Web Session Classes: Performance Metrics For N-Tier's Business Logic Issues

Ashok Kumar#
#Banasthali University,Banasthali(Raj.)-India

Dr. Saurabh Mukherjee$
Banasthali University,Banasthali(Raj.)-India

Yogesh Kumar Gupta@
Banasthali University,Banasthali(Raj.)-India

Abstract

Information is a vital role playing versatile thing from availability at church level to web through trends of books. WWW is now the exposed and up-to-date huge repository of information available to everyone, everywhere and every time. Nowadays web and web applications play an important role in every aspect that limelight the performance issues. The most common consideration is performance, because these systems must provide services with low response time, high availability, and certain throughput level. Whenever we think to address these performance issues we find tools like Yahoo!YSlow, GTmetrix, WebPageTest ,Google Page Speed etc that consider these issues relevant to application layer only. This is a hard truth that performance issues for an n-tier web application in application layer include 80%-90% of issues. In this work we addressed remaining 10%-20% performance issues related with other two layers of architecture. With the help of performance models, the performance metrics having dimensions like average response time, average download time, average processing time, session per second, page per second can be evaluated. The goal of our work is to consider these dimensions and the n-tier utilization of web applications while developing square side performance of n-tier web application. In this work we experiment a web application with concurrent user sessions having database reader classes and web service in order to validate the n-tier models in ASP.NET environment.

This paper is an experimental strives to develop and implement a framework with n-tier architecture to address performance issues tied with business and database layer. This work is an implementation experience for use of concurrent sessions with web service and reader classes to provide a keen analysis on dimensions of performance metrics evaluated.

Keywords

Performance Metrics, Web Service, Concurrent Sessions, Performance Dimensions and Load Volume.

1. Introduction

WWW is immense to obtain information and moreover information voyaged on the web is available in websites with n-tier architecture model. In the past few years new platforms and frameworks for programming environments have been marketed to leap the web development. These new models and techniques of programming are escalated the software development with exponential extent. In contrast to the developing issues of these applications, performance is the key issue. Owing to the heavy load of traffic on the network systems, these are compromising with expectation constraint that they must provide high-availability services with low response time [1,6]. This is the scenario of these systems directing performance as one of the most important factors.

Owing to the reason that performance issues directing navigation of information from these applications is the striking one to sail the web for several purposes. Performance optimization of a web application is a raptorial field to address a state of fast growing rate of amount of information on the web. At the ground level, performance issues strongly tied with application layer does not count all of these tied with overall performance of the application.

The aim of this paper is to raffle a framework, which will elevate web application's dexterity while cutting issues of all sides to surmount the way the Internet can be used to snag more and more information and services with more performance [2, 3, 4].

The first part of the paper covers related work that has been done mostly in the field of performance issues in web application development in over ongoing research project for deploying performance efficient applications. The second part defines architecture and functioning of developed framework environment for experiment. The third part provides an overview and critical analysis of experimental framework like experimental results, pseudo code, data structure etc.

2. Related work

These performance metrics depend on many factors. Several papers and research works outcome with various

factors that affect the performance of a web-based information system. In these works several statistical methods, hypothesis tests are used in order to retrieve factors influencing the performance [2,5]. However, the performance- related dimensions emerge very often only at the end of the software project but with the help of properly designed performance models, the performance metrics of a system can be determined at the earlier stages of the development process [1,3]. In nutshell, performance of a website is evaluated by considering various traffic and tier specific parameters. These are further also categorized as technical and non-technical or behavioral parameters [6,13].

Today one of the most prominent technologies of web -based information systems is Microsoft .NET. Our primary goal is to calculate and analyze the different dimensions of performance metrics of ASP.NET web applications based on n-tier architecture models handling multiple session classes, because these dimensions are the only performance metrics to which the users are directly exposed (Fig.1). Our final goal is to stick these dimensions to the utilization of the tiers.
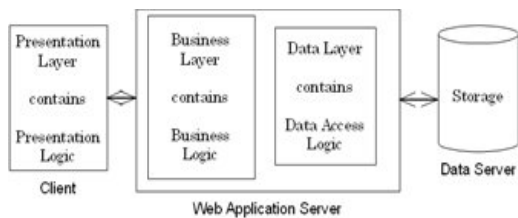

Fig. 1 Architecture of n-tier web application

Presentation layer: collection of classes and/or files that defines user interface or says application interface.
Business layer: collection of classes and/or files that defines functionality or says logic of application.
Data layer: collection of classes and/or files that defines functionality to access database or says database logic of application.
Data Server: A DBMS that is responsible to manage database or says backend of application.

3. Contributions

We have implemented a three-tier ASP.NET test web application having two different profiles profile4 (Fig. 2) and profile5 (Fig. 3).
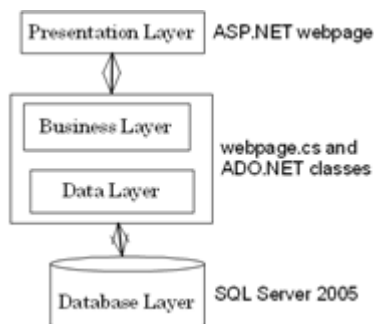

Fig.2 Architecture of profile4.

Profile5 is having web service module (database classes) compared to profile4 in our application to meet the needs of the measurement process of dimensions tied with business logic and data access layer.
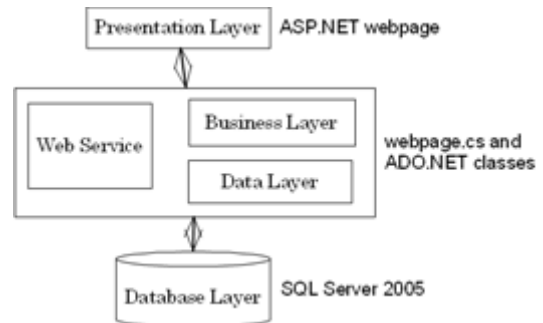

Fig.3 Architecture of profile5.

Thereafter, we have demonstrated and validated the models in the ASP.NET environment. Firstly, we have defined load volume of five users to both the profiles for seven page requests in each session and estimated the input values of the model parameters like session wait-time, test case volume, maximum time etc for test case to be performed in WAPT (Web Application Performance Testing).

Finally, we have tested a web application with two different profile sets running under concurrent user sessions, comparing the observed and predicted values in order to validate the models in the ASP.NET environment.

4. Performance

An estimated and approximate performance analysis can be done to compare the implemented strategies with each other. With the increase in availability of web pages on the Internet, the major problem faced by the deployed system is difficulty in information retrieval and this degradation factor is highly passionate with performance metrics of applications on web. With further increase in the size of the Internet, the problem grows exponentially.

In terms of performance parameters like bandwidth, CPU, memory, successful hit, successful session, response time, sending speed, receiving speed, processing time and downloading; profile5 in developed framework holds an edge above the conventional strategies of addressing performance dimensions associated with two tiers of n-tier architecture.

4.1 Experimental Screenshots

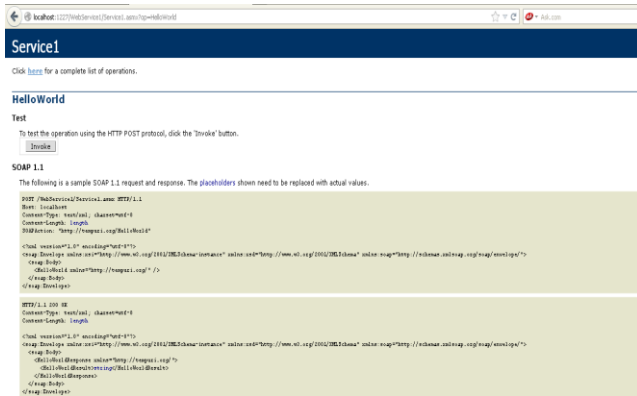A series of user interfaces of developed framework (Fig. 4, 5, 6 and 7)
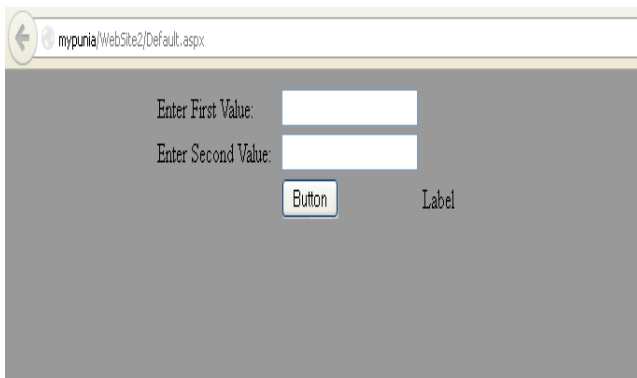
Fig. 4 Screen shots of web service.
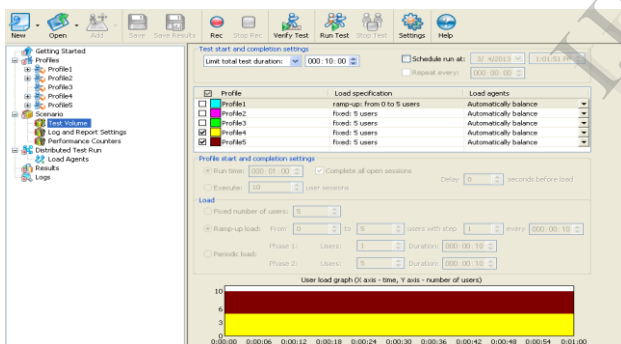

Fig. 5 Screen shots of website.

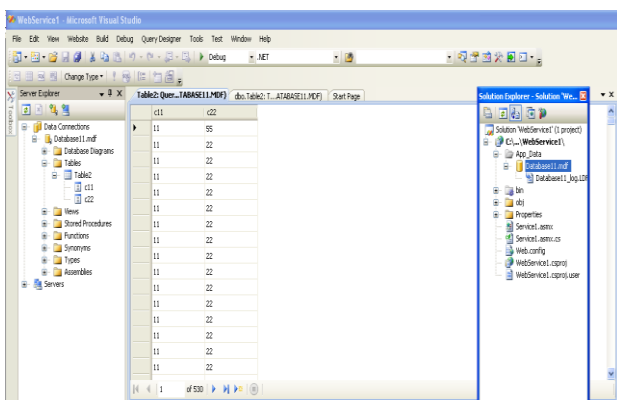
Fig.6 Screen shots of load testing.

Fig.7 Screen shots of back end.

4.2 Analysis

This optimized framework is running on an acer machine, a workstation with 685MHz processor, 12 GB of RAM,840 GB of local disk, 100 Mbit/sec Speed Internet, Windows Server 2003,IIS 7.5,Asp.Net run time framework 4.0,SQL Server 2008 and WAPT Pro 3.0.

In this paper, experimental statistics are presented of five concurrent sessions with seven page request for each session to compare with both the profiles; about these requests issued are published in literature. The profile5 is reported to have issued 15 successful sessions requests over time of 2 minutes with speed of 554 Kbits/sec and profile4 is reported to have issued 03 successful sessions requests over time of 2 minutes with speed of 180 Kbits/sec. Performance of any information retrieval system can be analyzed using parameters like coverage, user perception that are presented below:

4.2.1 Performance of profile4 and profile5:

Successful pages per second

Profile	0:00:00-0:00:06	0:00:06-0:00:12	0:00:12-0:00:18	0:00:18-0:00:24	0:00:24-0:00:30	0:00:30-0:00:36	0:00:36-0:00:42	0:00:42-0:00:48	0:00:48-0:00:54	0:00:54-0:01:00	Total
Profile4	1.67	0.17	0.50	1.17	0.50	0.17	0	0.17	1.50	0.33	0.62
Profile5	2.17	2	2.50	1.50	2.17	1.17	2.33	1.33	2.67	1.83	1.97
Total	3.83	2.17	3	2.67	2.67	1.33	2.33	1.50	4.17	2.17	2.58

Successful hits per second

Profile	0:00:00-0:00:06	0:00:06-0:00:12	0:00:12-0:00:18	0:00:18-0:00:24	0:00:24-0:00:30	0:00:30-0:00:36	0:00:36-0:00:42	0:00:42-0:00:48	0:00:48-0:00:54	0:00:54-0:01:00	Total
Profile4	1.67	0.17	0.50	1.17	0.50	0.17	0	0.17	1.50	0.33	0.62
Profile5	2.17	2	2.50	1.50	2.17	1.17	2.33	1.33	2.67	1.83	1.97
Total	3.83	2.17	3	2.67	2.67	1.33	2.33	1.50	4.17	2.17	2.58

Load Agent utilization, %

Name	Utilization	0:00:00-0:00:06	0:00:06-0:00:12	0:00:12-0:00:18	0:00:18-0:00:24	0:00:24-0:00:30	0:00:30-0:00:36	0:00:36-0:00:42	0:00:42-0:00:48	0:00:48-0:00:54	0:00:54-0:01:00	Total
localhost	CPU	3	2	3	2	2	1	1	2	2	2	2
	Memory Mb (%)	76(3)	77(3)	77(3)	77(3)	77(3)	77(3)	77(3)	77(3)	77(3)	77(3)	76(3)
	Network	0	0	0	0	0	0	0	0	0	0	0

4.2.2 Summary of profile4 and profile5:

Summary

Profile	Successful sessions	Failed sessions	Successful pages	Failed pages	Successful hits	Failed hits	Total KBytes sent	Total KBytes received	Avg Response time, sec (with page elements)
Profile4	3	0	37	0	37	0	32.2	48.1	0.02(0.02)
Profile5	15	0	118	0	118	0	104	165	0.02(0.02)

Number of active users

| Profile	0:00:00-0:00:06	0:00:06-0:00:12	0:00:12-0:00:18	0:00:18-0:00:24	0:00:24-0:00:30	0:00:30-0:00:36	0:00:36-0:00:42	0:00:42-0:00:48	0:00:48-0:00:54	0:00:54-0:01:00
Profile4	5	5	5	5	5	5	5	5	5	5
Profile5	5	5	5	5	5	5	5	5	5	5
Total	10	10	10	10	10	10	10	10	10	10

Successful sessions (Failed sessions)

| Profile	0:00:00-0:00:06	0:00:06-0:00:12	0:00:12-0:00:18	0:00:18-0:00:24	0:00:24-0:00:30	0:00:30-0:00:36	0:00:36-0:00:42	0:00:42-0:00:48	0:00:48-0:00:54	0:00:54-0:01:00	Total
Profile4	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	3(0)	0(0)	3(0)
Profile5	0(0)	0(0)	4(0)	1(0)	1(0)	1(0)	3(0)	0(0)	4(0)	1(0)	15(0)
Total	0(0)	0(0)	4(0)	1(0)	1(0)	1(0)	3(0)	0(0)	7(0)	1(0)	18(0)

Successful pages (Failed pages)

Profile	0:00:00-0:00:06	0:00:06-0:00:12	0:00:12-0:00:18	0:00:18-0:00:24	0:00:24-0:00:30	0:00:30-0:00:36	0:00:36-0:00:42	0:00:42-0:00:48	0:00:48-0:00:54	0:00:54-0:01:00	Total
Profile4	10(0)	1(0)	3(0)	7(0)	3(0)	1(0)	0(0)	1(0)	9(0)	2(0)	37(0)
Profile5	13(0)	12(0)	15(0)	9(0)	13(0)	7(0)	14(0)	8(0)	16(0)	11(0)	118(0)
Total	23(0)	13(0)	18(0)	16(0)	16(0)	8(0)	14(0)	9(0)	25(0)	13(0)	155(0)

4.2.3 Performance of profile4
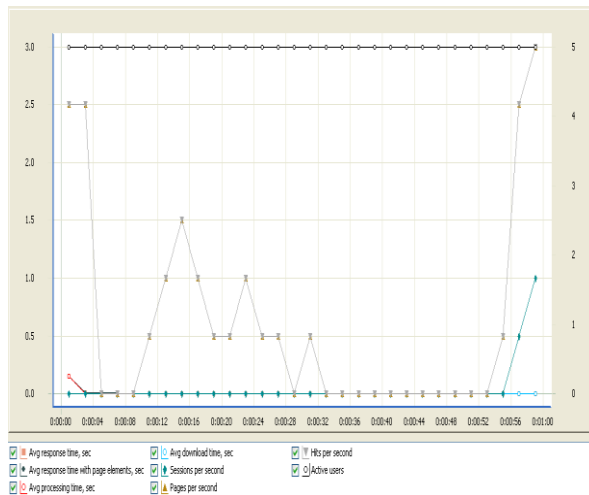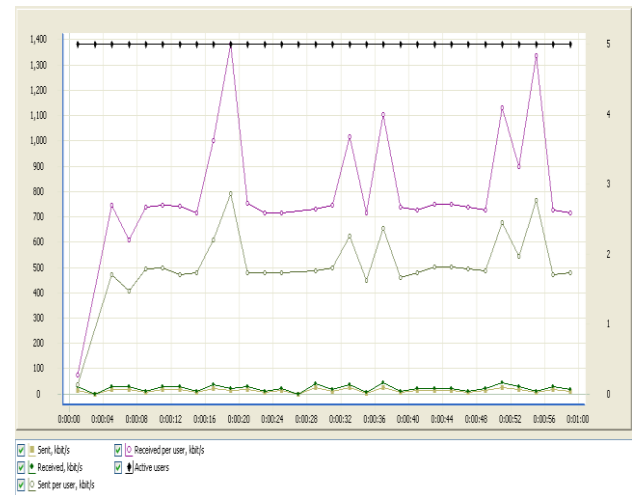
4.2.4 Bandwidth of profile4

4.2.5 Performance of Profile5:
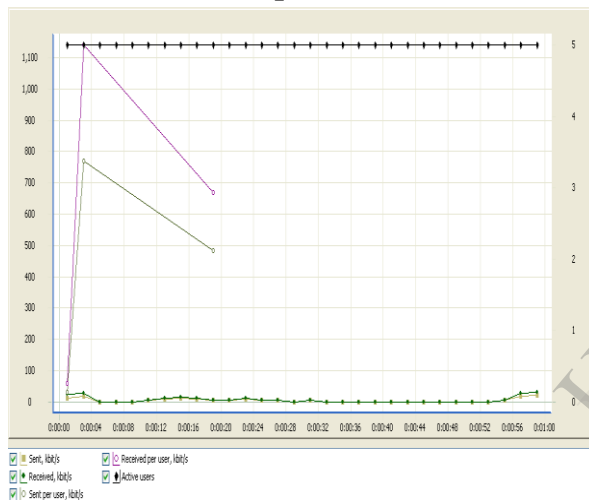
4.2.6 Bandwidth of Profile5:
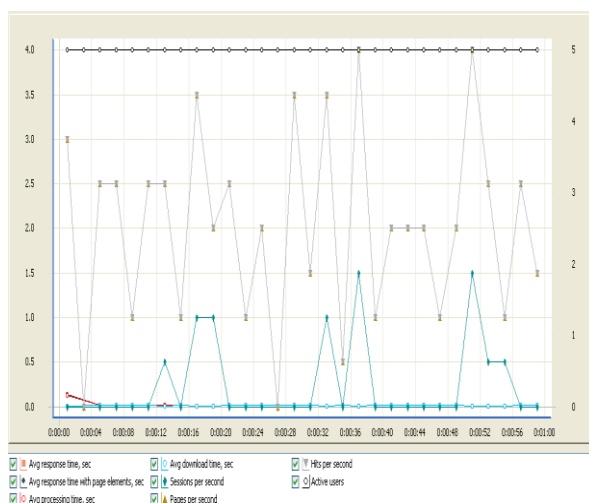
5. Conclusion

We have demonstrated a general model for the two profile environments with specific extension (profile5) is expected, which is more accurate than the baseline model handling multiple session classes only to validated n-tier architecture models handling web service and multiple session classes in ASP.NET environment.

This is a part of ongoing research work, to utilize advance technologies (like RDF) and model (MVC) in web programming. Owing to the lengthy size of coding work, this is not possible to present coding or technical details of all the modules of developed framework. But work is incomplete without functioning details of the basic modules i.e. application interface, business logic and ADO.NET module.

5.1 Coding of application interface module(profile4):

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="Default2.aspx.cs" Inherits="Default2" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML
1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<script runat ="server" >
  protected void Button1_Click(object sender,
EventArgs e)
  {
    SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionString
s["cons"].ConnectionString);
    con.Open();
    SqlCommand cmd = new SqlCommand("Insert into
table2 values('" + 11 + "','" + 22 + "')", con);
    cmd.ExecuteNonQuery();
    con.Close();
    Label1.Text = "submitted successfully";
    TextBox1.Text = "";
    TextBox2.Text = "";
```

```
      }
</script>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title>Untitled Page</title>
</head>
<body>
   <form id="form1" runat="server">
   <div>

      <asp:TextBox ID="TextBox1"
runat="server"></asp:TextBox>
      <br />
      <asp:TextBox ID="TextBox2"
runat="server"></asp:TextBox>
      <br />
      <asp:Button ID="Button1" runat="server"
onclick="Button1_Click" Text="Button" />
      <br />
      <asp:Label ID="Label1" runat="server"
Text="Label"></asp:Label>

   </div>
   </form>
</body>
</html>
```

5.2 Coding of application interface module(profile5):

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="Default.aspx.cs" Inherits="_Default" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML
1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<script runat ="server" >
   protected void Button1_Click(object sender,
EventArgs e)
   {
      ws123.Service1 ob = new ws123.Service1();
      Label1.Text = ob.HelloWorld();
      TextBox1.Text = "";
      TextBox2.Text = "";
   }
   </script>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title>Untitled Page</title>
</head>
<body>
   <form id="form1" runat="server">
   <div>

      <br />
      <asp:TextBox ID="TextBox1"
runat="server"></asp:TextBox>
      <br />
      <asp:TextBox ID="TextBox2"
runat="server"></asp:TextBox>
      <br />
```

```
      <br />
      <asp:Label ID="Label1" runat="server"
Text="Label"></asp:Label>
      <br />
      <br />
      <br />
      <br />
      <br />
      <asp:Button ID="Button1" runat="server"
onclick="Button1_Click" Text="Button" />

   </div>
   </form>
</body>
</html>
```

5.3 Coding of business logic module(profile4):

```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
using System.Data.SqlClient;

public partial class Default2 : System.Web.UI.Page
{

SqlConnection("server=.\\SQLEXPRESS;AttachDbFilen
ame=|DataDirectory|\\Database1.mdf;Integrated
Security=True;User
Instance=True;trusted_connection=yes");
   SqlConnection con = new
SqlConnection(ConfigurationManager.ConnectionString
s["cons"].ConnectionString);

   protected void Page_Load(object sender, EventArgs
e)
   {
   }
   protected void Button1_Click(object sender,
EventArgs e)
   {
      con.Open();
      SqlCommand cmd = new SqlCommand("Insert into
table2 values('" + 11 + "','" + 22 + "')", con);
      cmd.ExecuteNonQuery();
      con.Close();
      Label1.Text = "submitted successfully";
      TextBox1.Text = "";
      TextBox2.Text = "";

   }
}
```

5.4 Coding of business logic module(profile5):

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Xml.Linq;
using System.Data.SqlClient;
namespace WebService1
{
    /// <summary>
    /// Summary description for Service1
    /// </summary>
    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo =
WsiProfiles.BasicProfile1_1)]
    [ToolboxItem(false)]
    // To allow this Web Service to be called from script,
using ASP.NET AJAX, uncomment the following line.
    // [System.Web.Script.Services.ScriptService]
    public class Service1 :
System.Web.Services.WebService
    {

        [WebMethod]
        public string HelloWorld()
        {
            SqlConnection con = new
SqlConnection("server=.\\SQLEXPRESS;AttachDbFilen
ame=|DataDirectory|\\Database11.mdf;Integrated
Security=True;User
Instance=True;trusted_connection=yes");

            con.Open();
            SqlCommand cmd = new SqlCommand("Insert
into table2 values('" + 11 + "','" + 22 + "')", con);
            cmd.ExecuteNonQuery();
            con.Close();
            return "submitted successfully";
        }
    }
}
```

6. References

[1].Heydon, A. and Najork, M., 1999. Mercator: A Scalable, Extensible Web Crawler, International Journal of WWW, Vol. 2. No. 4. (1999) 219-229

[2].Bernardi, S., Donatelli, S., and Merseguer, J. From UML Sequence Diagrams and Statecharts to Analysable Petri Net Models. In Proceedings of ACM International Workshop Software and Performance. Rome, Italy, pp. 35-45, 2002.

[3].Bogárdi-Mészöly, Á., Levendovszky, T., Charaf, H.: Handling Session Classes for Predicting ASP.NET Performance Metrics , .NET Technologies – Short communication papers conference proceedings, Hungary, pp. 01-08, 2006.

[4].Shkapenyuk, V. and Suel, T.: Design and Implementation of a High-performance Distributed Web Crawler, In Proceedings of 18th Data Engineering Conference, (2002) 357-368

[5].Bogárdi- Mészöly, Á., Szitás, Z., Levendovszky, T., Charaf, H. Investigating Factors Influencing the Response Time in ASP.NET Web Applications. Proceedings of Lecture Notes in Computer Science, 3746, pp. 223-233, 2005.

[6].Ela Kumar, Ashok Kumar al,(IJCSIT)International Journal of Computer Science and Information Technologies, Vol. 2 (4) , 2011, 17001705(ISSN: 0975-9646).

[7].Urgaonkar, B. Dynamic Resource Management in Internet Hosting Platforms. Dissertation, Massachusetts, 2005.

[8].Urgaonkar, B., Pacifici, G., Shenoy, P., Speitzer, M., and Tantawi, A. An Analytical Model for Multi-tier Internet Services and its Applications. Journal of ACM SIGMETRICS Performance Evaluation Review, Vol. 33, No. 1, pp. 291-302, 2005.

[9].Ela Kumar, Ashok Kumar al, (IJCTT) International Journal of Computer Trends and Technology- Vol. 3(3)- 2012,342-350 (ISSN: 2231-2803).

[10].Ashok Kumar al, (IJSECT)International Journal of Science, Engineering and Computer Technology- Vol. 2(3)- 2012,33-38 (ISSN: 2229-4937).

[11].Sinclair, B., Mean Value Analysis. Computer Systems Performance Handout, 2005.

[12].Ashok Kumar al, CurlCrawler Optimization: A framework for crawling the web with URL tracking and canonicalization. , (IJERT) International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 1, January- 2013 ,01-10(ISSN: 2278-0181).

[13].Ashok Kumar al, (IJSECT) International Journal of Science, Engineering and Computer Technology- Vol. 2(3)- 2012,43-47 (ISSN: 2229-4937).

[14].Dr. Ela Kumar et al. /International Journal of Engineering and Technology Vol.2 (2), 2010, 38-44.

[15].Grosso,A., Podestagrave,R., Vecchiola,C., Boccalatte,A.: Design and Implementation of a Grid Architecture over an Agent-Based Framework , .NET Technologies – Short communication papers conference proceedings, Italy, pp. 27-34, 2006.

[16].Smith, C.U., and Williams, L. G. Performance Solutions: A Practical Guide to Creating Responsive, Scalable Software. Addison-Wesley, 2001.

[17].Configuration Parameters on E- Commerce Sites. In Proceedings of the ACM 5th International Workshop on Software and Performance, Palma, Illes Balears, Spain, pp. 53-64, 2005.

[18].Pocza,K., Biczo,M., Porkolab,Z.: Towards Effective Runtime Trace Generation Techniques in the .NET Framework (Hungary), .NET Technologies – Short communication papers conference proceedings, Italy, pp. 09-16, 2006.

[19].Edwards, J., McCurley, K. S., and Tomlin, J. A. (2001). "An daptive model for optimizing performance of an incremental web crawler". In Proceedings of the Tenth Conference on World Wide Web (Hong Kong: Elsevier Science).