AN EXPERIMENTAL PERFORMANCE EVALUATION OF RELATIONAL KEYWORD SEARCH SYSTEMS

Shashikala S.M.[1], Dr. Suresh L.[2]

[1,2]Department of Computer Science, Cambridge Institute of Technology, CiTech, Bangalore, (India)

ABSTRACT

In the past decade, extending the keyword search paradigm to relational data has been an active area of research within the database and information retrieval (IR) community. A large number of approaches have been proposed and implemented, but despite numerous publications, there remains a severe lack of standardization for system evaluations. This lack of standardization has resulted in contradictory results from Different evaluations and the numerous discrepancies muddle what advantages are proffered by different approaches. This paper does a full research on empirical performance evaluation of relational keyword search systems. What we have concluded is that though there are many search techniques available, nothing so far gives accurate retrieval results. One thing that prevents search techniques from working beyond small datasets is memory consumption. We also look into the relationship execution time and other factors, but what we conclude is that they does not affect search performance in any way. What we can conclude from this is that previous claims on the poor performance of evaluation systems are true, which brings out to the inevitable conclusion – there has to be some standardization.

Keywords: Query,keyword,file ranking,Testing

I. INTRODUCTION

The ubiquitous search text box has transformed the way People interact with information. Nearly half of all Internets Users use a search engine daily, performing in excess of 4 billion searches. The success of keyword search interfaces to make information access better—namely, a specialized query language or knowledge of the underlying structure of the data. Internet users increasingly demand keyword search interfaces for accessing information, and it is natural to extend this paradigm to relational data. This extension has However, we are not aware of any research projects that have transitioned from proof-of-concept implementations to deployed systems. Despite the significant number of research papers being published in this area, existing empirical evaluations ignore or Only partially address many important issues related to search performance. Lots of research papers have been published on this concept, but the problem lies in the fact that existing empirical evaluations partially address key issues on search performance. Baid et al. [1] bring out that existing systems do work, but their performance is unpredictable and hence cannot be used in real world retrieval tasks. This failure highlights the need for a robust, independent evaluation system. Existing problems in performance is obscured by design decisions such as data set choices or query construction. As a result we fudid empirical evaluation of existing relational keyword search techniques using a benchmark that is open for all to see and which will properly ascertain their exact performances in real world situations.

II. PROBLEM STATEMENT

A. Keyword Search without ranking.

B. Execution time is more.

Problem Solution

A. Keyword Search with ranking.

B. Execution Time consumption is less.

C. File length and Execution time can be seen.

D. Ranking can be seen by using chart

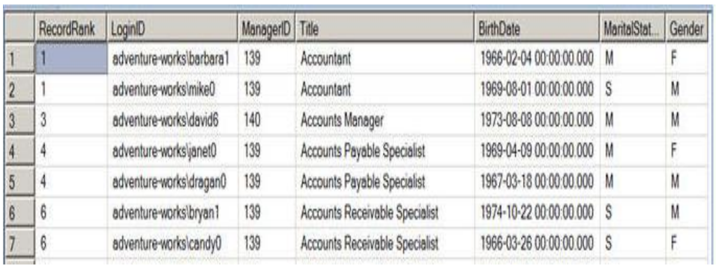
III. EXISTING SYSTEM

In existing system, extending the keyword search paradigm to relational data has been an active area of research within the database and information retrieval (IR) community. A large number of approaches have been proposed and implemented, but despite numerous publications, there remains a severe lack of standardization for system evaluations. This lack of standardization has resulted in contradictory results from Different evaluations and the numerous discrepancies muddle what advantages are proffered by different approaches.

IV. PROPOSED SYSTEM

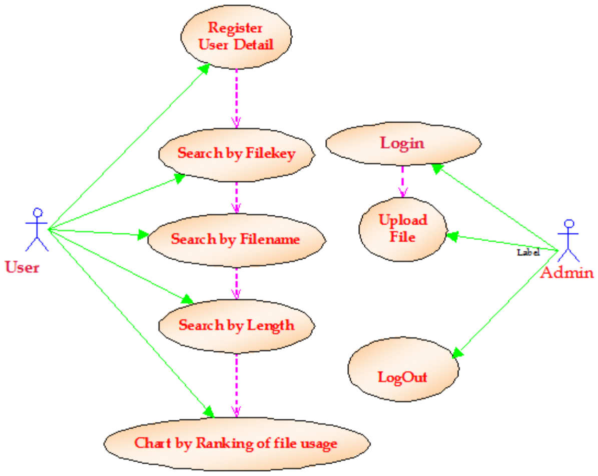
In proposed system, empirical performance evaluation of relational keyword search systems. Our results indicate that many existing search techniques do not provide acceptable performance for realistic retrieval tasks. In particular, memory consumption precludes many search techniques from scaling beyond small datasets with tens of thousands of vertices. We also explore the relationship between execution time and factors varied in previous evaluations; our analysis indicates that these factors have relatively little impact on performance. In summary, our work confirms previous claims regarding the unacceptable performance of these systems and underscores the need for standardization as exemplified by the IR community when evaluating these retrieval systems.

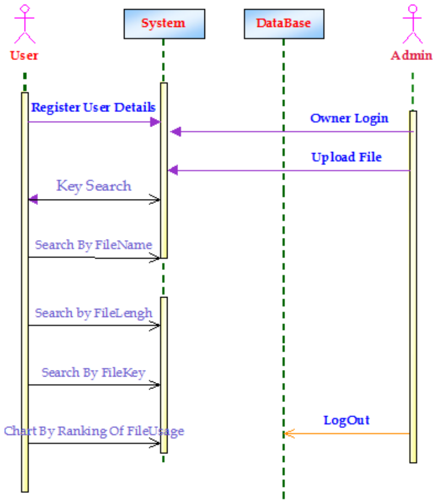
RANK - It does not only help in simplifying the query, but also improves the performance of the query. The RANK function instead of assigning a sequential number to each row as in the case of the ROW_NUMBER function, it assigns rank to each record starting with 1. If it encounters two or more records to have the same ORDER BY <columns> values, it is said to be a tie and all these records get the same rank. For example, in the first image below, you can see the first and second records have the same "Accountant" value in the Title column and hence they both got the same rank.



	RecordRank	LoginID	ManagerID	Title	BirthDate	MaritalStat...	Gender
1	1	adventure-works\barbara1	139	Accountant	1966-02-04 00:00:00.000	M	F
2	1	adventure-works\mike0	139	Accountant	1969-08-01 00:00:00.000	S	M
3	3	adventure-works\david6	140	Accounts Manager	1973-08-08 00:00:00.000	M	M
4	4	adventure-works\janet0	139	Accounts Payable Specialist	1969-04-09 00:00:00.000	M	F
5	4	adventure-works\dragan0	139	Accounts Payable Specialist	1967-03-18 00:00:00.000	M	M
6	6	adventure-works\bryan1	139	Accounts Receivable Specialist	1974-10-22 00:00:00.000	S	M
7	6	adventure-works\candy0	139	Accounts Receivable Specialist	1966-03-26 00:00:00.000	S	F

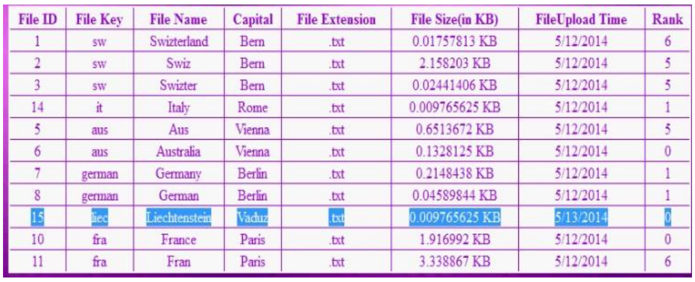
V. IMPLEMENTATION PART



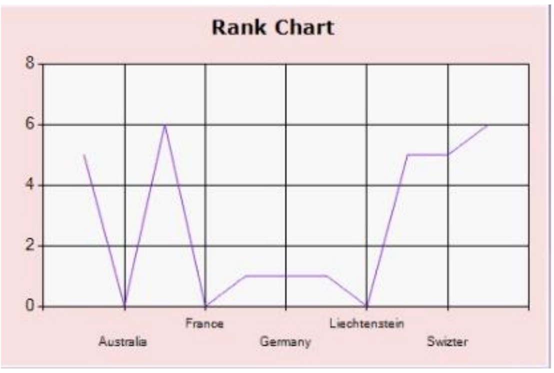


VI. ALGORITHMS

1. It results the files on basis of the file usage by Breadth-First algorithm.
2. Chart represents the ranking of the keyword searched by the user using Djikstra's shortest path algorithm.
3. Keyword search is essential for computing the results quickly by using Steriner Tree Problem and improves time-taken for the search by using Pseudo Polynomial Time alogirthm.
4. Discovers the files by its keyword and executes it in a fraction of second for the user by using sparse algorithm.



File ID	File Key	File Name	Capital	File Extension	File Size(in KB)	FileUpload Time	Rank
1	sw	Swizterland	Bern	.txt	0.01757813 KB	5/12/2014	6
2	sw	Swiz	Bern	.txt	2.158203 KB	5/12/2014	5
3	sw	Swizter	Bern	.txt	0.02441406 KB	5/12/2014	5
14	it	Italy	Rome	.txt	0.009765625 KB	5/12/2014	1
5	aus	Aus	Vienna	.txt	0.6513672 KB	5/12/2014	5
6	aus	Australia	Vienna	.txt	0.1328125 KB	5/12/2014	0
7	german	Germany	Berlin	.txt	0.2148438 KB	5/12/2014	1
8	german	German	Berlin	.txt	0.04589844 KB	5/12/2014	1
15	liec	Liechtenstein	Vaduz	.txt	0.009765625 KB	5/13/2014	0
10	fra	France	Paris	.txt	1.916992 KB	5/12/2014	0
11	fra	Fran	Paris	.txt	3.338867 KB	5/12/2014	6



VII. SYSTEM STUDY

7.1 Feasibility Study

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. Three key considerations involved in the feasibility analysis are,

7.1.1 Economical Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

7.1.2 Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

7.1.3 Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

VIII. SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product it is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

There are various types of test. Each test type addresses a specific testing requirement.

8.1 Types of Tests

8.1.1 Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

i. Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

ii.Test objectives

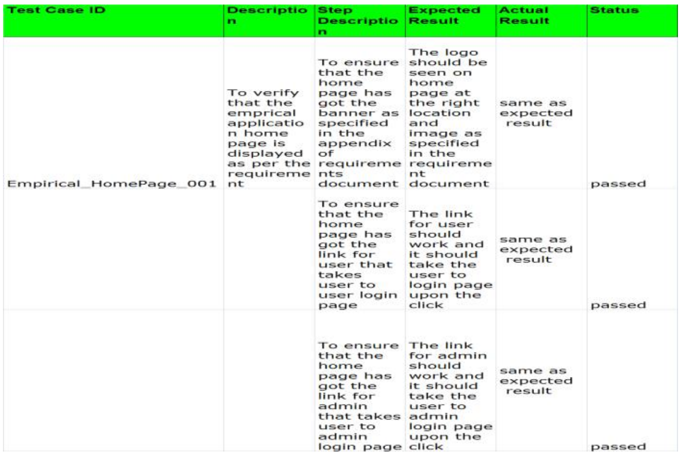
- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

iii. Features to be tested

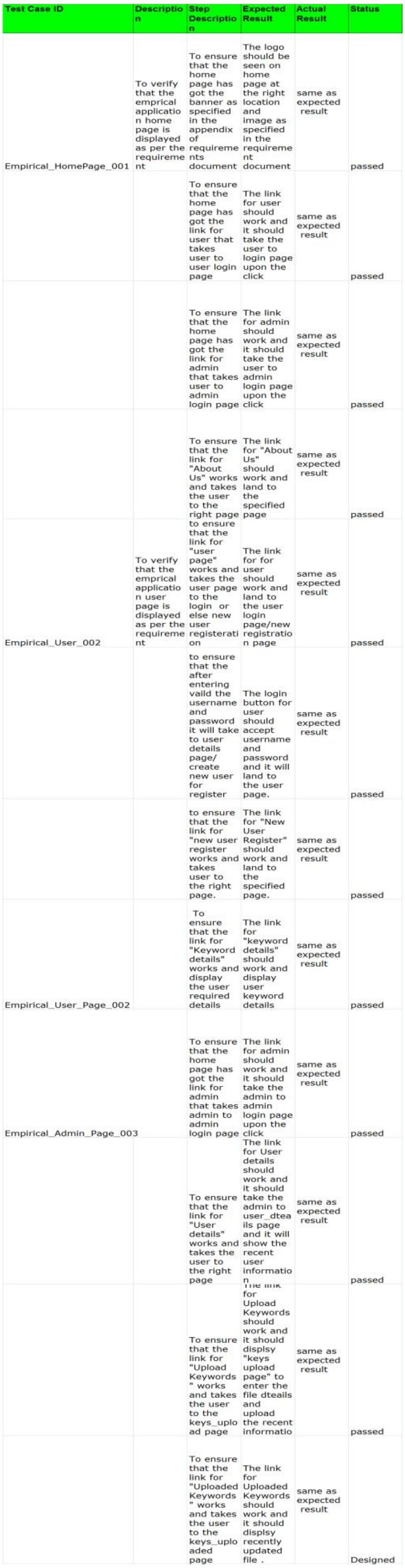
- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

iv. Test Results

below result tells all function are working properly . Hence result proved.



Test Case ID	Description	Step Description	Expected Result	Actual Result	Status
Empirical_HomePage_001	To verify that the emprical application home page is displayed as per the requirement	To ensure that the home page has got the banner as specified in the appendix of requirements document	The logo should be seen on home page at the right location and image as specified in the requirement document	same as expected result	passed
		To ensure that the home page has got the link for user that takes user to user login page	The link for user should work and it should take the user to login page upon the click	same as expected result	passed
		To ensure that the home page has got the link for admin that takes user to admin login page	The link for admin should work and it should take the user to admin login page upon the click	same as expected result	passed



Test Case ID	Description	Step Description	Expected Result	Actual Result	Status
Empirical_HomePage_001	To verify that the emprical application home page is displayed as per the requirement	To ensure that the home page has got the banner as specified in the appendix of requirements document	The logo should be seen on home page at the right location and image as specified in the requirement document	same as expected result	passed
		To ensure that the home page has got the link for user that takes user to user login page	The link for user should work and it should take the user to login page upon the click	same as expected result	passed
		To ensure that the home page has got the link for admin that takes user to admin login page	The link for admin should work and it should take the user to admin login page upon the click	same as expected result	passed
		To ensure that the link for "About Us" works and takes the user to the right page	The link for "About Us" should work and land to the specified page	same as expected result	passed
Empirical_User_002	To verify that the emprical application user page is displayed as per the requirement	to ensure that the link for "user page" works and takes the user page to the login or else new user registeration	The link for for user should work and land to the user login page/new registration page	same as expected result	passed
		to ensure that the after entering vaild the username and password it will take to user details page/ create new user for register	The login button for user should accept username and password and it will land to the user page.	same as expected result	passed
		to ensure that the link for "new user register works and takes user to the right page.	The link for "New User Register" should work and land to the specified page.	same as expected result	passed
Empirical_User_Page_002		To ensure that the link for "Keyword details" works and display the user required details	The link for "keyword details" should work and display user keyword details	same as expected result	passed
Empirical_Admin_Page_003		To ensure that the home page has got the link for admin that takes admin to admin login page	The link for admin should work and it should take the admin to admin login page upon the click	same as expected result	passed
		To ensure that the link for "User details" works and takes the user to the right page	The link for User details should work and it should take the admin to user_dteails page and it will show the recent user information	same as expected result	passed
		To ensure that the link for "Upload Keywords " works and takes the user to the keys_upload page	The link for Upload Keywords should work and it should displsy "keys upload page" to enter the file dteails and upload the recent informatio	same as expected result	passed
		To ensure that the link for "Uploaded Keywords " works and takes the user to the keys_uploaded page	The link for Uploaded Keywords should work and it should displsy recently updated file .	same as expected result	Designed

8.1.2 Integration Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

i. Test strategy and approach

Integration testing will be performed manually and functional tests will be written in detail.

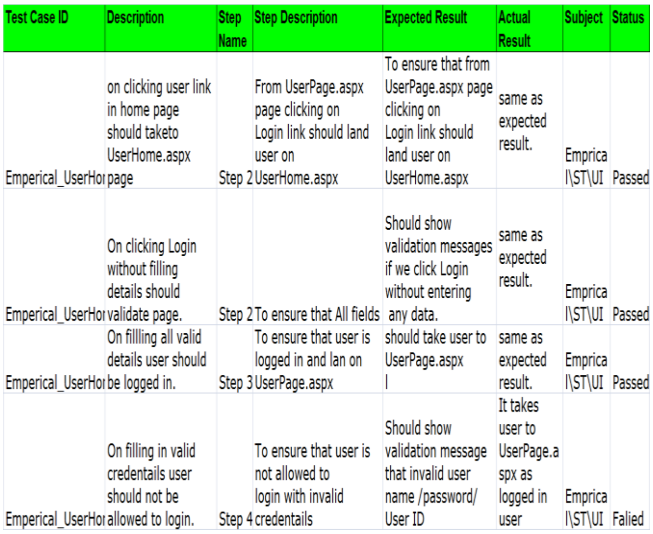
ii. Test objectives

- All the module are integrated
- Interface between all modules tested separatly

iii. Features to be tested

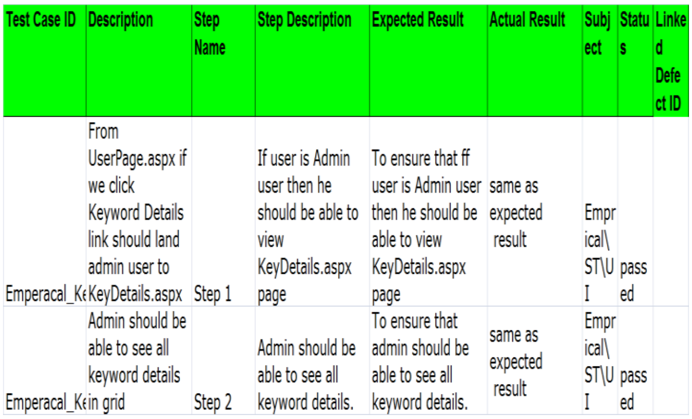
- User interface and query module
- Query module and search module
- Search module and report module

iv. Test Results



Test Case ID	Description	Step Name	Step Description	Expected Result	Actual Result	Subject	Status
Emperical_UserHor	on clicking user link in home page should taketo UserHome.aspx page	Step 2	From UserPage.aspx page clicking on Login link should land user on UserHome.aspx	To ensure that from UserPage.aspx page clicking on Login link should land user on UserHome.aspx	same as expected result.	Emprical\ST\UI	Passed
Emperical_UserHor	On clicking Login without filling details should validate page.	Step 2	To ensure that All fields	Should show validation messages if we click Login without entering any data.	same as expected result.	Emprical\ST\UI	Passed
Emperical_UserHor	On fillling all valid details user should be logged in.	Step 3	To ensure that user is logged in and lan on UserPage.aspx	should take user to UserPage.aspx |	same as expected result.	Emprical\ST\UI	Passed
Emperical_UserHor	On filling in valid credentails user should not be allowed to login.	Step 4	To ensure that user is not allowed to login with invalid credentails	Should show validation message that invalid user name /password/ User ID	It takes user to UserPage.aspx as logged in user	Emprical\ST\UI	Falied

Fig. Result 1



Test Case ID	Description	Step Name	Step Description	Expected Result	Actual Result	Subject	Status	Linked Defect ID
Emperacal_Ke	From UserPage.aspx if we click Keyword Details link should land admin user to KeyDetails.aspx	Step 1	If user is Admin user then he should be able to view KeyDetails.aspx page	To ensure that ff user is Admin user then he should be able to view KeyDetails.aspx page	same as expected result	Emprical\ST\UI	passed	
Emperacal_Ke	Admin should be able to see all keyword details in grid	Step 2	Admin should be able to see all keyword details.	To ensure that admin should be able to see all keyword details.	same as expected result	Emprical\ST\UI	passed	

Fig. Result 2

8.1.3 Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input	Identified classes of valid input must be accepted.
Invalid Input	Identified classes of invalid input must be rejected.
Functions	Identified functions must be exercised.
Output	Identified classes of application outputs must be exercised.
Systems/Procedures	Interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

Test result as fallow:

Test cases	Expected Result	Actual Result	Subject	Status
001	users should land on UserRegis.aspx	same as expected result	Emprical\ST\UI	passed
002	should show validation messages if we click sign up without entering any data.	All fields works fine but except Password and Confirm Password.	Emprical\ST\UI	Failed
003	user should get registered	same as expected result	Emprical\ST\UI	passed
004	User is redirected to UserHome.aspx	same as expected result	Emprical\ST\UI	passed

IX. SOFTWARE ENVIRONMENT

9.1 The .NET Framework

The .net Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet.

9.1.1 Objectives of .NET Framework

i. To provide a consistent object-oriented programming environment whether object codes is stored and executed locally on Internet-distributed, or executed remotely.

ii. To provide a code-execution environment to minimizes software deployment and guarantees safe execution of code.

iii. Eliminates the performance problems.

9.1.2 Features of SQL-Server

The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component. The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services. SQL-server database consists six types of objects. They are,

i. Table

A database is a collection of data about a specific topic. We can work with a table in two types,

i.i Design View

To build or modify the structure of a table we work in the table design view. We can specify what kind of data will be hold.

i.ii Datasheet View

To add, edit or analyses the data itself we work in tables datasheet view mode.

ii. Query

A query is a question that has to be asked the data. Access gathers data that answers the question from one or more table. The data that make up the answer is either dynaset (if you edit it) or a snapshot (it cannot be edited).Each time we run query, we get latest information in the dynaset. Access either displays the dynaset or snapshot for us to view or perform an action on it, such as deleting or updating.

9.2 Ajax

ASP.NET Ajax marks Microsoft's foray into the ever-growing Ajax framework market. Simply put, this new environment for building Web applications puts Ajax at the front and center of the .NET Framework.

X. CONSLUSION

Unlike many of the evaluations reported in the literature, ours is designed to investigate not the underlying algorithms but the overall, end-to-end performance of these retrieval systems. Hence, we favor a realistic query workload instead of a larger workload with queries that are unlikely to be representative (e.g., queries created by randomly selecting terms from the dataset).Overall, the performance of existing relational keyword search systems is somewhat disappointing, particularly with regard to the number of queries completed successfully in our query workload (see Table VI). Given previously published results (Table II), we were especially surprised by the number of timeout and memory exceptions that we witnessed. Because our larger execution times might only reflect our choice to use larger datasets, we focus on two concerns that we have related to memory utilization. Further research is unquestionably necessary to investigate the myriad of experimental design decisions that have a significant impact on the evaluation of relational keyword search systems. For example, our results indicate that existing Systems would be unable to search the entire IMDb database, which underscores the need for a progression of datasets that will allow researchers to make progress toward this objective. Creating a subset of the original dataset is common, but we are not aware of any work that identifies how to determine if a subset is representative of the original dataset. In addition, different research groups often have different schemas for the same data (e.g., IMDb), but the effect of different database schemas on experimental results has also not been studied.

XI. ACKNOWLEDGMENT

I have taken efforts in this projects .However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them. I am highly indebted to Dr. suresh L ,principal ,CITECH, Banglore .for their guidance and constants supervision as well as for providing necessary information regarding the project and also for their support in completing the project. I would like to express my gratitude towards my husband and my parents.

REFERENCES

[1] Baid, I. Rae, J. Li, A. Doan, and J. Naughton, “Toward Scalable Keyword Search over Relational Data,” Proceedings of the VLDB Endowment, vol. 3, no. 1, pp. 140–149, 2010.

[2] Bhalotia, A. Hulgeri, C. Nakhe, S. Chakrabarti, and S. Sudarshan,“Keyword Searching and Browsing in Databases using BANKS,” in Proceedings of the 18th International Conference on Data Engineering, ser. ICDE ’02, February 2002, pp. 431–440.

[3] http://blogs.msdn.com/b/craigfr/archive/2008/03/31/ranking-functions-rank-dense-rank-and-ntile.aspx

[4] S. Chaudhuri and G. Das, “Keyword Querying and Ranking in Databases,”Proceedings of the VLDB Endowment, vol. 2, pp. 1658–1659, August 2009..

[5] Y. Chen, W. Wang, Z. Liu, and X. Lin, “Keyword Search on Structured and Semi-Structured Data,” in Proceedings of the 35th SIGMOD International Conference on Management of Data, ser. SIGMOD ’09, June 2009, pp. 1005–1010.

[6] J. Coffman and A. C. Weaver, “A Framework for Evaluating Database Keyword Search Strategies,” in Proceedings of the 19th ACM International Conference on Information and Knowledge Management, ser. CIKM ’10, October 2010, pp. 729–738.

[7] B. B. Dalvi, M. Kshirsagar, and S. Sudarshan, “Keyword Search on External Memory Data Graphs,” Proceedings of the VLDB Endowment, vol. 1, no. 1, pp. 1189–1204, 2008.[7] E. W. Dijkstra, “A Note on Two Problems in Connexion with Graphs,” Numerische Mathematik, vol. 1, no. 1, pp. 269–271, 1959.

[8] K. Cios, W. Pedrycz, R. Swiniarski, L. Kurgan, Data Mining: A Knowledge Discovery Approach, Springer, ISBN: 978-0-387-33333-5, 2007.

[9] Margaret Dunham, Data Mining Introductory and Advanced Topics, ISBN: 0130888923, Prentice Hall, 2003.

[10] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy, editors, Advances in Knowledge Discovery and Data Mining, AAAI/MIT Press, 1996.

[11] Jiawei Han, Micheline Kamber, Data Mining : Concepts and Techniques, 2nd edition, Morgan Kaufmann, ISBN 1558609016, 2006.

[12] https://msdn.microsoft.com/en-us/library/dd172118(v%3Dvs.100).aspx&rct=j&frm=1&q=&esrc=s&sa=U&ei=W7n7VLyaBsWyuATB_ID4CA&ved=0CBoQFjAB& sig2=ABjehpLQPiQ07BXnwoOBQ&usg=AFQjCNFAfGvU_GSvBPkn--_wfgCXX9woQQ

[13] Keyword search over relational databases: a metadata approach.

[14] N. L. Sarda and Ankur Jain. Mragyati: A system for keyword-based searching in databases.

[15] Hinrich Sch¨utze, David A. Hull, and Jan O. Pedersen. A comparison of classifiers and document representations for the routing problem. In ACM SIGIR'95, (Special Issue of the SIGIR Forum), pages 229–237,1995.

[16] John C. Shafer and Rakesh Agrawal. Continuous querying in database-centric web applications. WWW9/Computer Networks, 33(1-6):519–531, 2000.

[17] Ron Weiss, Bienvenido V´elez, Mark A. Sheldon, Chanathip Nemprempre, Peter Szilagyi, Andrzej Duda and David K. Gifford. Hypursuit: A hierarchical network search engine that exploits content-link hypertext clustering. In Proc. of ACM Hypertext, pages 180–193, 1996.

[18] UteMasermann and Gottfried Vossen. Design and Implementation of a novel approach to Keyword Searching in Relational Databases. In Current Issues in databases and information systems, pages 171–184,September 2000.

[19] Kevin D. Munroe and Yannis Papakonstantinou. BBQ: A visual interface for integrated browsing and querying of xml. In Visual Database Systems, May 2000.

[20] Eran Palmon. Associative search method for heterogeneous databases with an integration mechanism configured to combine schema-free data models such as a hyperbase. United States Patent Number 5,740,421,Granted April 14, 1998, filed in 1995. Available at www.uspto.gov, 1998.

[21] Eran Palmon and Shai Geva. Associative search method with navigation for heterogeneous databases including an integration mechanism configured to combine schema-free data models such as a hyperbase. United States Patent Number 5,819,264, granted October 6, 1998, filed in 1995. Available at www.uspto.gov, 1998.