), pp. 248-262.

[4] "Modeling Software with Finite State Machines A Practical Approach", Auerbach Publications by Taylor & Francis Group, LLC © 2006

[5] K. Kandt, "Software Design Principles and Practices", Jet Propulsion Laboratory, NASA, http://trs-new.jpl.nasa.gov/dspace/bitstream/2014/10550/1/02-2593.pdf, (2003).

[6] San Murugesan, Yogesh Deshpande, Steve Hansen and Athula Ginige "Web Engineering: A New Discipline for Development of Web-based Systems",WebISM (Web-based Information Systems and Methodologies) Research Group, Campbelltown NSW 2560, Australia ,{s. murugesan, y. deshpande, s. hansen, a.ginige }@uws.edu.au (2001)

[7] Walt Scacchi , "Process Models in Software Engineering" , Institute for Software Research, University of California, Irvine February 2001 Revised Version, May 2001, October 2001 Final Version to appear in, J.J. Marciniak (ed.), Encyclopedia of Software Engineering, 2nd Edition, John Wiley and Sons, Inc, New York, December 2001.

[8] T.S. Chow, "Testing software design modeled by finite state machines", IEEE Transactions on Software Engineering SE, vol. 4, no. 3, (1978), pp. 178-187.

[9] Binayak Panda, Pradeepta Kumar Panigrahi, "A Model for Small Scale Website Development", International Journal of Computer Science and Technology, Vol.3, Issue.2, April-June 2012, pp 62-68

BIOGRAPHIES

Prof. Binayak Panda has received a bachelor's degree in Computer Science and Engineering from BPUT Odisha in the year 2005. In the year 2010 he has received a master of technology degree in Computer Science and Engineering from BPUT Odisha. Currently he is working as an Asst. Prof. In the Dept. of CSE at GIET Gunupur with 8 years of experience. He has 2 years of industry experiences in the field of Software testing and maintenance. His interested areas of research are Software Engineering and Real Time System. He is a life time member of ISTE.

Dr. Nilambar Sethi has received a master of technology degree from Utkal University, Odisha in Computer Science and awarded Ph.D from Berhampur University, Odisha in Computer Science. Currently he is working as Assoc. Prof. in Dept. of CSE at GIET Gunupur. He has 12 years of teaching experience. His interested areas of research are fuzzy logic and Automata Theory.