International Journal of Computer Science and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 3, Issue 1, Jan 2014, 85-92
© IASET

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

MODELLING WEB APPLICATION SYSTEMS WITH UML

ABDALLA ALAMEEN

Department of Computer Science and Information, College of Arts and Science, In Wadi Addawasir,
Salman Bin Abdulaziz University, KSA

ABSTRACT

As the increasing popularity and complexity of web applications and the emergence of new web application architecture, the design, development and maintenance of web applications are becoming complex and difficult. One way to manage the complexity is to model them. In this paper; we propose modeling web application architecture with UML, in which the design of web application is described from different viewpoints. It aims to facilitate the design and implementation of web applications and the reuse of previously gathered design experience.

KEYWORDS: Development Process Model, Modeling, Online Library Management, UML Web Applications, Use Case

INTRODUCTION

Modeling is a visual process used for constructing and documenting the design and structure of an application. It is a good idea to make at least some outline of web application system showing interdependencies and relationships between the components and subsystems during development.

Modeling can be introduced at any point in an existing project, as most modeling tools will read existing code, creating a visual model based on that code [5].

The old way of system modeling, known as the waterfall method, specifies that analysis, design, coding and deployment follow one another. Only when one is complete can the next one begins. If an analyst hands off analysis to a designer, who hands off a design to a developer, chances are that the three team members will rarely work together and share important insights. Usually the adherents of the waterfall method give coding a big amount of project time; so it takes a valuable time away from analysis and design [1].

The standard language used by most modeling tools on the market is called the Unified Modeling Language (UML) [2]. This language was developed to unify the many proprietary and incompatible modeling languages. Modeling with UML will become a necessity for reducing development time while ensuring that a program is well written the first time around.

DEVELOPMENT PROCESS MODEL

When building a web system, it is important to go through a series of predictable steps, a road map that helps you create a timely, high quality result and it is known as process model. Otherwise system ends up with lots of pits and falls. It is important because it provides stability, control, and organization to an activity that can, if left uncontrolled, become quite chaotic [1].

The UML contains number of graphical elements combined into diagrams. Because it is a language, the UML has rules for combining these elements [2].

The purpose of the diagrams is to present multiple views of web application systems, and this set of multiple views is called a model [4]. UML model describes what a system is supposed to do. It doesn't tell how to implement the system [4]. System development is a human activity. Without an easy-to-understand notation system, the development process has great potential for error. Consisting of a set of diagrams, the UML provides a standard that enables the system analyst to build a multifaceted blueprint that's comprehensible to clients, programmers and everyone involved in the development process. It's necessary to have all these diagrams because each one speaks to a different stakeholder in the system [4].

Varieties of models are available for web engineering and design. Each represents an attempt to bring order to an inherently chaotic activity. Each of the models has been characterized in a way that assists in the control and co-ordination of the real software project. The selection of process model is depending on the system type. The process model choice is based on the nature of the project and application, the methods and tools to be used and the controls and deliverables that are required [3].

UML can be used to model different kinds of systems: software systems, hardware systems and real-world organizations [2]. UML offers a set of diagrams in which to model web application systems that are:

- **Use Case Diagram:** For modeling the business processes.
- **Sequence Diagram:** For modeling message passing between objects.
- **Collaboration Diagram:** For modeling object interactions.
- **State Diagram:** For modeling the behavior of objects in the system.
- **Activity Diagram:** For modeling the behavior of Use Cases, objects or operations.
- **Class Diagram:** For modeling the static structure of classes in the system.
- **Object Diagram:** For modeling the static structure of objects in the system.
- **Component Diagram:** For modeling components.
- **Deployment Diagram:** For modeling distribution of the system.

WEB APPLICATION SYSTEM DESIGN

Web application system design is actually a multi-step process that focuses on four distinct attributes of a program that is data structure, system architecture, interface representation and procedural detail. The design process translates requirements into a representation of the system that can be assessed for quality before implementation.

Imagine a web base electronic library that can lends books, journals, CDs and videos to borrowers who are registered in the system. This application is called Online Library Management System (OLMS). The main task should be keeping track of all books, journals, CDs and videos with the status of each item (out of lone, in stock etc). In this application we are modeling and explaining the system with a sequence of UML diagrams. The first step for the system analysis is the specification of the requirements. In an object oriented and UML approach the requirements are identified with help of identifying of cases of use of the system [1]. This is done by UML use case diagrams. The main goal of this part is to identify the most characteristic use cases, and the actors (i.e. people or other types of “users” of the system). In Figure 1. UML use case diagram shows examples of how the system is intended to be used.

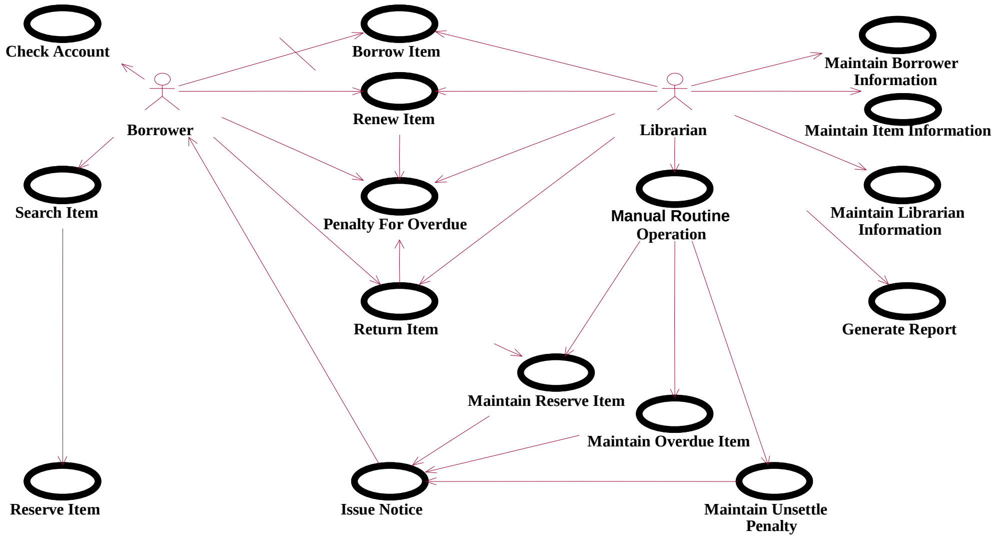

Figure 1: Use Cases of Online Library Management System

The next step in the analysis and design process is to identify the objects the system deal with. From the problem description and use case diagrams it is easy to identify the objects involved in the system: library item, item publisher, routine operations, report, reserve item, borrow and return item, borrower and librarian. We specify the objects by specifying the classes with their attributes and services (methods) they provide. In UML this is done by a class diagram. This diagram also includes specifications of relations between the objects as shown in Figure 2.

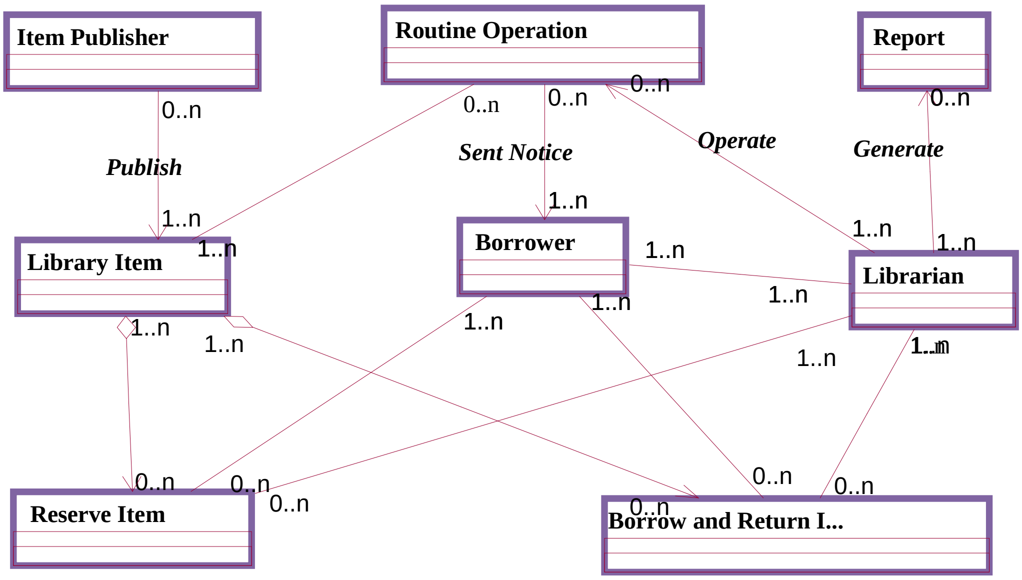

Figure 2: UML Class Diagram

As the system interacts with users and possibly with other systems, the objects that make up the system go through necessary changes to accommodate the interactions [8, 9]. If we're going to model systems, we must have a mechanism to model change. That mechanism in UML is State diagrams. Figure 3 describes the various states for Library items in state diagram.

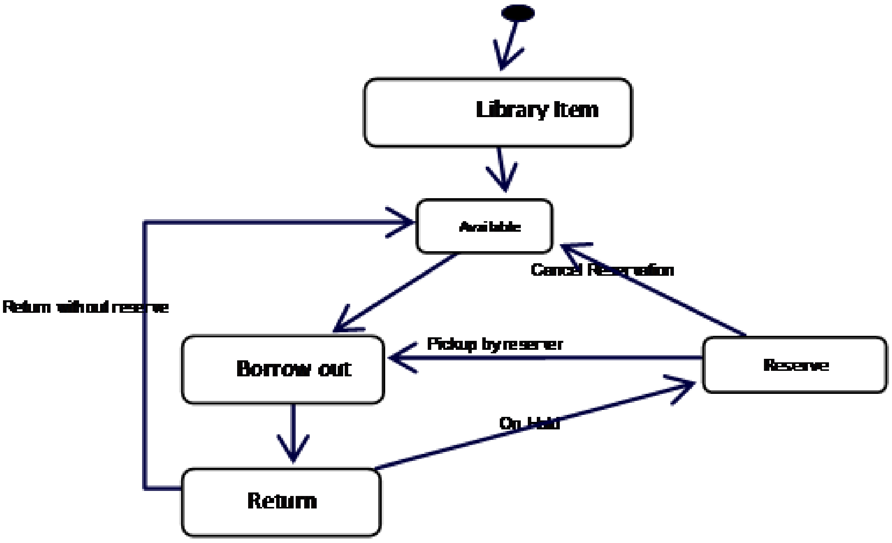

Figure 3: Library Item State Diagram

The UML class diagram shows the static characteristics (i.e. the structure) of the system [8]. The dynamic behaviour of the system can be described by activity diagrams and interaction diagrams. Activity diagrams are used to express the states of web application systems or internal inside classes and its transition from state to state triggered by a particular event. Activity diagrams are variations of finite-state machines which are a standard method used in software analysis, design and programming [9].

Figure 3 Shows UML activity diagram that describes the internal state of the class Reserve Item. It describes that Librarian can hold the items that are not available in this moment for the borrower as reserved items, but when items are returned, the system will auto send out an email to notice the borrower base on first reserve first inform queue. However if the borrower didn’t pickup the items in three working days, system will auto remove him/her from the reserve list.

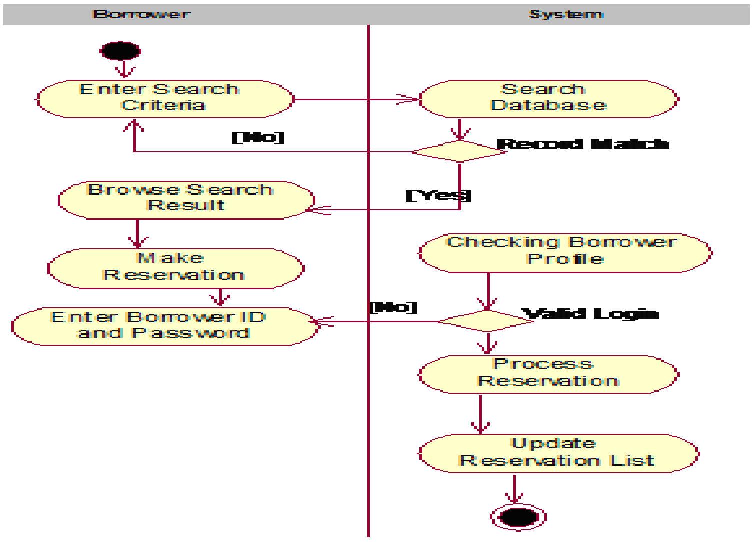

Figure 4: Reserve Item Activity Diagram

UML sequence diagrams are used to show interaction between different objects in a time sequence [8, 9]. The vertical line denotes time, the rectangles appearance of the objects and the arrows invocation of services of particular objects or interaction between the objects.

Figure 4 describes Item borrow sequence diagram, in which the Borrower comes over and wishes to borrow the item. The Librarian gets the borrower ID from borrower library card checking the quota. Then, the system checks the borrower quota and makes sure it is not yet up to the limit (the total item that can borrow out).The Librarian records the item borrowed out according to the item unique number and repeats this process until all items wanted to be borrowed out have key in borrow record. Finally, the Librarian prints out the current borrow list as per request.

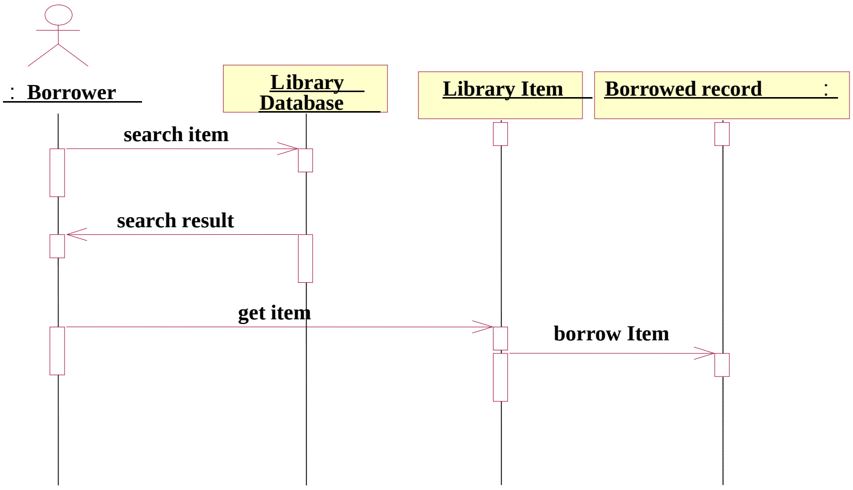

Figure 5: Item Borrow Sequence Diagram

A component is a code module. Component diagrams are physical analogs of class diagram. Deployment diagrams show the physical configurations of software and hardware [10]. The following diagram in Figure 5 shows the relationships between the components that make up online Library Management System.

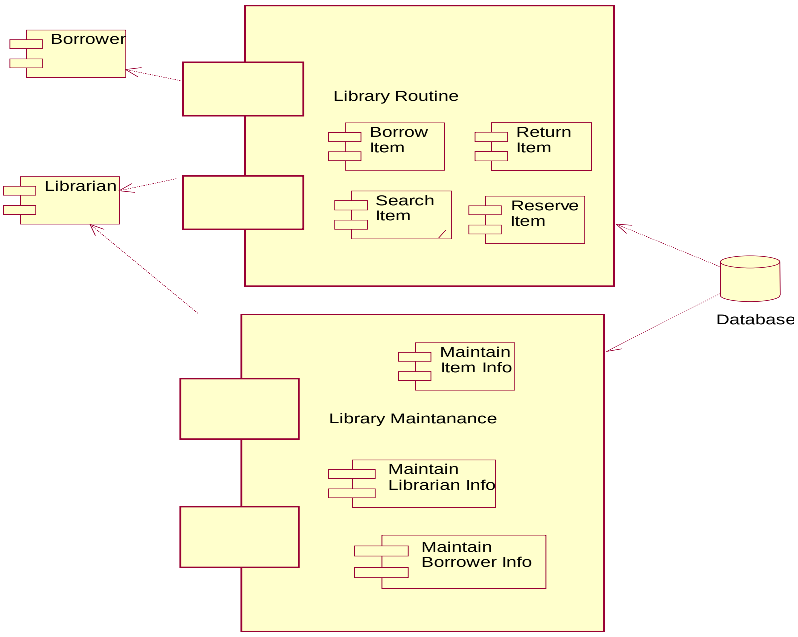

Figure 6: Component Diagram

Component diagrams are used to describe the static deployment view of a system [2]. These diagrams are mainly used by system engineers. Deployment diagrams consist of nodes and their relationships.

An efficient deployment diagram is an integral part of software application development. Figure 7 shows how web application system can be deployed.

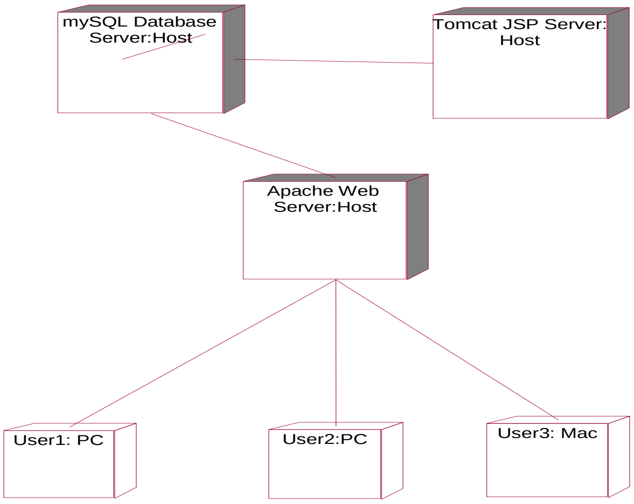

Figure 7: Library Management System Deployment Diagram

CONCLUSIONS AND FUTURE WORK

The ultimate objective of this study was to use the results and conclusions obtained from engineering web application system with UML .We used different UML diagrams to model some aspect of web application system from various viewpoints. It is aimed at facilitating the design and implementation of large, complex web applications

In fact modeling is the core of modern software development. Like every larger engineering task, software development needs a plan. Models serve us as a plan and allow achieving multiple goals [5]. It provides a structure for problem solving, experiment to explore multiple solutions for web application, furnishes abstractions to manage complexity, reduces time-to-market for business problem solutions, decreases development costs and manages the risk of mistakes.

This paper outlined modeling web application architecture with UML where the design of web application is described from different viewpoints. The models presented here require refining and the support of tools. Planned work is to test this notation in the field and to reshape it following the feedback of its target audience and developing the notation itself in a user centered manner.

REFERENCES

1. Whitten, N., *Managing Software Development Projects: Formula for Success*, 5nd ed., John Wiley & Sons, Inc, N.Y., US., 2008.
2. Rational Rose Web Modeler, http://www-03.ibm.com/software/products/en/rosemod, visited on 10/12/2013.
3. Conallen. *Building Web Applications with UML*. Addison-Wesley, 2nd ed., 2002
4. De Troyer, O., Leune, C.: "WSDM: *A User-Centered Design Method for Web Sites*", Computer Networks and ISDN systems, Proceedings of the 7th International World Wide Web Conference, pp. 85 - 94, Publ. Elsevier, Brisbane, Australia,1998.
5. Dennis, Alan, Barbara Haley Wixom, and David Tegarden , *Systems Analysis and Design with UML 2.0*: An Object-Oriented Approach. John Wiley & Sons, New York, 2005
6. Analysing Requirements and Defining Solution Architectures: MCSD Training Kit for exam 70-100, Microsoft Press, USA, 20*13*

7. Fournier, Roger. *Methodology for Client /Server and Web Application Development*. Prentice Hall, ISBN: 0-13-598426-2, 1999.

8. Booch, Grady. *Object-oriented analysis and design with applications*. Addison Wesley, ISBN: 0-8053-5340-2, 1994.

9. Larman, Craig. *Applying UML and patterns*. Prentice-Hall, ISBN: 0-13-748880-7, 1998.

10. Powell, Thomas A. *Web design: the complete reference*. Osborne/McGraw-Hill, 2000.

AUTHOR DETAILS

Abdalla Abdelrahim Al-Ameen received his BSc degree in computer science from International University of Africa, Sudan in 1997. He completed his MSc in information technology form AL Neelain University, Sudan. He received his PhD degree in Computer Science from Department of Computer Science and Information Technology at AL Neelain University, Sudan. Currently, he is working as Assistant Professor at Computer Science and information Department, College of Arts and Science in WadiAddawasir, Salman Bin Abdulaziz University, KSA. His fields of interest are in data security, web design, cryptography, E -government, E-learning and web application modeling.