

SCOREBOARD TOOL

By

ANUSH SRINIVASAN

B.TECH, J.N.T.U, 2006

A REPORT

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Computing and Information Sciences
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas
2008

Approved By:

Major Professor

Dr. DANIEL ANDRESEN

Abstract

Scoreboard tool is a web application developed to provide a platform where administrators can conduct quizzes (typically in an organization) or employees can take a quiz (individually or in a team). These quizzes are generally conducted to improve the knowledge of the users (typically employees) taking the quiz.

This project is a tool where the user can take online quizzes which comprise of various categories like technology, science, math etc . The categories have various sub-categories. The admin can also perform CRUD on teams. The quizzes are created by the admin using a database which has the questions for the quizzes. The quizzes are basically divided into quizzes with multiple choice questions and timed quizzes where a user has a specified time to complete the quiz. Once a user logs in he can view his quiz history where he can view the scores of the various quizzes taken and also the time and date when taken. At the end of the quiz the user is given the option of rating the quiz and also entering his/her opinion of the quiz. The users will also have an option to view their scores graphically and compare their scores with other teams in the form of reports and also as graphs.

The important feature of this application is that the scores will be reported using reporting services. And another important feature is instead of manually testing the application test cases will be written for automated testing.

Table of Contents

List of Figures	v
Acknowledgements	vi
1. Introduction	1
1.1 Goal	1
1.2 Scope	1
1.3 Need of the application	2
1.4 Related Work and Problems.....	2
1.5 Platform Specifications – Deployment	3
1.5.1 Hardware Specification	3
1.5.2 Software Specification	3
2. System Requirement Analysis	4
2.1 Information Gathering.....	4
2.2 System Feasibility	4
2.2.1 Economic Feasibility.....	4
2.2.2 Technical Feasibility	5
2.2.3 Behavioral Feasibility	5
3. System Analysis	6
3.1 Use Case Diagram.....	6
3.2 ERDiagram.....	7
3.3 Class Diagram	8
3.4 Data Flow Diagram.....	9
3.5 State Transition Diagram	10
4. Design	12
4.1 Design Goals	12
4.1.1 Architectural ContextDiagram.....	17
4.1.2 Description of Architectural Design	12
4.1.3 Control Hierarchy.....	13
4.2 Procedural/Modular Approach.....	13
4.2.1 Initial Login Module	13
4.2.2 Quiz Data and Score Submission Module.....	14
4.2.3 Query (User History) Module	14
4.2.4 View Score and Graphs Module	14
5. Implementation	15
5.2 User Interface Design and Implementation.....	16
5.3 Technical Discussions.....	25
6. Testing.....	26
6.1 Unit Testing.....	26
6.1.1 Test Results.....	27
6.2 Integration Testing	27
6.3 Validation Testing.....	28
6.4 White Box Testing	29
6.5 Stress and Performance Testing.....	30

6.5.1 Comparison b/w constant users and varying loop count.....	32
6.5.2 Comparison b/w varying users and constant loop count.....	32
6.5.3 Comparison b/w two different pages using constant values.....	33
7. Results & Challenges	34
7.1 Challenges	34
8. Conclusions	35
8.1 Limitations	35
8.2 Scope for Future Work.....	35
9. References	36

List of Figures

Figure 3.1 Use Case Diagram	6
Figure 3.2 ER Diagram	7
Figure 3.3 Class Diagram	8
Figure 3.4 Context Level Dataflow Diagram	9
Figure 3.4.1 First Level Data Flow Diagram	10
Figure 3.4.2 Second Level Data Flow Diagram.....	10
Figure 3.5 State Transition Diagram	11
Figure 4.1 Architectural Context Diagram.....	12
Figure 4.2 Control Hierarchy Diagram	13
Figure 5.1 Database Implementation Using SQL Server 2005.....	15
Figure 5.2 Login Page	16
Figure 5.3 Home Page.....	17
Figure 5.4 User Home Page	17
Figure 5.5 Quiz Start Page	18
Figure 5.6 Quiz Questions Page.....	19
Figure 5.7 Results page	19
Figure 5.8 Timed Quiz Page	20
Figure 5.9 Problem Page	21
Figure 5.10 Report and Charts	22
Figure 5.11 Chart showing scores	22
Figure 5.12 Report showing scores of all users	23
Figure 5.13 Chart showing scores of individual user.....	24
Figure 5.14 Report showing scores of individual user.....	24
Figure 6.1 Test report.....	28
Figure 6.2 Graph showing summary of tests conducted.....	31
Figure 6.3 Report showing summary of tests conducted.....	31

Acknowledgements

My heartiest thanks to Dr Daniel Andresen for serving as my Major Professor and his encouragement, guidance and valuable advice which enabled me in completing my course work.

I would like to thank Dr Gurdip Singh and Dr Mitchell Neilsen for serving in my committee and for their valuable cooperation during this project.

1. Introduction

1.1 Goal

The goal of this application is to provide a web interface wherein users can take quizzes online and also have a graphical presentation of their scores and also compare scores with other users. To give a more detailed understanding of things the users also have the option of retrieving and viewing details in the form of reports from the database. The data is generally related to the quizzes taken by the users. By having a report of the scores the users can know the areas in which they are weak and know how others have performed in the same tests. The application provides the following facilities:

- A convenient and easy way to view scores in the form of reports and using the web interface.
- A simple and intuitive interface for the users to take quizzes.
- An option for the users to take quizzes in a limited time basis and setting the time they are comfortable with.
- View the reports to analyze their weak and strong areas .
- Enable the users to rate the quizzes and also give their opinions about the quizzes taken.

1.2 Scope

The application can be used as a tool for testing the user's skills in various areas and also help the users know their strengths and weaknesses by viewing their test results in the form of reports. The application can also be used in any organization where the users can form teams and take the quizzes and compare the scores from team to team which will enable creating a healthy competition among the employees. In the case of an organization, if any quiz is conducted related to any technical aspect the reports can be used as a parameter to analyze how good each employee is at a particular technology.

1.3 Need of the application

There are very few online quiz applications that offer the users the convenience of viewing and comparing their score in the form of reports and graphs using a user friendly and interactive website. Such issues are handled by his application by providing a user friendly and interactive interface with built with the latest tools.

1.4 Related Work and Problems

There are many online applications that provide online quizzes but developing an application from scratch which is very user friendly is important as the end users might not have much technical knowledge. Measures have been taken to make all the functionalities of the application self explanatory.

1.5 Platform Specifications – Deployment

1.5.1 Hardware Specification

Processor P III

RAM 128 MB

Minimum Space Required 10 MB

Display 16 bit color

1.5.2 Software Specification

Operating Environment : Windows XP

Platform : .Net Framework & IIS Visual Studio 2008

Programming Language: C#

Database : SQL Server 2005

Reporting Services: SQL Server Business Intelligence Development Studio

2. System Requirement Analysis

2.1 Information Gathering

The idea behind the development of this application was to use and learn new tools in .Net Framework. I went through various sites having the same application to have an idea of how to go about things as the application should be convenient enough even for first time users to take the quizzes without any difficulties. I felt that the most convenient way of showing the users scores was in the form of graphs and reports and did a lot of research to implement the most appropriate reporting service.

I also did a lot of research in implementing the latest ASP.NET tools in order to improve the user interaction and performance of the application. A lot of research was also done on automating the testing

2.2 System Feasibility

The system feasibility can be divided into the following sections:

2.2.1 Economic Feasibility

As the project requires a system with bare minimum requirements, it is economically feasible. The end users will require only an internet connection to use the application .

2.2.2 Technical Feasibility

To deploy the application, the only technical aspects needed are mentioned below:

Operating Environment : Win 2000/XP

Platform : Net Framework & IIS

Database : SQL Server 2005 with advanced services.

Other Software: SQL Server Business Intelligence Development Studio

For Users:

Any Internet Browser

Internet Connection

2.2.3 Behavioral Feasibility

The application is developed in such a way that even a first time user can use all the features of the application without any type of technical guidance. All the features are self explanatory .

3. System Analysis

On analyzing the requirements and functionality of an online quiz application , I had three important diagrams by the end of the analysis phase. The ER diagram, data flow diagram and the state transition diagram which were a basis for finding out entities and relationships between them, the flow of information and the various states the application can have.

3.1 Use Case Diagram Use case diagrams describe what a system does from the standpoint of an external observer. The use case diagram above figure depicts the relationships among the actors and usecases where actors represent the external entities of the system and use cases represent the functional parts of the system



Figure 3.1 Use case Diagram

3.2 ER Diagram

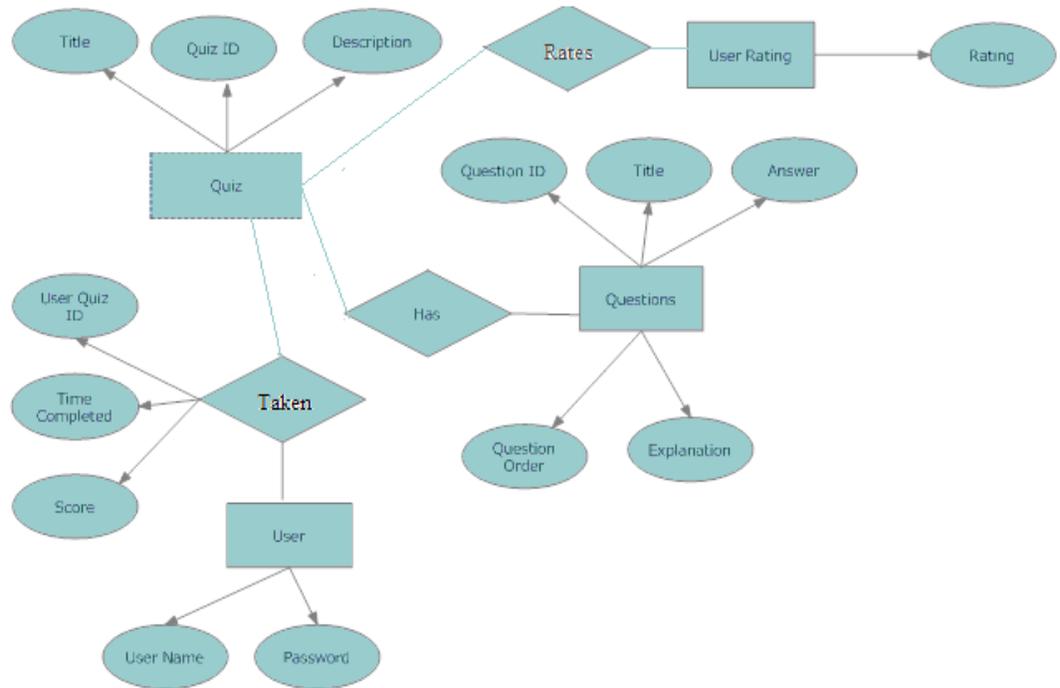


Figure 3.2 ER Diagram

3.3 Class Diagram

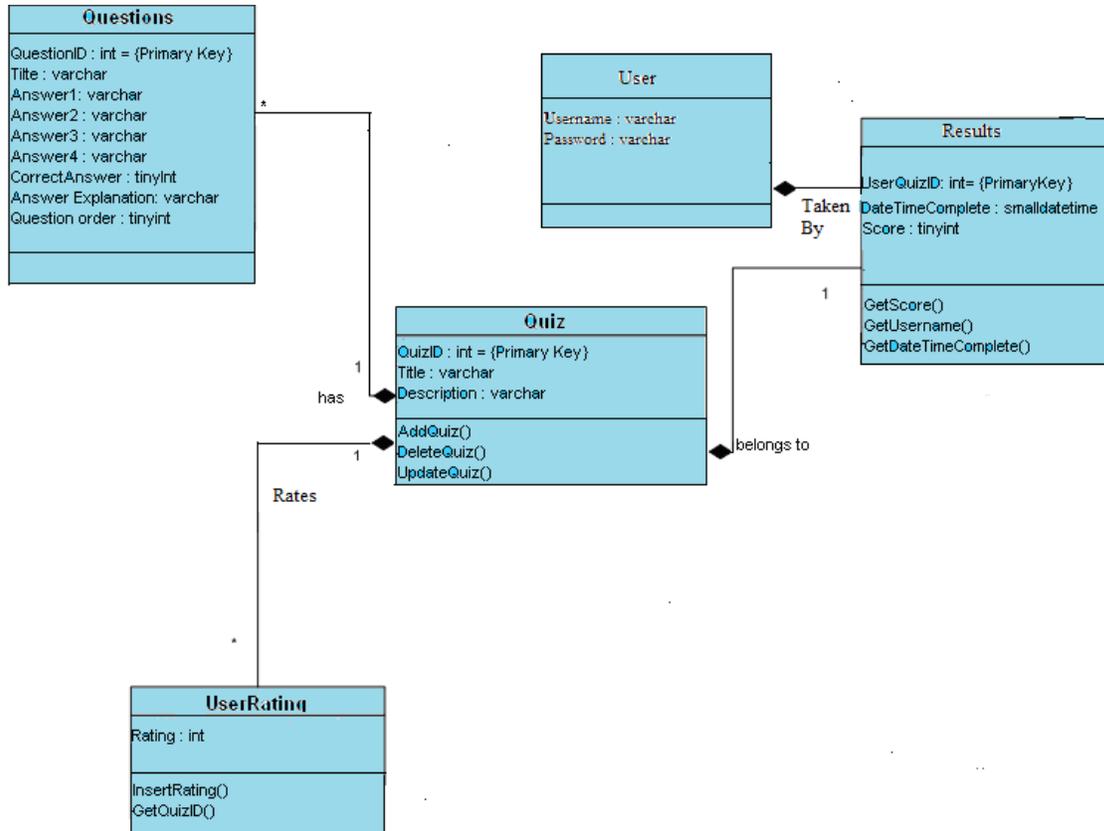


Figure 3.3 Class Diagram

3.4 Data Flow Diagram

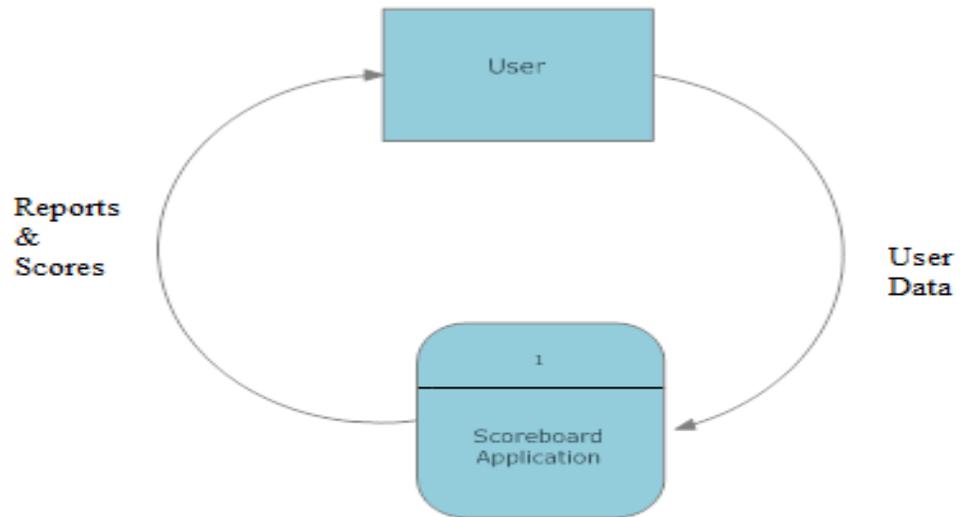


Figure 3.4 Context Level Data Flow Diagram

3.4.1 First Level Data Flow Diagram

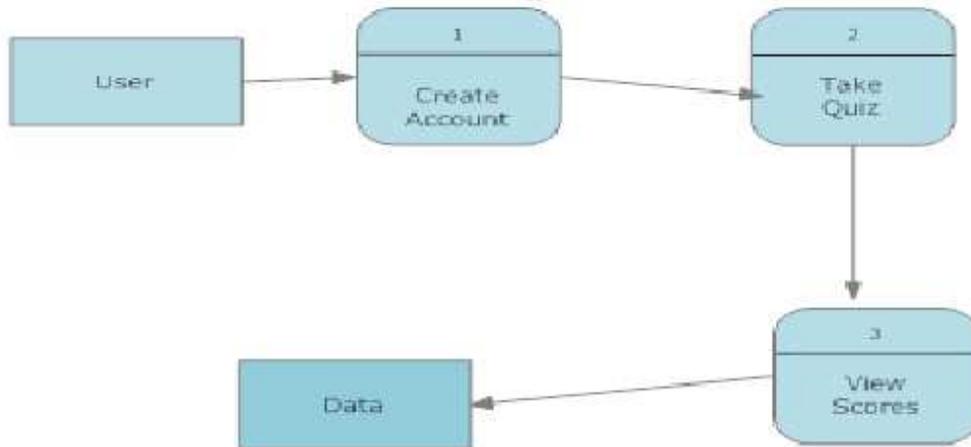


Figure 3.4.1 First Level Context Level Data Flow Diagram

3.4.2 Second Level Data Flow Diagram

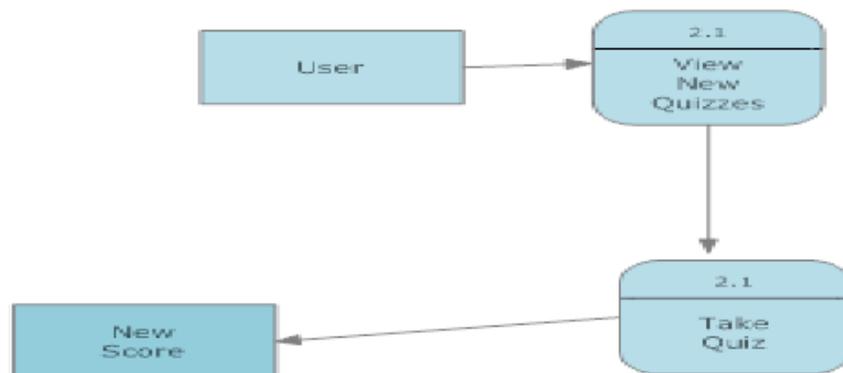


Figure 3.4.2 Second Level Context Level Data Flow Diagram

3.5 State Transition Diagram

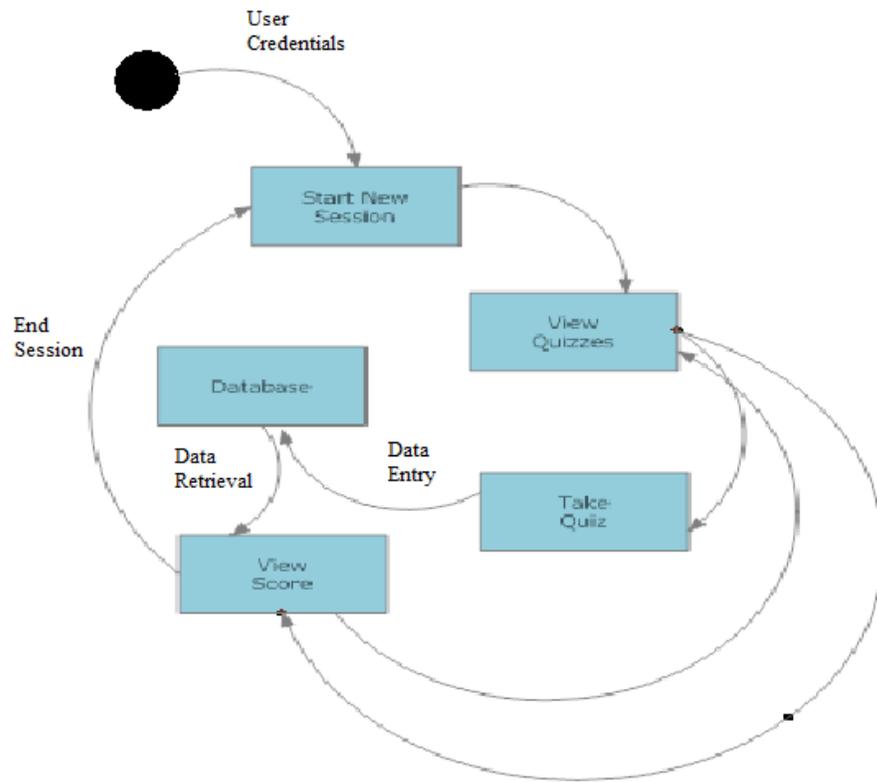


Figure 3.5 State Transition Diagram

4. Design

4.1 Design Goals

- Designing the web application involves designing the user details forms, the profile display forms, questions forms, the results forms and navigation between the various forms.
- Designing the interactive reports and the graphs which should also enable the user to take report print outs.



Figure 4.1 Architectural Context Diagram

4.1.2 Description of Architectural Design

In the above context diagram, data is provided to and received from the Scoreboard Tool Application. The arrows indicate whether information is received or sent by the application. The sources and sinks are represented by the closed boxes.

In the system, we can observe that the user interacts with the application through a user interface. The inputs to the system are the user credentials provided by the user and the Quiz Data and the Score once the user has taken a quiz. The quiz information, details, user scores, date and time taken are displayed through grid views. The output also comprises of charts and reports of the users scores.

4.2 Control Hierarchy

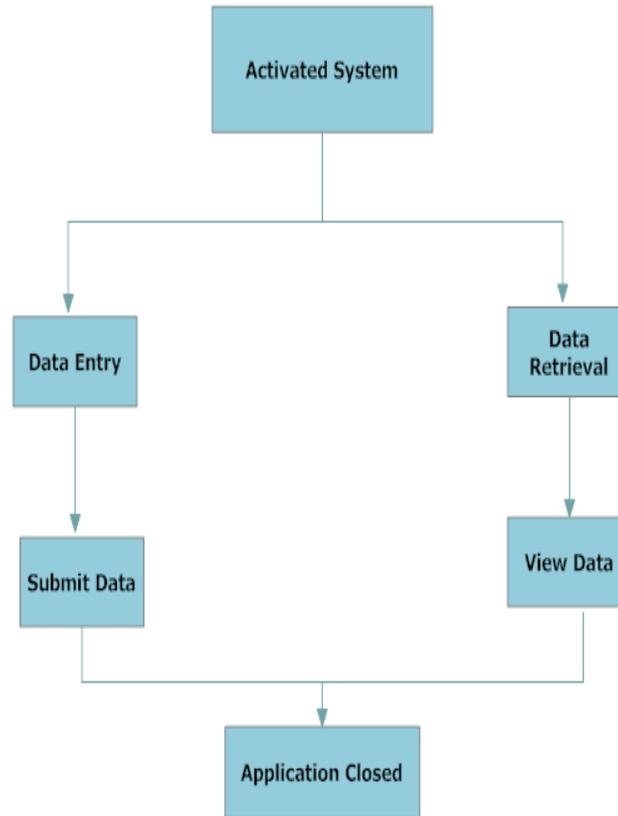


Figure 4.2 Control Hierarchy Diagram

4.2 Procedural/Modular Approach

Given below are the modules into which the Scoreboard Tool is divided .

4.2.1 Initial User Login Module

This module starts with the generation of a session for the user wherein the user details are taken and the session of the user is maintained until the end of the application. The session begins when the user enters valid credentials.

4.2.2 Quiz Data and Score Submission Module

Every Quiz has an UniqueID where the Quiz details such as QuizID, Title and Description are concatenated by the UserQuizID, DateTimeCompleted , Score and Username ID every time a user takes a quiz. Every time a user enters the details the entire session details are stored in the database.

4.2.3 Query (User History) Module

The database is queried for information about a user's history every time a user logs in. The quizzes taken, scores and date and time when taken are displayed. The entire user history is retrieved using the login information provided by the user.

4.2.4 View Scores as Reports and Graphs Module

A separate database is used for querying for generating the reports and graphs of the user's scores and also for comparing the scores of all the other users. Each time the user logs in or takes a quiz the new scores are updated and new reports and graphs are generated.

5. Implementation

The design of the database was as per the analysis phase. SQL Server 2005 was used for developing the database.

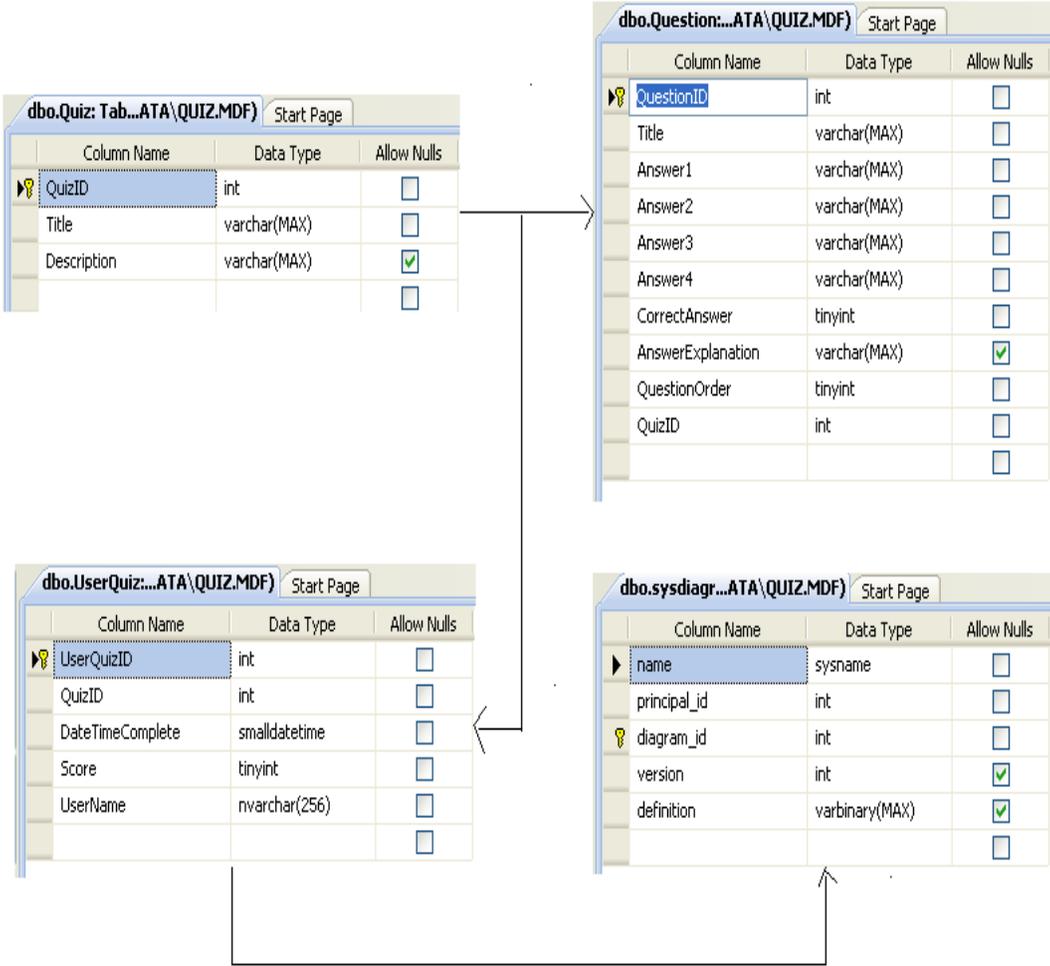


Figure 5.1 Database Implementation using SQL Server 2005

5.1 User Interface Design and Implementation

The user interface has been developed using Visual Studio 2008 . The login page is created using the user login template. This is the page where the user session starts and all the details are stored using the login details.



The screenshot shows the login page for 'Scoreboard Tool'. At the top right, there are links for 'Skip to site navigation' and 'Skip to content'. The logo on the left includes a stylized bird icon and the text 'Scoreboard Tool' with the tagline 'play and learn...'. Below the logo is a banner image of a wooden block with the number '5' on top, resting on a document with handwritten text and a 'Date' field. A navigation menu contains 'home', 'about', and 'contact'. The login form is titled 'Login' and contains the following elements:

- User Name:
- Password:
- Remember me next time.
-

Figure 5.2 Login Page

The user logs in using this page. Only when the user enters the correct login details will he will be able to view the available quizzes and his history.



Multiple Choice Quizzes



Scoreboard tool consists a variety of multiple choice quizzes. Take the quiz, check your score and also compare your score with others. You also have the option of rating a quiz.

Timed Quizzes



Take the timed quizzes and see how well you score in limited time. Take the quiz check your score. View your scores as reports and graphs. Compare your scores with others and also rate the quiz you take.

Figure 5.3 Home Page



[home](#) [about](#) [contact](#)

Welcome back Fred! | [Logout](#)

Select the Type of Quiz:

Available Tests

Algebra
Test your skills in basic algebra.

Probability
Test how good you are at probability.

Fractions
Test your skills at simple simple fractions.

Test Result History

Quiz	Completed	Score
Are you as healthy as you think you are?	11/27/2008	90
Are you as healthy as you think you are?	11/27/2008	90
Quiz	Completed	Score
Algebra	11/27/2008	100
Algebra	11/27/2008	100

Figure 5.4 User Home Page

The user home page shows the user history consisting of the scores in different quizzes, list of available quizzes and also the date when a quiz was taken.



Currently logged in as: Fred

Ready to Start the Test

You are about to take the quiz. The quiz contains multiple choice style questions. Complete each question from the drop down list box then select the Next button. You will not be able to go back and change your answers once you select the Next button, so choose carefully.

When you are ready to begin, select the Start button below.

Start

Figure 5.5 Quiz Start Page

The Quiz Start Page just gives the instructions of the quiz, type of quiz and also how to navigate through the quiz.



Currently logged in as: Fred

Current Test:

Question: What is the capital of Assyria

Answer 1: Ninevah

Answer 2: Mosul

Answer 3: I don't know that! Auuuugh!

Answer 4: Babylon

Your Answer:

- Answer1
- Answer2
- Answer3
- Answer4

[Next](#)

Figure 5.6 Questions Page

Currently logged in as: Fred

	Question	Correct Answer	Your Answer	Result
Select	1	2		Incorrect
Select	2	1	1	Correct
Select	3	2	1	Incorrect
Select	4	4	2	Incorrect

Question No.: 1

Question: What is your name?

Answer 1: Sir Robin of Camelot

Answer 2: Sir Lancelot of Camelot

Answer 3: Sir Galahad of Camelot

Answer 4: King Arthur

Correct Answer: 2

Explanation: While all the members of the group were asked this question, I decided to make #2 the correct answer.

Rate the quiz!

How much do you like this quiz? ★★★★★

[Submit](#)

Figure 5.7 Results Page

The Results Page displays the score of the quiz the user has taken where the user can select a particular question and find out the correct answer and also the explanation for the answer. There is also an option for the user to rate the taken quiz..

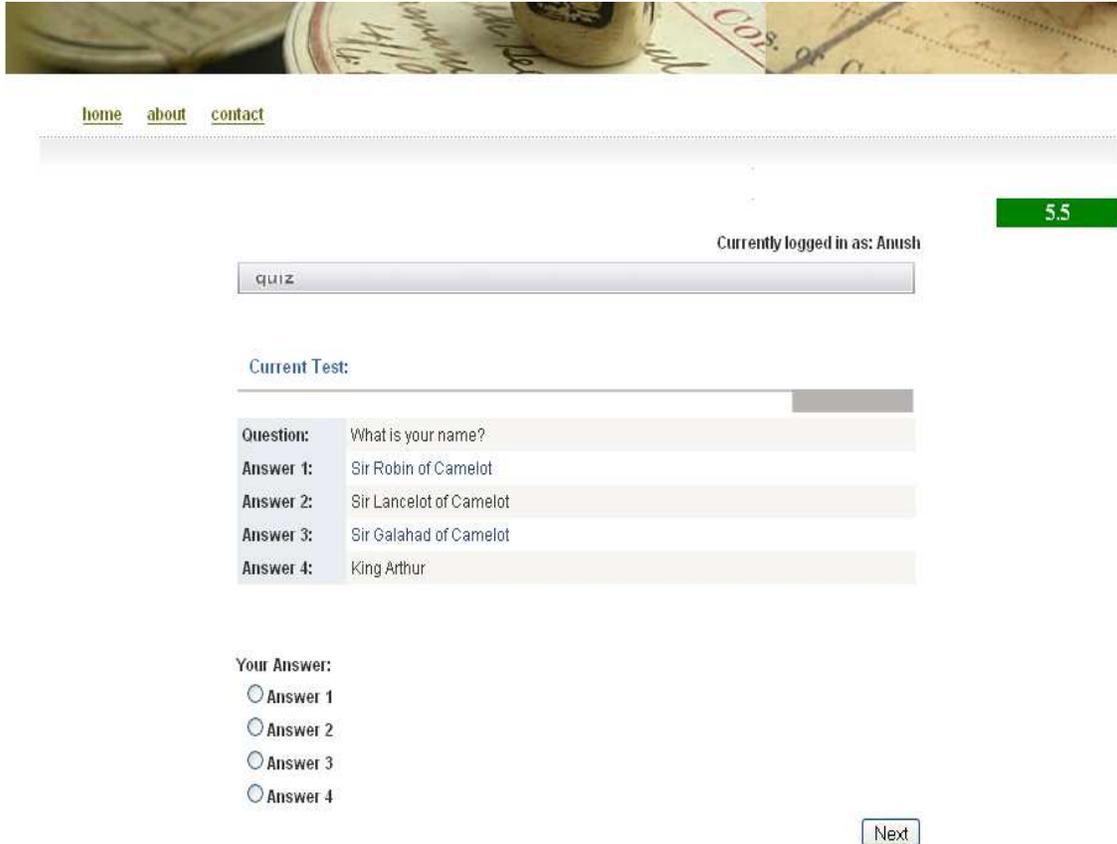


Figure 5.8 Timed Quiz Page

In the above page we can see that a timer is present indicating that it is timed quiz where each question should be solved in 10 seconds.



Sorry!!!

You have taken too much time to solve the question please select proceed to view your result.

Figure 5.9 Problem Page

Incase the user is unable to complete a question the specified time then the quiz stops and the user is redirected to the problem page.



Figure 5.10 Report Home Page containing the Reports and charts for the users to view

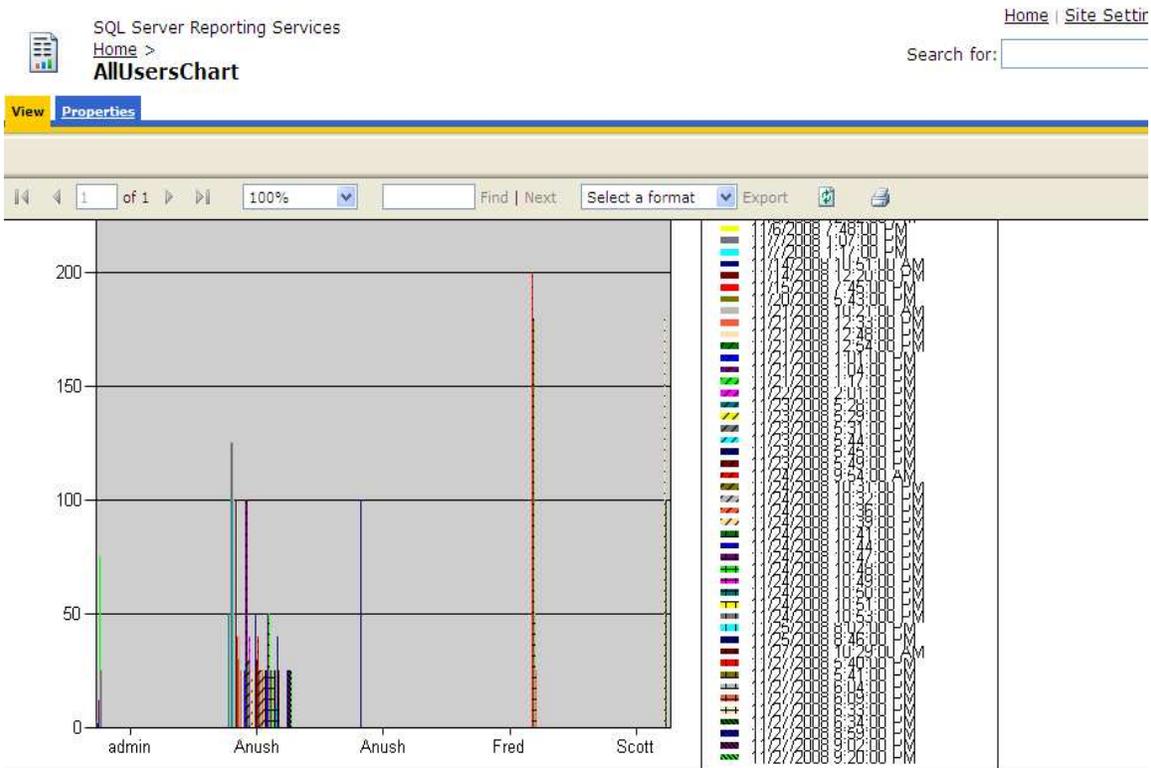


Figure 5.11 Chart showing the scores of all the users



View Properties

AllUsersReport

User Quiz ID	Quiz ID	Date Time Complete	Score	User Name
1	1	7/30/2005 2:47:00 PM	3	admin
2	2	7/30/2005 3:01:00 PM	2	admin
3	1	7/30/2005 6:09:00 PM	12	admin
5	2	7/31/2005 4:26:00 PM	75	admin
6	1	8/24/2005 4:04:00 PM	25	admin
7	1	11/6/2008 12:32:00 PM	25	Anush
8	1	11/6/2008 12:32:00 PM	25	Anush
9	2	11/6/2008 7:48:00 PM	0	Anush
10	9	11/7/2008	100	Anush

Figure 5.12 Report showing the scores of all the users

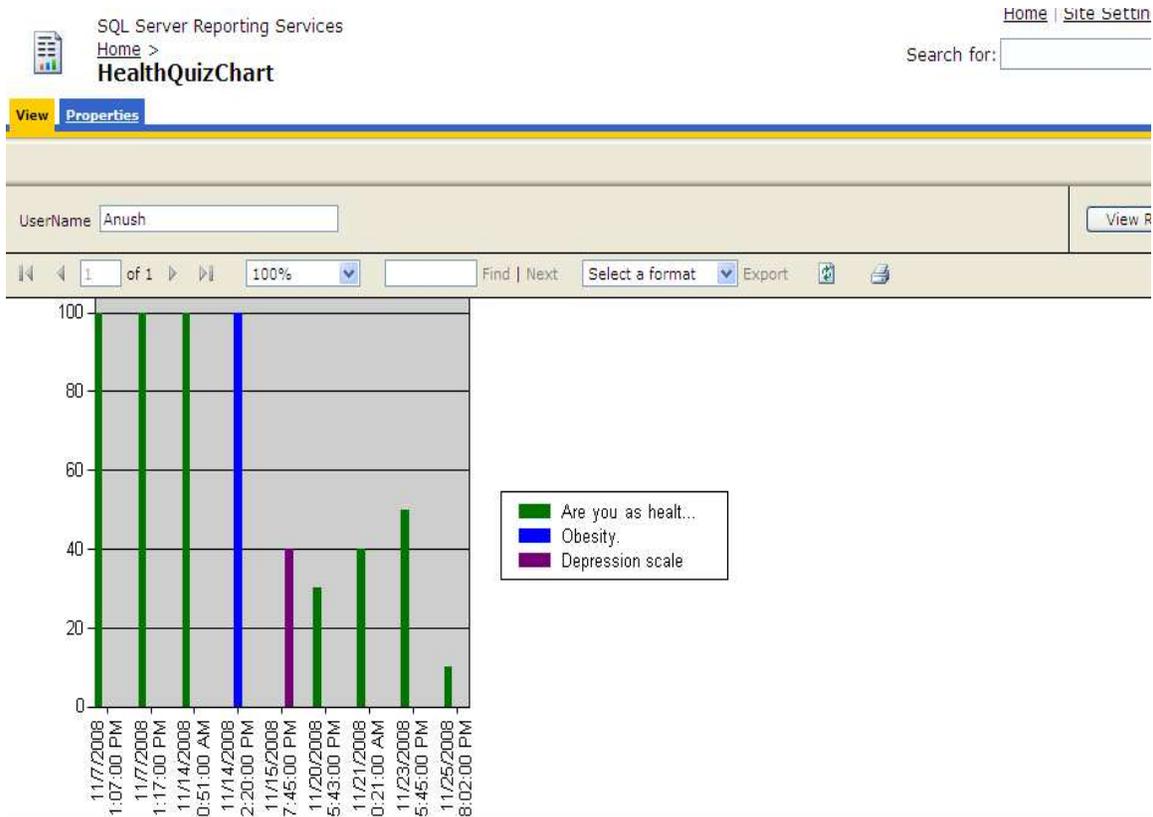


Figure 5.13 Chart showing the scores of a particular user in a particular quiz.

SQL Server Reporting Services
Home >
HealthQuizReport

Home | Site Settings
Search for:

View Properties

UserName: View R

1 of 1 100% Find | Next Select a format Export

Quiz2Report

Title	Date Time Complete	Score
Are you as healthy as you think you are?	11/7/2008 1:07:00 PM	100
Are you as healthy as you think you are?	11/7/2008 1:17:00 PM	100
Are you as healthy as you think you are?	11/14/2008 10:51:00 AM	100
Obesity.	11/14/2008 12:20:00 PM	100
Depression scale	11/15/2008 7:45:00 PM	40
Are you as healthy as you think you are?	11/20/2008 5:43:00 PM	30
Are you as	11/21/2008	40

Figure 5.14 Report showing the scores of a particular user in a particular quiz.

5.3 Technical Discussions

All the user details are stored in the table **UserQuiz** . The reports and charts are generated using the contents of this table. The contents of the drop downlists are not retrieved from the database but are retrieved from XML files in order to reduce the number of page loads and thus improving the speed of navigation between the pages and reducing redundancy.

The application has been developed in such a way that the database access has been reduced to the maximum. Since most of the pages have common content, AJAX has been implemented so that only the content that needs to be updated is loaded and not the entire content of pages thus enabling lower bandwidth usage and load time. The use of these asynchronous requests even allows the User Interface to be more interactive.

Reports and charts are generated using SQL Server reporting services and SQL Server Business Intelligence which help creating interactive reports by sharing the existing database and not creating a separate database.

To ensure maximum performance of the application latest automated testing tools have been used instead of manually testing the application.

6. Testing

Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. The purpose of testing can be quality assurance, verification and validation, or reliability estimation.

6.1 Unit Testing

The primary goal of unit testing is to take the smallest piece of testable software in the application, isolate it from the remainder of the code, and determine whether it behaves exactly as you expect. Each unit is tested separately before integrating them into modules to test the interfaces between modules. Unit testing has proven its value in that a large percentage of defects are identified during its use. All the modules of the application have been tested simultaneously as they have been developed. The feature of automated unit testing which is available in Visual Studio 2008 has been utilized for this purpose. Over here I have written separate methods to call the values for example the score of the user, rating of a particular test etc. An expected value is taken and tested whether the tested value meets the expected value. The functionality of the modules was also tested as a separate unit for example the functionality of the next button which the user clicks after solving a particular problem was tested whether it falls into separate cases in the logic. All the test cases were tested independently to ensure that all the functionalities of the application worked perfectly as separate units. Separate test cases were written for each method where the user's scores are submitted, rating of the each quiz is submitted and also the entire user history is updated. To ensure that the exact score of the user is returned, I have written a separate test case wherein I have made sure that the tested value meets the actual value. In this way all the individual user submissions have been tested as separate units.

Even in the reporting module, in order to ensure that accurate reports were generated the user details were tested separately for its functionality to ensure that the correct user values are being processed while generating the reports

6.1.1 Test Results

I have tested the following functions:

- Get Score to check if the correct score of the user is returned.
- Get Rating to check if the correct user rating is entered.
- Get Available Tests to check if the correct number of available test are returned.
- Get History to check if the user history to a particular user is returned.
- Get Average Score to check whether the average score of a user in all tests is returned correctly.

6.2 Integration Testing

Integration testing is a software development process which program units are combined and tested as groups in multiple ways. In integration testing a system consisting of different modules is tested for problems arising from component interaction. There are two major ways of carrying out an integration test, called the bottom-up method and the top-down method.

The integration testing in this project is done in a bottom up fashion i.e. in this project first atomic modules were constructed and tested and after unit testing the modules are integrated one by one and then tested the system for problems arising from component interaction.

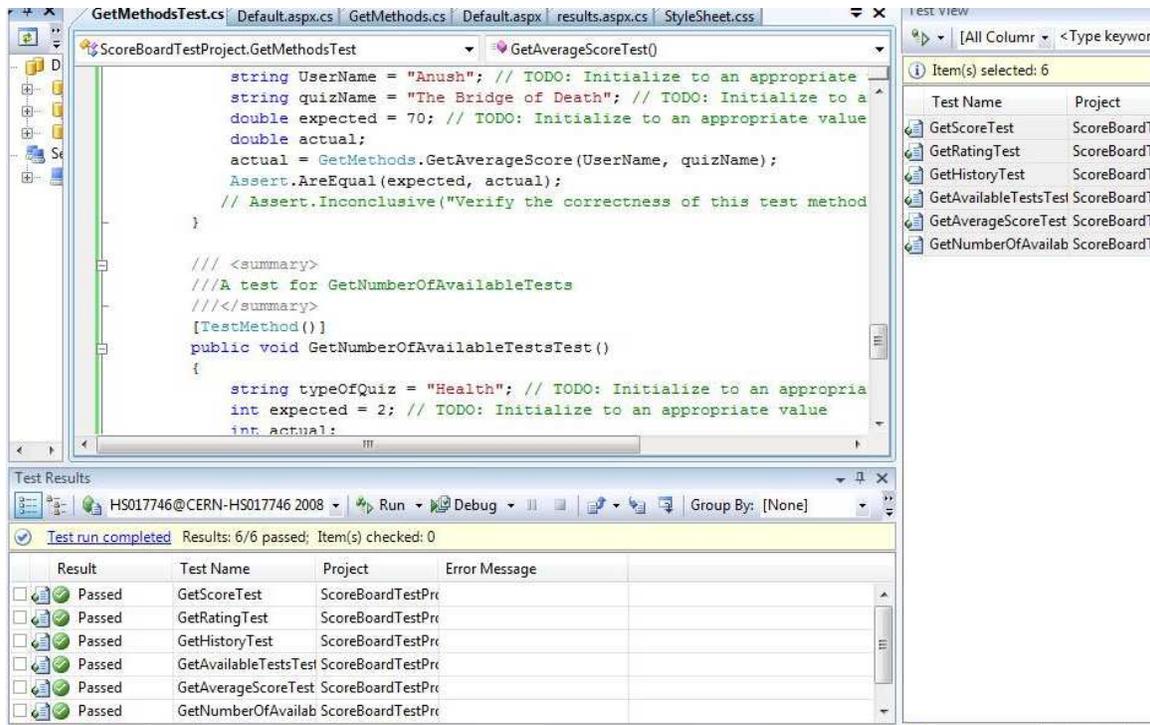


Figure 6.1 The test report showing the unit tests that have been run together have passed.

6.3 Validation Testing

Validation testing ensures that all the functional, behavioral and performance requirements are met..

Validation Testing comprises of three main components.

- Validation test criteria (number in place of number and char in place of char)
- Configuration review (to ensure the completeness of s/w configuration.)
- Alpha & Beta testing- Alpha Testing is the testing that is performed when the application is being developed and Beta Testing is the testing that is performed after the application has been deployed.-Test Cases

- A number of test cases have been created to ensure that in each functionality of the application the tested value meets the desired or expected value.

- Correct authentication is provided.
- As the same form is used by all the users for different tests accurate data is passed and stored in the database.
- There is no duplication of data..
- All the buttons used navigate to the desired pages.
- Quizzes that have already taken should be notified.
- Once the time limit of a question is exceeded the quiz should end.
- Ensure that only when a question is answered, the user can move onto the next question.

6.4 White Box Testing

White box testing (a.k.a. clear box testing, glass box testing or structural testing) uses an internal perspective of the system to design test cases based on internal structure.

It requires programming skills to identify all paths through the software. The tester chooses test case inputs to exercise paths through the code and determines the appropriate outputs.

Typical white box test design techniques include:

- Control flow testing
- Data flow testing
- Branch Testing

All the modules have tested by supplying values at the logical parts and verifying that the appropriate results are derived. All the conditional and looping statements have been tested for accuracy..

6.5 Stress and Performance Testing

Performance testing is an essential element in successfully deploying a Web application. It's important to understand how the application and your Web server farm would behave as more and more users visit the Web site.

Jakarta JMeter, a tool for testing applications was used to simulate the virtual users (clients) and test the performance of the system. It can be used to test performance both on static and dynamic resources (files, Servlets, Perl scripts, Java Objects, Data Bases and Queries, FTP Servers and more). It can be used to simulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load types. It can be used to make a graphical analysis of performance and test the server/script/object behavior under heavy concurrent load. I have done performance testing to achieve an estimate of the peak and sustained load the application. This has done with few pages like the User Login & Quiz details Page (simple with less database access) and the Questions Page) (extensive database access and business logic intensive).

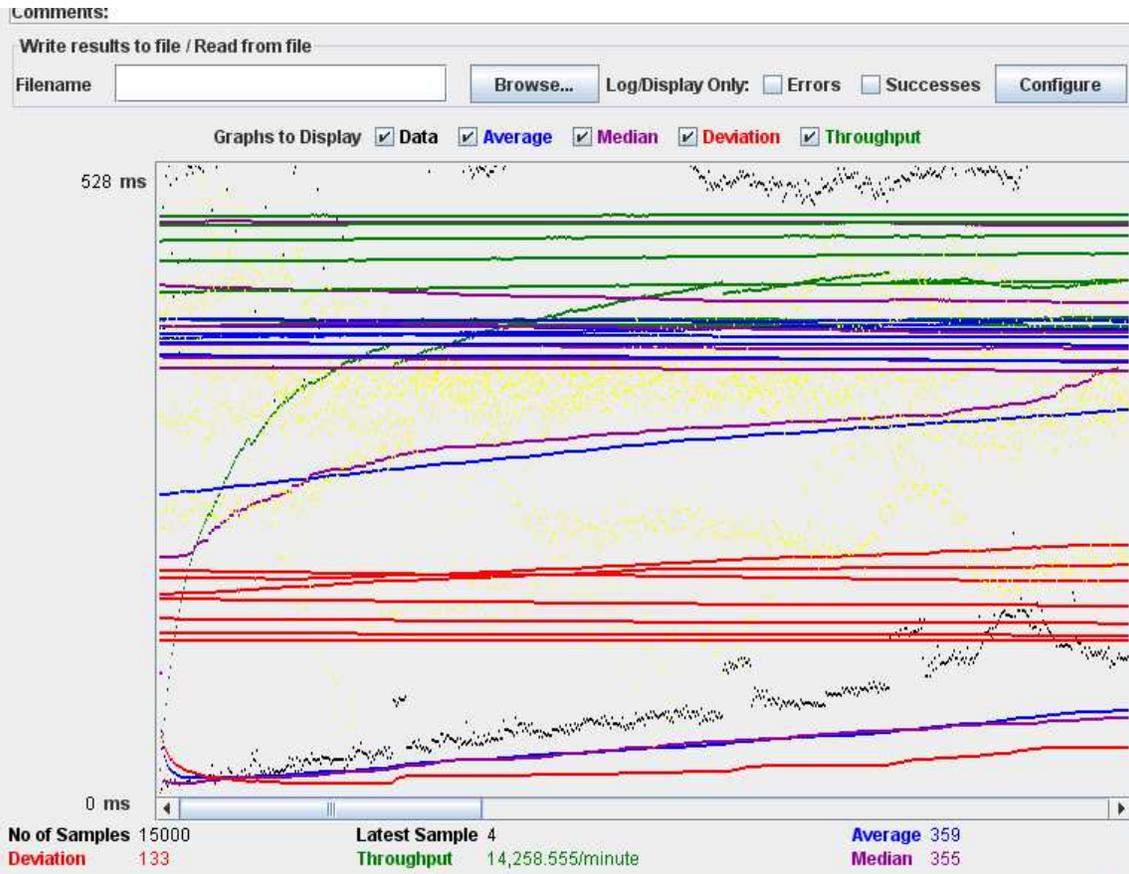


Figure 6.2 A graph showing the summary of a test conducted with 100 users and with Loop Count set to 100

Summary Report

Name: Summary Report

Comments:

Write results to file / Read from file

Filename Browse... Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
HTTP Requ...	25000	4673	5	10062	1789.94	56.72%	51.5/sec	329.45	6548.
TOTAL	25000	4673	5	10062	1789.94	56.72%	51.5/sec	329.45	6548.

Figure 6.3 The summary of a test conducted with 100 users and with Loop Count set to 100

6.5.1 Comparison of constant Users with varying Loop Count

Users	Loop Count	Average Response ms	Throughput
100	100	359	237.6/sec
100	250	467	194.8/sec
100	500	491	194.9/sec

From the above comparison we can see the increase in average response with the increase in loop count.

6.5.2 Comparison of varying Users with constant Loop Count

Users	Loop Count	Average Response ms	Throughput
250	250	1130	211.4/sec
350	250	1475	227.3/sec
500	250	2160	220.7/sec

We can see that there is a large increase in the average response when the number of users are increased when compared to the increase in the loop count.

Observations from the above two comparisons:

Response Time increases rapidly with number of users but not very much when the users are kept constant and only loop-count is increased. The main reason behind this is if the number of users are kept constant and only the loop count is increased, the number of requests handled by the server per second also remains constant for every execution of the loop count and for every increase in the loop count. Hence the response time will not increase drastically but when the number of users are increased, and loop count is kept constant, the requests handled by the server per second increases with the increase in the number of users and hence the large variation in the response time.

6.5.3 Comparison of two different pages using same number of users and loop count values.

Users	Loop Count	User Login & Quiz Details Page	Questions page
100	100	359	1923
250	250	1130	5143

We can notice that there is large variation in the response time with large database access and more logic.

Observations from the above comparison:

From the above comparison it is clear that as the amount of database access and logic increases the response time also increases.

Factors affecting Response Time:

- Limited System Hardware Resources (CPU, RAM, Disks) and Configuration
- JMeter Tests and Application running on the same machine

From the performance tests conducted through JMeter it is evident that the application can handle a large load because the response times were good even when run on a system with less configuration and the performance is sure to improve if the application is run on a better serve

7. Results & Challenges

The targeted users of this application are the users who want to take an online quiz. It may also be used in any organization where it can be used as a common application for the employees who would like to take the quizzes as teams.

7.1 Challenges

- The initial challenge was to design a proper structure for the application which should be user friendly..
- A database which should be capable of storing large records..
- Learning and implementing the latest .Net technologies with little or no guidance.
- Learning how to use SQL Server Reporting Services and generating reports using SQL Server Business Intelligence tool.
- Configuring SQL Server and report server to my application was a major task

8. Conclusions

The Scoreboard Tool application was designed to provide users with a user friendly interface wherein they can take quizzes online, check their usage history and view their scores. The application also enables the users to view interactive charts and reports for viewing and comparing scores with other users.

8.1 Limitations

The only issue with the application that has to be resolved is the integration of the reports and charts with the asp.net application. This was caused due to some changes made while configuring SQL Server.

8.2 Scope for Future Work

The following things can be done in future.

- A save option can be provided to save the progress made by the user in a quiz just in case the internet connectivity is lost or the user wants to take the quiz some other time.
- Proper integration of the reports with the application.
- Improve space utilization of the pages.

9. References

- Ajax Controls

<http://www.asp.net/ajax/>

- Microsoft load testing tool

<http://support.microsoft.com/kb/231282>

- New Microsoft Tools and SQL Server reporting Services

<http://msdn.microsoft.com/en-us/default.aspx>

- Other sites for information regarding testing and general information.

<http://www.wikipedia.org/>

www.businessobjects.com

<http://jakarta.apache.org/jmeter/usermanual/index.html>