Chapter - 1

INTRODUCTION
CHAPTER 1

Chapter - 1. Introduction

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Name of the Sub-Title</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Importance of Web Applications</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Importance of Design Patterns</td>
<td>4</td>
</tr>
<tr>
<td>1.3</td>
<td>Quality and performance parameters</td>
<td>5</td>
</tr>
<tr>
<td>1.4</td>
<td>Motivation</td>
<td>8</td>
</tr>
<tr>
<td>1.5</td>
<td>Contribution of the Thesis</td>
<td>11</td>
</tr>
<tr>
<td>1.6</td>
<td>Organization of the Thesis</td>
<td>12</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

A web application is a software that runs in a web browser. It is created by browser-supported programming language such as the combination of HTML (Hyper Text Markup Language), CSS (Cascading Style Sheets), and JavaScript. Web applications are computer programs allowing users to submit and retrieve data to/from a database over the Internet using web browser. When the information is generated dynamically by the web application through a web server, the data is presented to the user within their browser.

1.1 Importance of Web Applications

Web application is the cornerstone of making business online for most of the enterprises in 21st century. With such applications businesses are able to reach global audience and sell products and services online. This has eliminated many hardships of traditional marketing and sales. The online business has bestowed up on advantages to both consumers and service providers.

With online web applications, the customers can perform various activities like surfing products and services, purchasing, selling, transferring money and so on with the comfort of their home. As there is tremendous growth and innovation in mobile technologies,
it helped enterprises to have responsive web applications that are rendered in hand held devices as well.

This has added more comfort to consumers as people of all walks of life are using mobile devices or smart phones [4] for communication. This is the significance given to online applications in post-Internet era. Web applications are expected to have highest quality and performance. User experience that [3] a web application provides is an integral part of the application’s performance.

If customers of a commercial web site experience slow responsiveness it is unlikely that they prefer the web site irrespective of other excellent services it offers. Therefore, in the best interest of the provider of web application, consistent and rich user experience always is an inevitable. Rich user experience besides good usability is essential for the success of businesses going online.

There are many factors that influence usability, quality and performance of web applications. They include server side resources, bandwidth, response time, the scalable design of web applications and so on. According to Patil et al. [7] high Response Time should be the important characteristic of modern web applications. With respect to design and implementation of web applications, architecture and underlying design patterns can bring about consistency, quality, and performance benefits to web applications. In fact it ensures systematic approach in designing web applications.
The technological innovations over WWW (World Wide Web) such as Web 2.0 facilitate designing web applications with rich user experience. For instance asynchronous communication through AJAX (Asynchronous JavaScript and XML) will let the servers push content to web browsers without full page refreshes. Thus with AJAX rich internet applications can be built.

1.2 Importance of Design Patterns

Design pattern is a proven solution to a frequently occurring design problem. Design patterns play an important role in web application design. The last decade has witnessed an increasing usage of design patterns in web application design for improving quality and performance. Web application developers have been using creational, behavioral and structural design patterns since they are based on design principles.

Recurring object oriented design problems are addressed by the design patterns. A design pattern, when used appropriately, is capable of improving quality and performance of a web application. There might be many reasons for inconsistent responsiveness of a web site such as high load, components failure and so on. However, from user perspective, these are not important as he looks for good service. This needs some sort of consistent and proven solution. Design patterns are such blueprints or proven solutions or industry best practices which can be used to improve the performance of web applications.
Nevertheless we believe that consistent performance of web application can be leveraged by appropriate use of design patterns[1]. However, there is a fact to be kept in mind that resources associated with web server can influence the performance of a web application. Assuming optimal resource availability in web server, design patterns can improve performance of web applications.

There are many good reasons to use design patterns as they can get rid of reinventing the wheel by promoting reuse. Moreover they are proven, and expressive.

1.3 Quality and Performance Parameters

The different quality and performance parameters of web applications are response time, throughput, fault tolerance, scalability and availability. Access time [10] is the delay or Response Time between the time when request is made and the time at which response is rendered. That is response time is a measure that tell how long user waits to get response to a query. Response Time is also called Latency.

Throughput is used to measure [5] the workload of a web application. In fact it is best referred to as quantification of requests or responses in relation to time. In other words the amount of work accomplished in a certain amount of time period.

A deviation from the normal expected execution is known as Fault. Fault tolerance is the capability of a system to recover from a
fault or error without exhibiting failure. Web applications are widely used by businesses to reach global audience. E-commerce and other commercial applications that leverage business performance need to be given highest priority since their performance leads success.

Having said this it is essential to make such applications fault tolerant. Faults might occur for many reasons such as software, hardware, network and human errors. Building a web application without fail is a challenging task. Apart from measures taken at server level with respect to fault tolerant behavior of web server the application level fault tolerant structure or strategies play an important role to achieve fail-safe standard.

Fault tolerance is also known as graceful degradation [6] which helps the application to achieve operational continuity. It can recover fast from unknown or unexpected faults. Rather than failing, the fault tolerant web application continues working, most probably, in less than ideal fashion until the fault is overcome.

Scalability and availability are two highly desirable attributes pertaining to reliability of a web application that renders state-of-the-art services to online users. Scalability is the ability to grow, the ability to serve increased number of requests or clients. Building a scalable application with round the clock availability is a challenging problem in the light of ever increasing population of potential users.
Dramatic increase in users to web application causes bursts of requests that put the application to acid test. On the other hand web application availability represents the degree of operational continuity. The measurement for availability is the actual quantification of failure recovery time. High availability and unlimited scalability are the two indispensable quality attributes of a web application in the real world.

These features bestow rich user experience as far as operational continuity and ability to handle growing workload are concerned. By taking server side measures it is possible to achieve these two desirable features. However, there is possibility to have architectural pattern along with underlying design patterns to promote these quality features of web application.

According to Sanderson [11] a web application is scalable when each user gets same quality irrespective of the number of users concurrently accessing the application. Poor usability [12] and poor scalability results in web applications when designers are not aware of web design patterns [13]. Web design patterns are useful for developing web framework. Web Framework has been used in real-world enterprise Web applications. Web frameworks were also written without considering scalability usability and simplicity [14]. Enterprise web applications that generate content dynamically and data grids throw scalability challenges [15].
1.4 Motivation

The main reasons for improving access time, throughput, fault tolerance, scalability and availability are to improve quality and performance of web applications. Quality and Performance problems always occur with improper designing of the web applications.

Let us consider two web based applications namely Hospital Management System (HMS) and Library Management System (LMS). These applications are running in the real world. They are applications without using any architectural pattern. In this research we used these applications for our empirical study to evaluate the proposed framework incrementally as new design patterns are included into the main architectural pattern.

HMS (Hospital Management System) and LMS (Library Management System) are web applications with varying degree of tuples in the datasets. They are used as it is and with our framework to evaluate the quality and performance attributes such as response time, throughput, fault tolerance, scalability and availability. When people using Online Hospital Management System it is observed that the response time to download all prescriptions will take more than expected.

Due to poor response time people may not get all prescriptions. Similarly if it is Online Library Management System downloading all books will take more time. So in the above two systems response time
will be more to download different types of data. Sometimes when download is started it may fail because of number of faults in the system which reduces quality.

If the user wants the system that system will not be available because the system download is not completed. These are the two examples which explain the importance of quality and performance of web applications. So there is a great need to develop the web application with good quality and performance. To do this first we need to develop good web application design. So we first develop web application architecture using architectural patterns.

The architectural patterns that have been found in the literature are useful in certain areas for improving quality and performance of web applications. These patterns include MVC (Model View Controller), JEMSF (Job Evaluation model specific framework), OOHDM (Object Oriented Hypermedia Design Model), PAC (Presentation Abstraction Control), SMT (Simultaneous Multi Threading), PCMEF (Presentation Control Mediator Entity Foundation), ISPWAD (Internet Security Provider Web based Application Design) and so on.

The interesting thread in all these frameworks is that each framework is focused on certain aspects of design of web applications. For instance, MVC has division of labor with many advantages. However, it does give freedom in all the layers for implementation. So Model View Controller (MVC)[18] is one of the well-known architecture
which renders many advantages to web applications including maintainability[16], availability, scalability [17] etc. PAC throws light into presentation part only. SMT is best used for optimum resource utilization.

JEMSF uses design patterns such as DTO (Data Transfer Object) and DAO (Database Access Object) besides connection pooling. It has left scope for many other design patterns that cater to scalability, availability and fault tolerant web application development. OOHDM mainly focuses on hypermedia utilization in object oriented fashion for making navigations dynamic and interactive. It is the mix of conceptual model, navigational classes, and interface objects while ISPWAD throws light into security aspects of web applications.

In this context, the web applications in terms of response time, throughput, fault tolerance, scalability and availability, have to be given paramount importance. The existing frameworks catered to these features partially. There was no architectural pattern with underlying design patterns that encapsulates all these parameters pertaining to quality and performance of web applications. This is the motivation behind taking up this research work.

There are different drawbacks of existing system. They are

- The existing architectural patterns are either focusing on particular attribute of quality or navigational behavior of web applications.
• There is no comprehensive architectural pattern with underlying design patterns that could cater to all desired quality and performance attributes of web applications.

• Fault tolerance, scalability and availability features were essential for a successful web application which could not be found in a single architectural pattern.

1.5 Contribution of the Thesis

The contribution of the Thesis is given below.

1. We proposed an architectural pattern by name eXtensible Web Application Development Framework (XWADF) with underlying design patterns in all the layers for improving response time and throughput. The proposed architectural pattern is extensible and we intend to add more design patterns to the architecture in future.

2. Our architectural pattern was evaluated through a case study web application that exhibits the subtle difference in performance between the web application that uses our architecture and the web application that does not use it.

3. Response Time and Throughput both are used to find the performance of web applications. The Response Time is further divided into two elements known as fetch Response Time and render time. Fetch Response Time is the time to load web page
into browser while the render time is the time required to receive elements references by the loaded web page [19].

4. Our architectural pattern XWADF has been enhanced by incorporating the design patterns proposed for promoting fault tolerant web applications into the architectural pattern.

5. Fault Tolerant metrics were built to test the case study applications and provide fault tolerant improvements achieved through the use of our enhanced architectural pattern XWADF.

6. Our architectural pattern XWADF has been enhanced further by incorporating the design patterns identified for promoting scalability and availability of web applications into the architectural pattern.

7. Scalability and Availability metrics were built to test the case study applications and provide reliability improvements achieved through the use of our enhanced architectural pattern XWADF.

8. Different Refactoring Algorithms has been provided which helps developers to upgrade their enterprise web applications in conformity with the proposed architectural pattern for performance and quality gains in terms of throughput, response time, fault tolerance, scalability and availability.

1.6 Organization of the Thesis

This thesis is aimed at providing an architectural solution that improves quality and performance of web applications. The design,
implementation and evaluation of architectural pattern has been made. The remainder of the thesis report is structured into chapters that provide incremental overview of our contributions to fulfill the aim.

In Chapter 2, a detailed review of literature is carried out on various architectural patterns that contributed to improving quality and performance of web applications. This chapter throws light into the insights of design and implementation of fault tolerant, scalable, and available web applications. Its coverage includes quality and performance attributes, architectural patterns, design patterns vs. fault tolerant web applications, design patterns vs. scalability and availability of web applications and the summary of findings besides the research scope.

In Chapter 3, an architectural pattern based on MVC is proposed with underlying design patterns. The architectural pattern is named as eXtensible Web Application Development Framework (XWADF) which is extensible in nature and intended to get enhanced incrementally. The architectural pattern with underlying design patterns is empirically tested with two case study applications namely HMS (Hospital Management System) and LMS (Library Management System). Response Time and throughput metrics are used to measure the performance of web applications built using our framework. In addition to this “Refactoring Algorithm” which helps developers to
upgrade their enterprise web applications in conformity with the proposed architectural pattern for performance gains in terms of throughput and response time.

In Chapter 4, details of incremental enhancements made to XWADF architectural pattern are provided. The details include strategies or methods that can withstand faults in web applications at application level. These strategies are embodied into some design patterns to make web applications fault tolerant. Afterwards design patterns are incorporated into the architectural pattern to make our architectural pattern useful tool to build highly robust web applications that exhibit desirable quality attributes such as fault tolerance. In addition to this “Fault Tolerant Refactoring Algorithm” which helps developers to upgrade their enterprise web applications in conformity with the proposed enhanced architectural pattern for quality gains in terms of fault tolerance.

In Chapter 5, our architectural pattern XWADF is enhanced further by incorporating the design patterns identified for promoting scalability and availability of web applications into the architectural pattern. It describes two case study applications that demonstrate the usefulness of the enhanced architecture as far as scalability and availability are concerned. Then reliability metrics are used to test the case study applications and provide reliability improvements achieved through the use of our architectural pattern XWADF. In addition to
this “Scalability and Availability Refactoring Algorithm” which helps developers to upgrade their enterprise web applications in conformity with the proposed enhanced architectural pattern for quality gains in terms of scalability and availability.

In Chapter 6, the results are drawn. Two case studies Hospital Management System (HMS) and Library Management System (LMS) are used. The database is used with different entities. For HMS the different entities (records) are 50000, 75000, 100000. Similarly for LMS the different entities are 10000, 25000, 50000. All these results are compared and discussed by without using design patterns and with using the design patterns.

In Chapter 7, conclusions are drawn. These conclusions provide research outcomes with respect to the aims of the study. It also provides directions for possible future enhancements to be made.