**Individual web revisitation by setting and substance importance input**

**Abstract**

A pertinence paramount input mechanism is additionally involved to tailor to individual’s recollection vigor and revisitation habits. Our dynamic management of setting and substance recollections including decay and reinforcement strategy can mimic users’ retrieval and recall mechanism. Acquire back to aforetime viewed web pages is a mundane yet uneasy task for users due to the immensely colossal volume of oneself accessed information on the web. This project purchase human’s natural recollect process of utilizing episodic and semantic recollection cues to smooth recall, and give an Individual web revisitation Process known as “Web Page Preview” through setting and substance users search keywords. Unexpressed methods for setting and substance user memories’ accession, storage, decay, and utilization for page re-discovery are discussed. Among time, location, and activity context factors in Web Page Prep, activity is the best recall cue, and context + content predicated re-discovery distributes the best performance, compared to context predicated re-discovery and content predicated re-discovery.

**Present System:**

From the Existing information, a number of techniques and implements like bookmarks, history implements, search engines, etc. systems have been developed to fortify Individual web revisitation. In subsisting search engine, and fetched germane antecedently viewed results from its cache. The incipiently available results were then merged with the aforetime viewed results to engender a list that fortified intuitive re-discovery and contained incipient information.

**For Ex:**

**History Implements**

History implements of web browsers preserve user’s accessed URLs chronologically according to visit time (e.g., present-day, before day, before week, etc.), and accessed page denominations and contents.

**Search Engines**

According to what search engines are utilized for re-discovery aforetime found search results. It explored the distinctions between queries that had substantial/minimal changes between the anterior query and the revisit query.

**Disadvantages:**

**•** No search for web revisitation.

• Depends on only time and date.

**Suggest System:**

Motivated by the theoretical discoverys, this paper explores how to leverage our natural recollect process of utilizing episodic and semantic recollection cues to facilitate Individual web revisitation. We present an Individual web revisitation technique, called “Web Page Preview” that sanctions end users to get back to their anteriorly concentrated pages through access context and page content end users search keywords. Unexpressed techniques for setting and substance memories’ accession, storage, and utilization for web page recollect are discussed. Preparation for web revisitation. When a utilizer accesses a web page, which is of potential to be revisited in future by the utilizer (i.e., page used time is overall by the threshold), the context accession and administration module captures the Present access context (from the currently running computer programs our application pic time duration, geo location, doing activities inferred) into a possibility context tree.

**Advantages:**

**•** New technology for Individual web revisitation.

• Depend on both setting and substance users search keywords**.**

Project System Requisites

Project Hardware Requisites

Hardware : Pentium

Speed : 1.1 GHz

RAM : 1GB

Hard Disk : 20 GB

Min Software requisites For Project Development

Operating System : Windows Family

Technology : Java and J2EE

Web Technologies : Html, JavaScript, CSS

Web Server : Apache Tomcat 7.0/8.0

Database : Free Download My SQL 5.5 or Higher

UML's : Star UML

Java Version : Open Source JDK 1.7 or 1.8

**1. Introduction to Project**

Present Days, the web is playing a paramount role in distributing information to users’ accessibility. A web page can be geo localized by a fine-tuned URL’s, & exhibits the search page content as used time-varying Screenshot. Among the mundane web deportments, web revisitation is to re-find the anteriorly viewed web pages, not only the page URL’s, but additionally the page Screenshot at that users access timestamp [1]. A six-week utilizer study with twenty three participants showed proximately fifty eight of web users belonged to web revisitation [2]. Need one more year study 45 % of Questions by min of 114 Research members were re-discovery sent requests [3]. According to [4], by and large, consistently page stacked was at that point went by afore by the same utilizer, and the ratio of revisited pages among all visits ranges between twenty percentages and seventeen two percentages .theoretical studies show that humans rely on both episodic recollection and semantic recollection to recall information or events from the past. Human’s episodic recollection receives and stores terrestrial dated episodes or events, together with their spatial-temporal cognations, while human’s semantic recollection, then again, is an organized record of realities, denouements, concepts and skills that one has acquired from the external world. Semantic statistics is gotten from aggregated long winded recollection. Episodic recollection can be thought of as a “map” that ties together things in semantic recollection. The two getting back make up the classification of human user’s declarative recollection, and collaborate in user’s information recollecting activities [5]. Thus, when a user’s web revisitation deportment transpires, he/she inclines to utilize episodic recollection, interweaved with semantic recollection, to get back the the anteriorly concentrated pages. Here, semantic recollection accommodates content information of aforetime concentrated pages, and episodic recollection keeps these pages’ access context “e.g., time used, location accessed, concurrent activities done, etc.” [6], [7]. Motivated by the theoretical discoveries, this Project explores how to leverage our natural recollect process of utilizing episodic and semantic recollection cues to facilitate Individual web revisitation. Considering the dissimilarity of users in memorizing anterior access context and page content cues, a pertinence paramountcy input contraption is involved to enhance Individual web revisitation production**.**

**2. Littérateur survey on our Project**

**Survey on “Improving web page revisitation: analysis, design and evaluation” (A. Cockburn, S. Greenberg, S. Jones, B. Mckenzie, and M. Moyle.)**

Utilizer deportment. We commence by reporting the recent results of a web use log-analysis, which show that revisiting pages is an ascendant activity on the web. System and utilizer models of the current demeanor of the Back button. Authors describe a facile to reiterate experiment to demonstrate that many users misunderstand the rudimentary deportment of the main interface implement for revisitation the Back button. We additionally discuss why, despite this misunderstanding, Back is heavily utilized. Ameliorating the effectiveness of the Back command.[1] authors describe the efficiency circumscriptions of the interface components used to issue the Back summon, and present an assessment of a gesture-predicated shortcut (akin to the plan as of late gave in the Opera web program). Amending understanding and effectiveness of the Back model. Authors describe an alternative ‘temporal’ comportment for the Back button and Forward button, and present the results of its evaluation. Ameliorating the presentation of revisitation implements.[1] In discussing the implicative insinuations of the earlier discoverys, we demonstrate how next-generation web browsers could integrate and enhance the diverse implements for revisitation that are accessible in current browsers (history lists ,Back and Forward and bookmarks). This work is perpetual, and preliminary Reports are emboldening

**Survey on “How people revisit web pages: empirical discoverys and implicative insinuations for the design of history systems “(L. Tauscher and S. Greenberg.)**

In paper authors provides empirical data that justifies the desideratum for congruous history mechanisms in graphical web browsers. Our analysis of different designs vigorously suggests that the predictability of URLs presented by current stack-predicated models of history can be ameliorated upon. Utilizing the methodology and design guidelines herein designers can validate and refine current history mechanisms and investigate incipient approaches. There are still many questions. With no answers [2] we require to evaluate the physical and cognitive effort for reviewing a particular conditioned set of history list presages. We require to validate the design guidelines that we have suggest.[2] We withal need to assess how utilization patterns change along with future browser interfaces (such as redesigned history mechanisms) and HTML extensions by using java frames . We do suspect that some of the numbers reported here would not transmute dramatically, such as the high recurrence rate and the recency effect.[2] In contrast, the numbers reflecting low utilization of history tool should transmute (hopefully) as the browser interface is ameliorated.

**Survey on “Memento: amalgamating content and context to avail webpage re-visitation” (C. E. Kulkarni, S. Raju, and R. Udupa)**

Our current system can facilely be modified to assign different weights to the page-content and page-context. Authors plan to conduct more detailed studies to understand when content is more consequential, and congruously assign weights in our algorithm. Authors plan to integrate Memento into subsisting browsers. [8] This will additionally avail us evaluate how Memento performs in conjunction with other re-visitation support like exact match search for browsing history While clients frequently return to pages on the Web, implement bolster for such re-appearance is still destitute. [8] Current implements for browser histories only provide users with rudimental data, for example, the date of the last visit and assignment of the page went to. [8] In this paper, the authors portray a framework that furnishes clients with graphic theme expresses that avail re-discovery. [8] Not at all like earlier work, our framework considers both the substance of a site page and the setting in which the page was gone by user. Preliminary the authors suggest in this system suggests users find this approach of coalescing content with context subsidiary.

**Survey On: Ameliorating revisitation browsers capability by utilizing a dynamic bookmarks personal toolbar” (J. A. Gamez, J. L. Mateo, and J. M. Puerta.)**

In this paper authors present an incipient approach to integrate perspicacity to Internet browsers utilizer interface. Our contribution is predicated on amending browsers revisitation capabilities by learning a model from user’s navigation deportment, that later is utilized to soothsay a set of bookmarks liable to be used next. This group of bookmarks must be a list of moderate size less than or equal to ten because our goal is to show them in the browser personal toolbar called as bookmarks. We cerebrate that dealing with this component of the utilizer interface is propitious for revisitation the contrary to history or bookmarks list tree it is always visible and on the utilizer can access the desired web page by utilizing a single mouse click. In this work we fixate on performing the comparison of several (computationally) simple types in order to identify a good candidate to be utilized as utilizer navigation model. From the experiments carried out we identify that an amalgamation of Verdant Bayes with One R could be a good cull.[5] However, web’s expedited growing has turned it into an astronomical amount of information poorly structured, from which retrieving germane information is often a quite laborious and intricate task even for experienced users. Because of this, once a site page has been found and judged as intriguing by a given utilizer, it is more facile for her to remember its deliver with a specific end goal to return to it, than having to probe for it again later on (agreeing to [1] 60% of the pages a person visually perceives are revisits). [4] Current programs join standard implements for web pages revisitation. Apart from back/forward buttons we can distinguish

**Survey On: “Beyond the conventional suspects: context-vigilant revisitation support” (R. Kawase, G. Papadakis, E. Herder, and W. Nejdl.)**

A lot of our exercises on the Web include returns to pages or destinations. Explanations behind returning to incorporate dynamic checking of substance, confirmation of data,, conventional utilization of online accommodations, and reoccurring tasks. Browsers support for revisitation is mainly fixated on frequently and recently visited pages. [5] In this paper authors Propos a recommendations using browser toolbar as it dynamic beyond these customary suspects, balancing diversity and pertinence. The recommendation method used is a coalescence of ranking and propagation methods. Experimental results demonstrate that this calculation performs fundamentally superior to the benchmark method.[5] Additionally analyzes address the inquiry whether it is more felicitous to recommend categorical pages or rather (portal pages of) Web sites. Authors conducted two utilizer ponders with a dynamic toolbar that depends on our suggestion calculation. In this context, the outcomes corroborate that users appreciate and utilize the logical proposals gave by the toolbar. [5] In this paper, authors introduced a generic framework for contextual presage of revisits. The framework consists of two tiers of methods: which rank resources predicated as ranking method, on the regency and/or frequency of access to this resource, also, spread strategies, which distinguish things that are normally gone to together with the at present went to resource. Experimental evaluation shows that coalescing ranking methods with propagation ones drastically ameliorates performance.[5] In a second experiment, we found that site presage is less perplexed than page prognostication, and that the performance of a presage strategy mainly depends on variance in the users’ online comportment (in particular, the page and site entropy). The best-performing prognostication strategy has been incorporated with regards to a dynamic program toolbar, the PivotBar. Two utilizer studies with the PivotBar corroborate that users appreciate and utilize the logical proposals gave by the toolbar. In integration, the log data shows that a consequential amount of revisits has taken place via the PivotBar.

**3. System analysis**

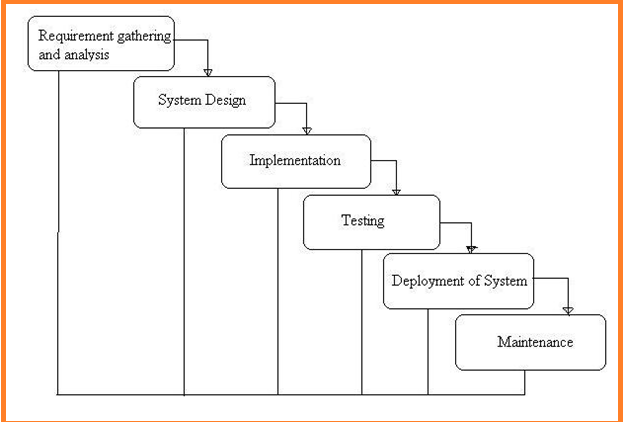


Fig:-1 Project SDLC

• Project Requisites Accumulating and Analysis

• Application System Design

• Practical Implementation

• Manual Testing of My Application

• Application Deployment of System

• Maintenance of the Project

**Requisites Accumulating and Analysis**

It’s the first and foremost stage of the any project as our is a an academic leave for requisites amassing we followed of IEEE Journals and Amassed so many IEEE Relegated papers and final culled a Paper designated “Individual web revisitation by setting and substance importance input and for analysis stage we took referees from the paper and did literature survey of some papers and amassed all the Requisites of the project in this stage

**System Design**

In System Design has divided into three types like GUI Designing, UML Designing with avails in development of project in facile way with different actor and its utilizer case by utilizer case diagram, flow of the project utilizing sequence, Class diagram gives information about different class in the project with methods that have to be utilized in the project if comes to our project our UML Will utilizable in this way The third and post import for the project in system design is Data base design where we endeavor to design data base predicated on the number of modules in our project

**Implementation**

The Implementation is Phase where we endeavor to give the practical output of the work done in designing stage and most of Coding in Business logic lay coms into action in this stage its main and crucial part of the project

**Testing**

**Unit Testing**

It is done by the developer itself in every stage of the project and fine-tuning the bug and module predicated additionally done by the developer only here we are going to solve all the runtime errors

**Manual Testing**

As our Project is academic Leave we can do any automatic testing so we follow manual testing by endeavor and error methods

**Deployment of System**

Once the project is total yare we will come to deployment of client system in genuinely world as its academic leave we did deployment i our college lab only with all need Software’s with having Windows OS

**Maintenance**

The Maintenance of our Project is one time process only

**Functional Requirements**

* **User Sign up :-** all the application users have to give all the mandatory Fields and get an Account in our application to access our application
* **User Login:-** To access the application we are verifying the users login user name and Password
* **Admin Login:-** he is super user of the application where he can login into the application with his/her user name and password
* **Admin Add Activity:-** Our super user is going to add all the Activity in to application where is saves in our data base
* **Result Performance:-** we can see the performance of the applicant in form of Graph
* **Search:-** user will search the result as per the requirements
* **History:-** Show the our applicant
* **Activities :-** users Search Activities will be displayed in
* **Revisitation:-** so the Revisitation Search results of each user

**Application needs Non-Functional Requisites**

**Expanded System admin security:** overseer to eschew the abuse of the application by PC ought to be exceptionally secured and available.

**Compactness:** The Presentation of this application is facile to utilize so it is looks simple for the using client to comprehend and react to identically tantamount.

**Unwavering quality:** and the functionalities accessible in the application this substructure has high probability to convey us the required inquiries.

**Time take for Reaction:** The time taken by the application to culminate an undertaking given by the client is very fast.

**Multifariousness:** Our application can be stretched out to incorporate the vicissitudes done by applications present now to enhance the performance of the item. This is implicatively insinuated for the future works that will be done on the application.

**Vigor:** The project is blame tolerant concerning illicit client/beneficiary sources of info. Blunder checking has been worked in the platforms to avert platforms disappointment.

**4. Implementation**

**Modules**

**Admin**

Admin is super utilizer for application where he wills do some operations like integrate context to the server show performance of the discovery rate, average precision, average recall.

**Search Utilizer**

The search utilizer search for the information need to him/her predicated on the different users search keywords predicated on the requisites different activity revisitation and Web revisitation

**Authentication and sanction**

Authentication and sanction is the function of designating access privileges to our application by utilizer getting Registered with all the desideratum fields for our application security and security in general and to access control by authenticate in to the application by his/her utilizer name & password in particular More formally to sanction that utilizer in our application for non-registered utilizer application will define an access policy

**Composition for application web revisitation**

By any a utilizer accesses a web page, which is of potential to be revisited later by the utilizer “threshold value of page access time”, the context accession and management module captures the current access context “from the currently running computer programs time of access, gio location, activities done inferred” into a probabilistic context tree. Meantime, the content extraction and management module performs the unigram based extraction from the exhibited page division and obtains a list of probabilistic content terms. The likelihood of acquired context instances and extracted content terms reflect how likely the utilizer will refer to them as recollection cues to get back to the aforetime concentrated page.

**Application Web revisitation**

In future when a utilizer requests to need back to an aforetime concentrated page through context and/or content users search keywords, the re-entrance by context users search keywords module and re-entrance by content users search keywords module probe “the probabilistic context tree repository and probabilistic term list repository”, respectively. The result generation and consequentiality input adjustment module cumulates the two search results and returns to the utilizer a ranked list of all visited page URLs. The pertinence paramountcy input process dynamically tunes influential parameters (counting

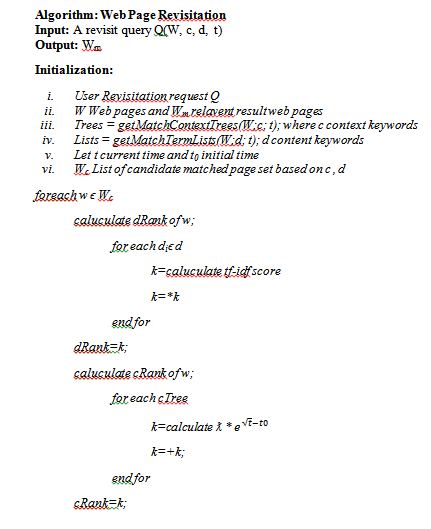
End used memories’ rates, threshold value time spent on page, threshold value of size of window used , weight vectors in computing the sodality and scores of impression), which are critical to the construction and management of setting and substance recollections for every Individual web revisitation.

**Application Web Page Preview**

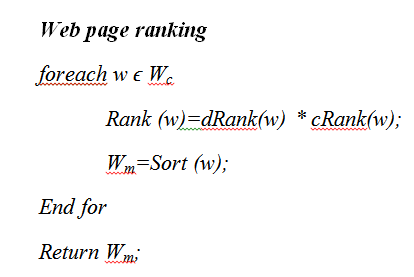
That sanctions users to get back to their aforetime concentrated pages through access context and page content users search keywords. Unexpressed techniques for setting and substance memories’ accession, storage, and utilization for web page recall are discussed.

**Algorithm used in Project**

**Web Page Revisitation**

******

**Web page ranking**

****

**Architecture diagram**

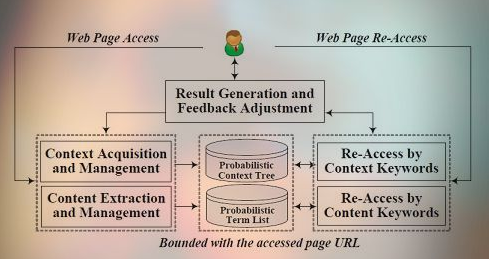


Fig:-2 Project Architecture diagram

**About Project Software’s**

JAVA, Apache Server, MSQL, EDIT ++

In our web Application Development we are using one tier architecture as total applicant will be developed in single system with all the three layers of application development like presentation layer where we use our web technologies to make of GUI of the application like HTML, HTML-5, CSS, JS Etc. and in second layer we have to make our business logic or called as implementation of application where we are using java, J2EE and also we use JDBC to connect from our Business layer to data base layer and final our data base layer where we develop the Data structure of the application

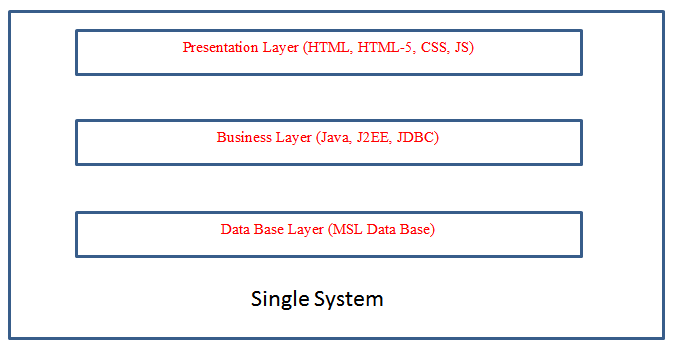


Fig:-3 Single tire Architecture Project Development

**How we used java in our Project Development**

**Installation and Setup in our system**

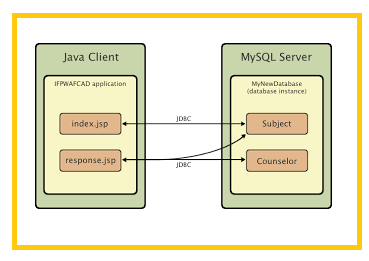
The Software we download form the oracle website as it’s an open source as per the software we have installed it in our system and for we have set the system path of java in our OS location We have used the main logic of our algorithm by core java concepts only for web application we have used all JSP concepts and to connect data base we have used JDBC with all this concept we have done the application in Single tire Architecture Project

**Data Storage in MYSQL**

We have taken open source software MYSQL from the provide website and run in our system we used for creating our project data base related tables as per project requirement’s even for user friendly access of my sql we used Software called SQL Yog where we can do all the operation of mysql by click & use

**About the role of apace tomcat webserver**

As our project is a web applicant we need webserver so for that we used again open sour software where our total project source code will be in webapps of the server form that location the application run into web browser where users can see the implementation of the total project



Application Development Structure

**Sample Code**

**Revisitation -1**

<%

String content=request.getParameter("content");

String context=request.getParameter("context");

session.setAttribute("content",content);

session.setAttribute("context",context);

endeavor{

Connection con = databasecon.getconnection();

Verbal expression st=con.createStatement();

String Sql="cull \* from context where seconds>30 and utilizer\_='"+session.getAttribute("email")+"' ";

String[] words=content.split("\\s+");

for(String w: words){

Sql=Sql+" and data\_ like '%"+w+"%' ";

}

System.out.println(Sql);

ResultSet rs=st.executeQuery(Sql);

%>

<div class="grid\_3 grid\_5 wow fadeInUp animated" data-wow-delay=".5s">

<%

int count=0;

String temp="";

while(rs.next())

{%>

<h3>

<span class="label label-default"><%=++count%></span>

<span class="label label-admonishment"><%

temp=rs.getString("denomination\_");

try{out.println(temp.substring(0,37)+"..");

}catch(Exception e){

out.println(temp);

}

%>

</span>

<span class="label label-primary"><%temp=rs.getString("uri");

try{out.println(temp.substring(0,50)+"..");

}catch(Exception e){

out.println(temp);

}

%></span>

<span class="label label-prosperity"><%=rs.getString("date\_")%></span>

<span class="label label-info"><%=rs.getString("seconds")%> Seconds</span>

</h3>

<%

}

rs=st.executeQuery("cull count(\*) from context where seconds>30 and utilizer\_='"+session.getAttribute("email")+"' ");

rs.next();

session.setAttribute("toturl",rs.getDouble(1));

session.setAttribute("matched",count);

}catch(Exception e){

System.out.println("Sec update exception: "+e);

}

%><br><br>

<div class="opinion">

<form method="post" action="revisitation2.jsp">

<h3><input type="submit" Value="Time Ratio"></h2>

</form>

</div>

**Revisitation -2**

<%

String content=(String)session.getAttribute("content");

System.out.println(content);

endeavor{

Connection con = databasecon.getconnection();

System.out.println(con);

Verbalization st=con.createStatement();

String Sql="cull \* from context where seconds>30 and utilizer\_='"+session.getAttribute("email")+"' ";

System.out.println(Sql);

String[] words=content.split("\\s+");

System.out.println(words);

for(String w: words){

Sql=Sql+" and data\_ like '%"+w+"%' ";

}

System.out.println(Sql);

System.out.println(st);

ResultSet rs=st.executeQuery(Sql);

System.out.println(rs);

%>

<table width="100%" border cellpadding="10">

<tr><th>URL ID<th>URL<th>Visitation time<th>Maximal time<th>Len(w,d)

<%

int count=0;

String temp="";

while(rs.next())

{%>

<tr><td><%=rs.getString(1)%><td><%=rs.getString(2)%><td><%=rs.getString("seconds")%> Seconds<td>500 Seconds<td><%=(rs.getDouble("seconds")/500)%>

<%

}

}catch(Exception e){

System.out.println("Sec update exception: "+e);

}

%>

</table>

<br><br>

<div class="opinion">

<form method="post" action="revisitation3.jsp">

<h3><input type="submit" Value="TF-IDF Caluculation"></h2>

</form>

</div>

**Process Java File**

public class Processer

{

public static Set<String> main()

{

Set<String> set=new HashSet<String>();

try {

String line;

String temp[]=null;

Process p = Runtime.getRuntime().exec

(System.getenv("windir") +"\\system32\\"+"tasklist.exe");

BufferedReader input = incipient BufferedReader(incipient InputStreamReader(p.getInputStream()));

while ((line = input.readLine()) != null) {

//System.out.println(line);

temp=line.split("\\s+");

System.out.println(temp[0]);

set.add(temp[0]);

}

input.close();

} catch (Exception err) {

err.printStackTrace();

}

return set;

}

public static void main(String[] args){

System.out.println(main());

}

}

**5. System Design**

**Class**



Fig:-4 Project Class

**Description:**

In PC code planning, a class plot inside the Unified Modeling Language we used in our Project Development is Star (UML) could be a sort of static structure outline that delineates the structure of a system by showing the system's characterizations, their qualities, operations (or methodologies), and moreover the associations among the groupings. It elucidates that class contains data.

Our Total Application on Use Case



Fig:-5 Project User Case

**Description:**

A usage case outline inside the Unified Modeling Language we used in our Project Development is Star (UML) could be a sort of behavioral chart portrayed out by and produced using a Use-case examination. Its inspiration is to gift a graphical layout of the presence of mind gave by a system to the extent performing specialists, their targets (addressed as use cases), and any conditions between those use cases. The most explanation behind a use case diagram is to show what structure limits are played out that on-screen character. Parts of the entertainers inside the system will be diagram.

**User Sequence:**



Fig:-6 Project User Sequence

**Description:**

A gathering diagram in Unified Modeling Language we used in our Project Development is Star (UML) could be a sensibly association graph that shows however frames work with each other and in what mastermind. It’s a create of a Message Sequence Chart. Progression diagrams are regularly known as event plots, event conditions, and short lived approach outlines.

**User Collaboration:**

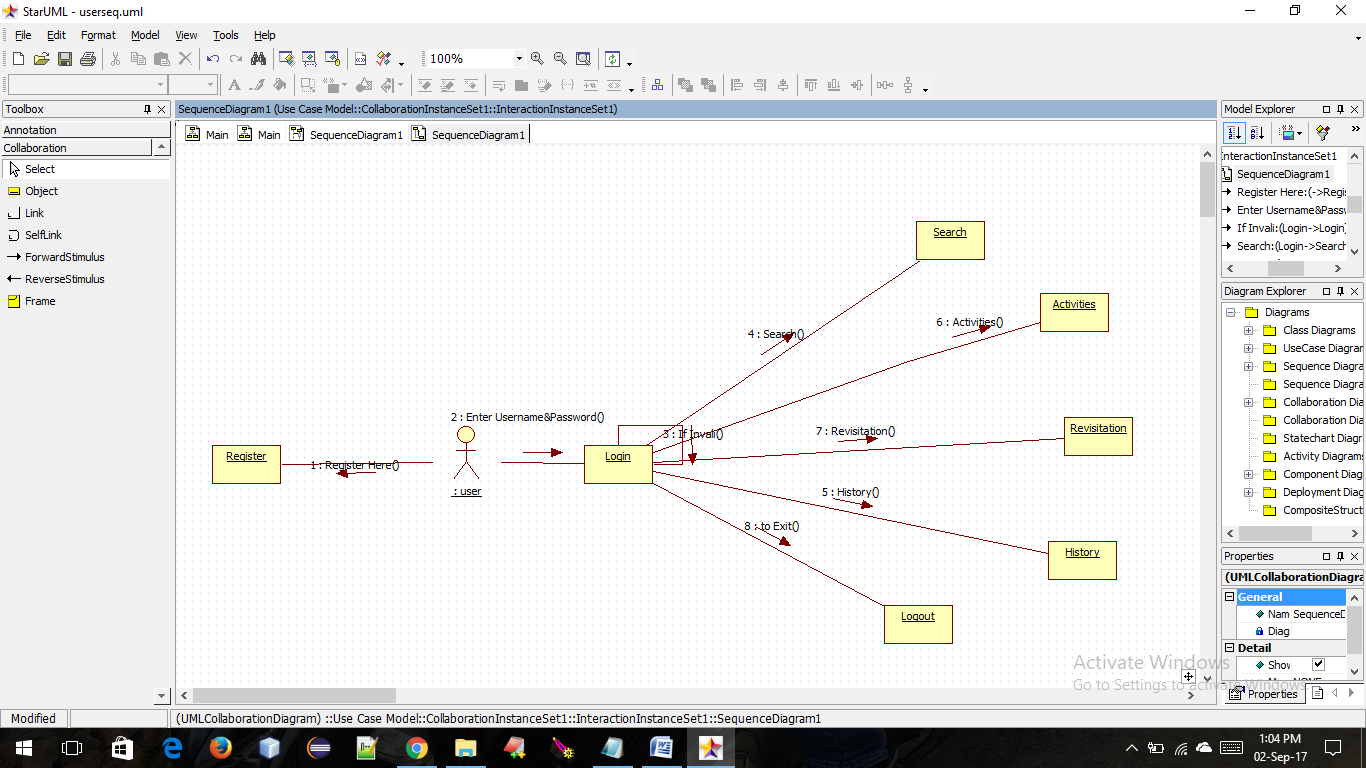


Fig:-6 Project User Collaboration

**Admin Sequence:**



Fig:-7 Admin Sequence

**Admin Collaboration:**

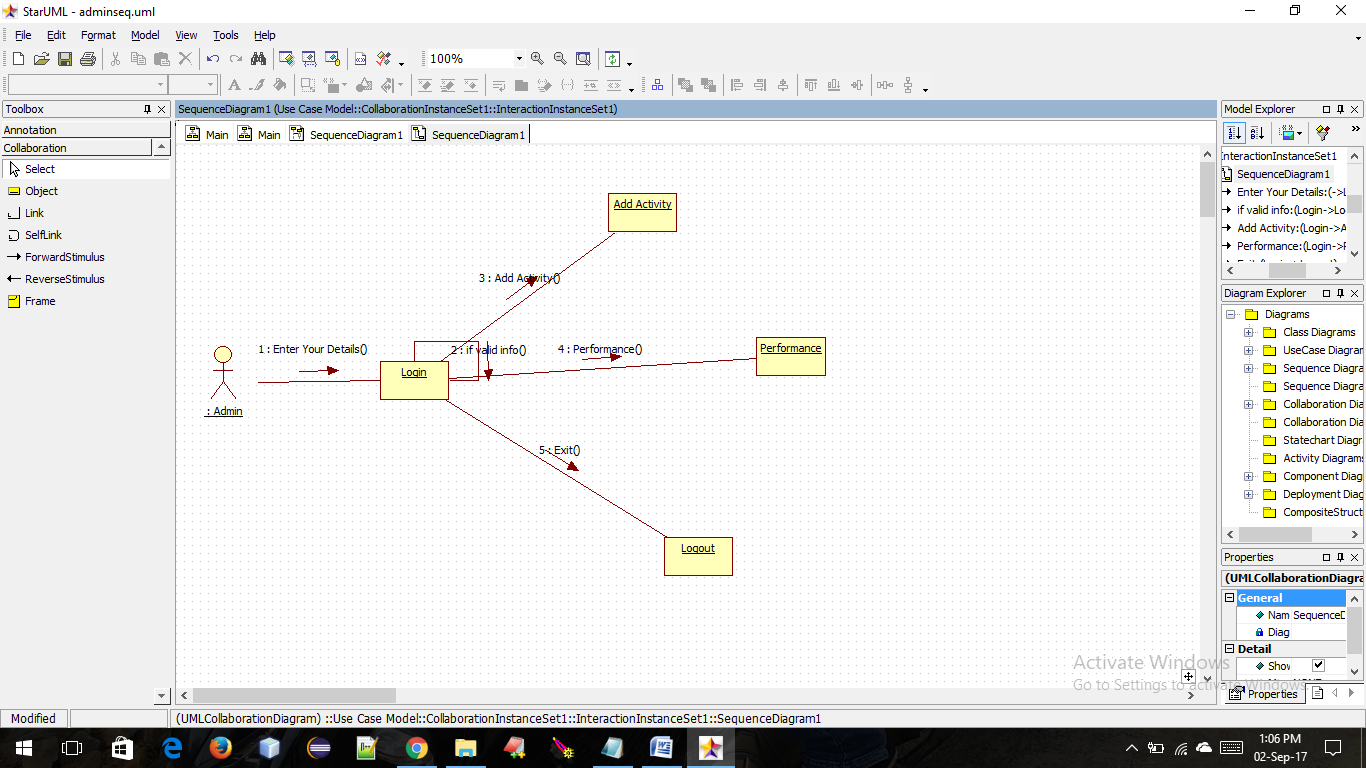


Fig:-8 Admin Collaboration

**State Chart**



Fig:-9 Project State Chart

**Description:**

state graph (state machine define or statechart chart) A state graph, likewise referred to as a state machine graph or statechart define, could be a illustration of the states an issue will accomplish and additionally the advances between those states within the Unified Modeling Language we used in our Project Development is Star (UML).

**Activity of Project**



Fig:-10 Project active

**Description:**

Activity outlines are graphical depictions of work procedures of stepwise activities and exercises with help for decision, cycle and synchronization. Inside the Unified Modeling Language we used in our Project Dvelpoemnt is Star , developments layouts will be regular depict the business and operational in little stages work procedures of parts in the midst of a structure. Relate in Nursing activity diagram shows the flood of organization.

**Component:**



Fig:-10 Project Component

**Description**

A part graph, otherwise referred to as AN UML phase chart, depicts the association and wiring of the physical division in a very framework. Phase graphs are oftentimes interested in facilitate demonstrate usage points of interest and twofold watch that every a part of the framework's needed capacities is secured by organized improvement. Within the main style of UML, elements incorporated into these charts were physical: archives, information table, documents, and executable, each physical element with a region. Within the realm of UML a pair of, these elements aren't most physical however rather a lot of calculated stay solitary define parts, for instance, a business procedure that provides or expects interfaces to speak with completely different develops within the framework.

**Deployment Diagram**

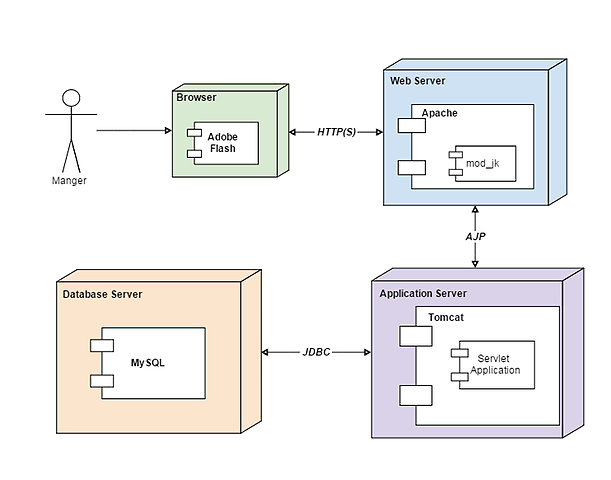
****

Fig:-11 Java Application Deployment Diagram

**Data Base design**

**E-R design**



Fig:-12 Project ER-Diagram

**Description:**

A substance relationship graph (ERD) may be a portrayal of knowledge within an area. It contains of components and conjointly connections between components. A component are often AN clear, physical protest, as an example, a faculty or understudy, or an idea, as an example or an exchange.

**DFD:**



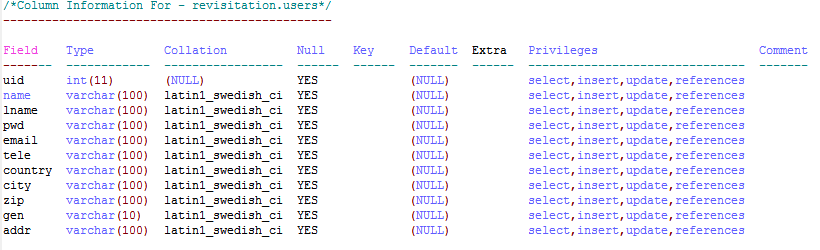
Fig:-12 Application Data Flow

**Description:**

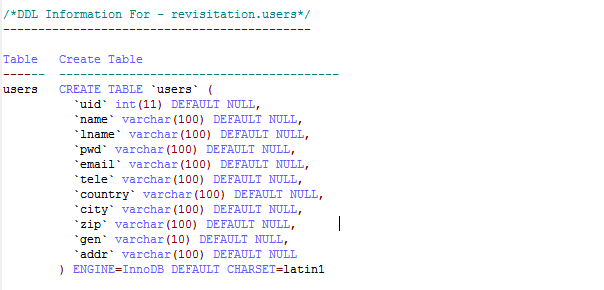
An info stream graph (DFD) maps out the stream of knowledge for any procedure or framework. It utilizes characterized pictures like rectangles, circles and bolts, additionally to short content marks, to point info inputs, yields, reposition focuses and therefore the courses between each goal. Info flowcharts will run from basic, even hand-drawn method outlines, to prime to bottom, multi-level DFDs that dive dynamically a lot of profound into however the data is taken care of.

**Data Base Tables**

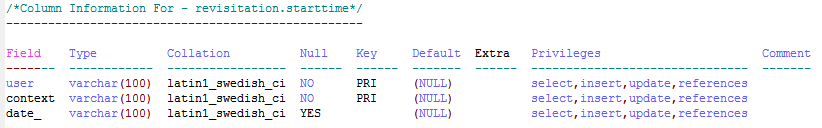
**User Registration Table**

****

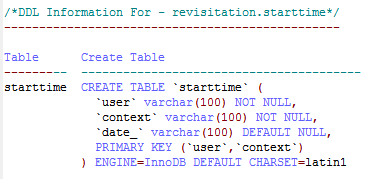
**SQL Query for Create User Table**

****

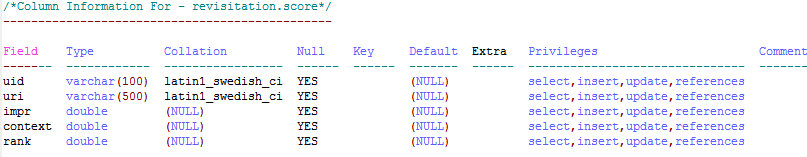
**Start Time Table**

****

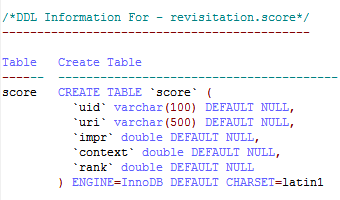
**SQL Query for Start Time Table**

****

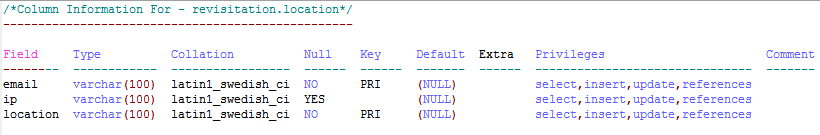
**Score Table**

****

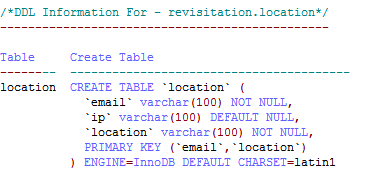
**SQL Query for Score table**

****

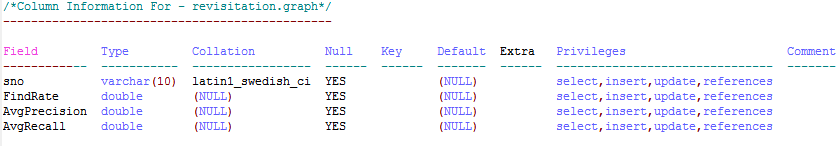
**Location Table**

****

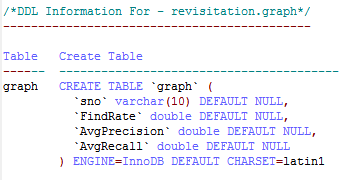
**SQL Query for Location table**

****

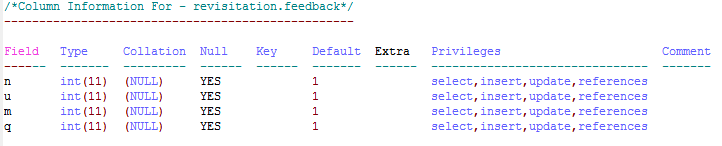
**Graph Table**

****

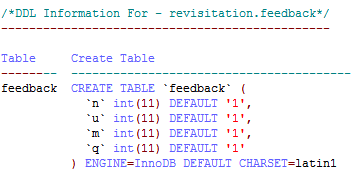
**SQL Query for Graph Table**

****

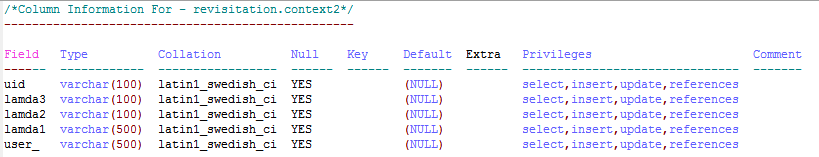
**Importance input Table**

****

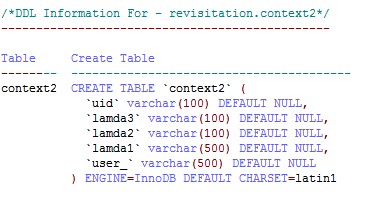
**SQL Query for Importance input Table**

****

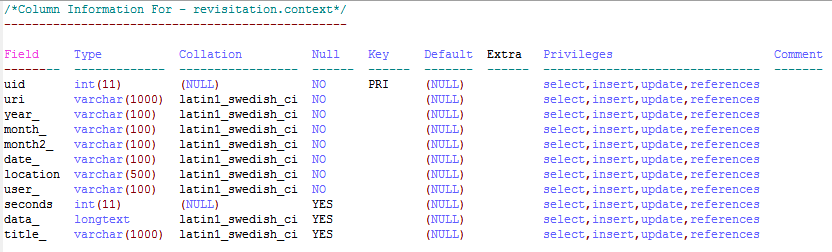
**Context-2 Table**

****

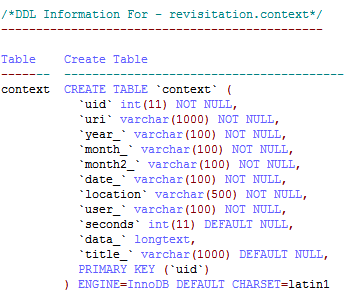
**SQL Query for Context-2 Table**

****

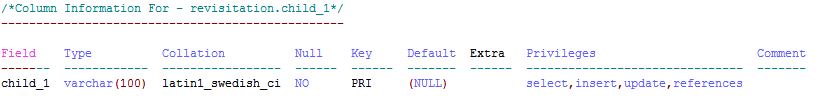
**Context Table**

****

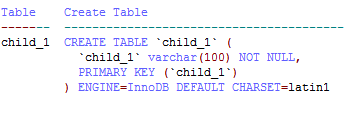
**SQL Query for Context Table**

****

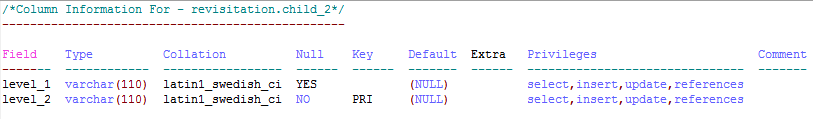
**Child-1 Table**

****

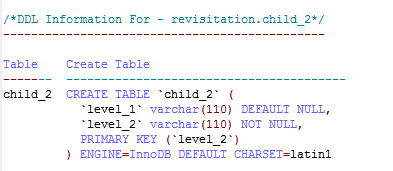
**SQL Query For Child-1 Table**

****

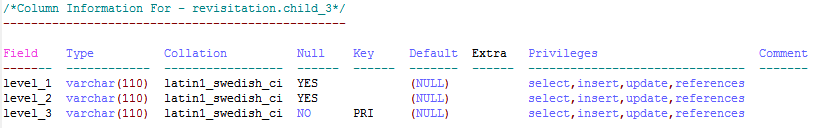
**Child-2 Table**

****

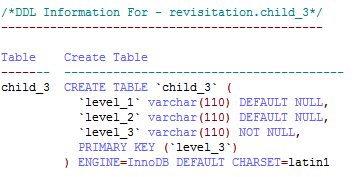
**SQL Query for Child-2 Table**

****

**Child-3 Table**

****

**SQL Query for Child-3 table**

****

**6. Software testing**

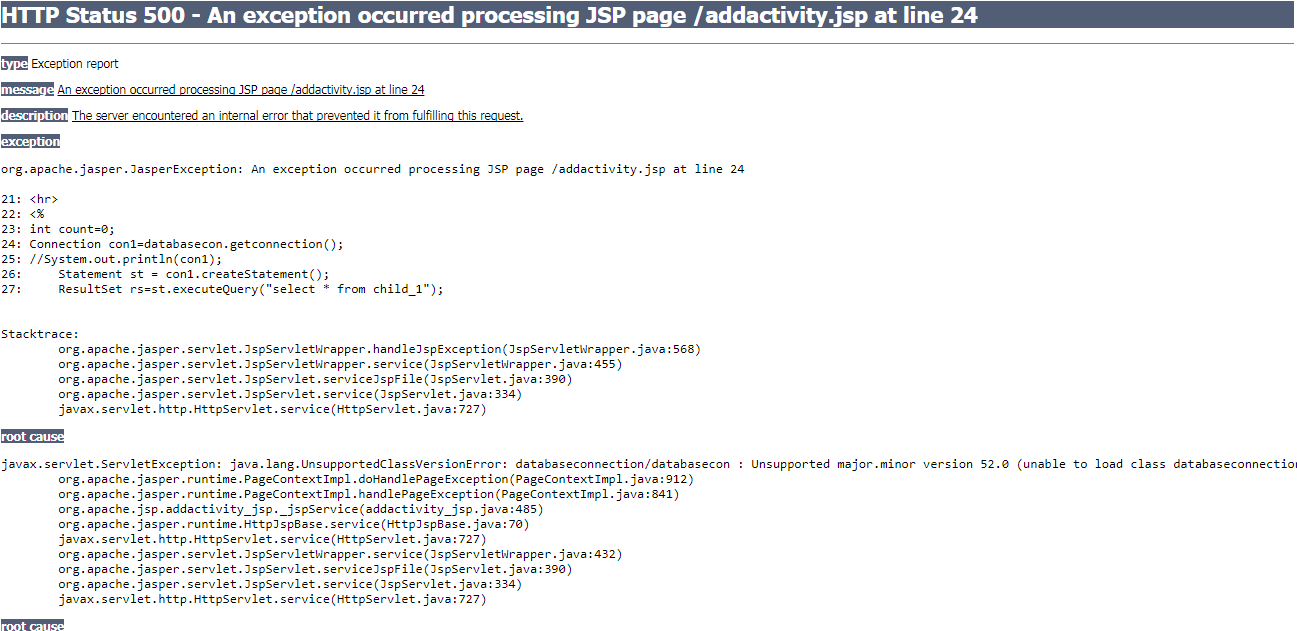
Software testing is one of the main stages of project development life cycle to provide our cessation utilizer with information about the quality of the application and ours, in our Project we have under gone some stages of testing like unit testing where it’s done in development stage of the project when we are in implementation of the application after the Project is yare we have done manual testing with different Case of all the different modules in the application we have even done browser compatibility testing in different web browsers in market, even we have done Client side validation testing on our application

**Unit testing**

The unit testing is done in the stage of implementation of the project only the error are solved in development stage some of the error we come across in development are given below

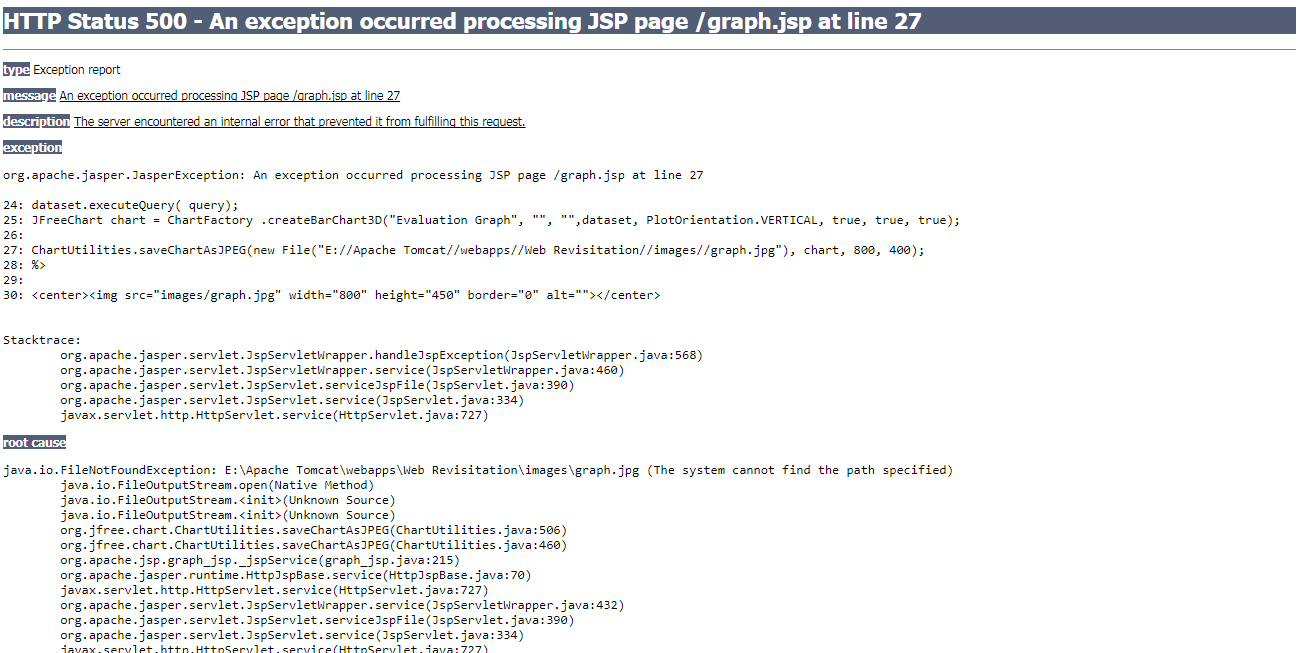
**Testing done when application is in development stage**

**Class version Error in our application**

****

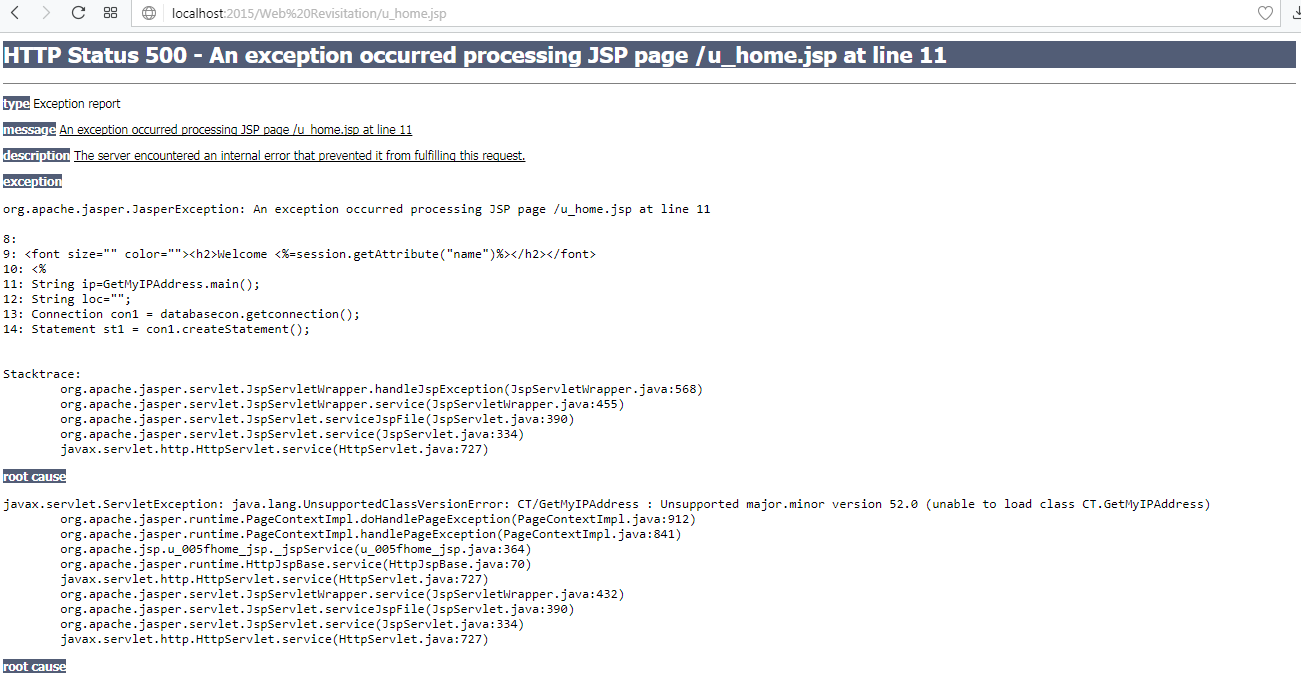
This Error Come when we move our application from one system to other and mainly when we version issues in the software’s we us

**Path related error in our application**

****

This Error Came when I have Performance Metrics to show in graph when I missed my server directly path in the system so we got this error in the applicant in development stage

**Server Connection Error**

****

**Manual testing on project application**

**TEST CASES**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** #1 | | **Test Case Description** - Validations in Registration Form | | | |
| **S#** | **Prerequisites** | | **S#** | **Test Data Requirement** | |
| 1 | User should be Registered | | 1 | Data should be valid | |
| **Test Condition** | | | | | |
| Entering data in registration form | | | | | |
| **Step #** | **Step Details** | **Expected Results** | | **Actual Results** | **Pass/Fail/Not Executed/Suspended** |
| 1 | User gives First and Last Name | Pop showing email verification message | | Enter valid email/password | Fail |
| 2 | Submitting the form without entering any details | Pop showing email verification message | | Enter email /password | Fail |
| 3 | User enters invalid format of email id | Pop showing email verification message | | Enter valid email id | Fail |
| 4 | User enters a phone number with < 10 digits | Pop showing email verification message | | Enter valid phone number | Fail |
| 5 | Entering valid username and password | Pop showing email verification message | | Pop showing email verification message | Pass |

Table 1 Registration test case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** #2 | | **Test Case Description** - Validations in Login Form | | | |
| **S#** | **Prerequisites** | | **S#** | **Test Data Requirement** | |
| 1 | User should have an email id | | 1 | Data should be valid | |
| **Test Condition** | | | | | |
| Entering data in login form | | | | | |
| **Step #** | **Step Details** | **Expected Results** | | **Actual Results** | **Pass/Fail/Not Executed/Suspended** |
| 1 | User gives a email or password of <6 characters | User logged in | | Enter valid email/password | Fail |
| 2 | Submitting the form without entering any details | User logged in | | Enter email /password | Fail |
| 3 | User enters wrong Email and (or) password | User logged in | | Enter correct email /password | Fail |

Table 2 Login test case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** #3 | | **Test Case Description** - Search box | | | |
| **S#** | **Prerequisites** | | **S#** | **Test Data Requirement** | |
| 1 | Any user | | 1 |  | |
| **Test Condition** | | | | | |
| Entering data in search box | | | | | |
| **Step #** | **Step Details** | **Expected Results** | | **Actual Results** | **Pass/Fail/Not Executed/Suspended** |
| 1 | User enters a value and clicks on search | Results in the form of url’s | | Nothing happens | Fail |

Table 3 Search test case

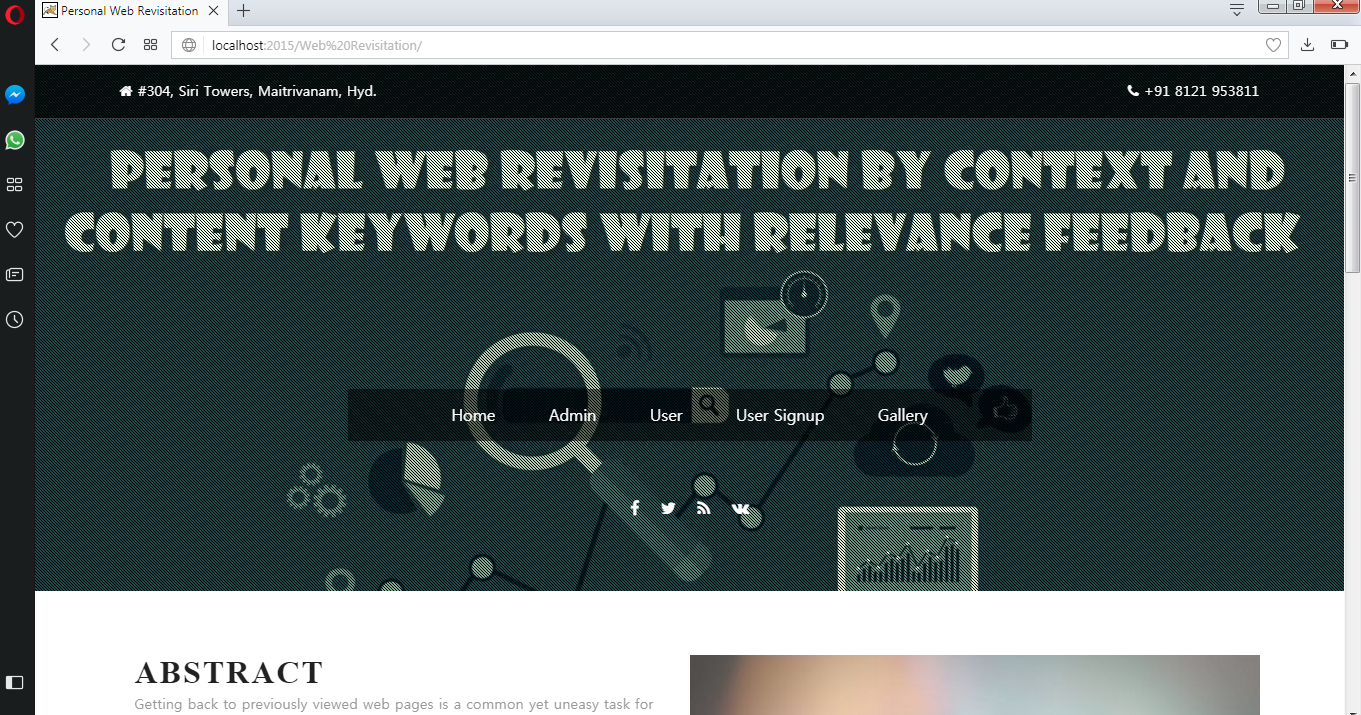
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** #4 | | **Test Case Description** - Testing Filters | | | |
| **S#** | **Prerequisites** | | **S#** | **Test Data Requirement** | |
| 1 | User | |  |  | |
| **Test Condition** | | | | | |
| Seed Sites | | | | | |
| **Step #** | **Step Details** | **Expected Results** | | **Actual Results** | **Pass/Fail/Not Executed/Suspended** |
| 1 | Click on history  Tab | Shows relevant history | | Get information | Pass |
| 2 | Click on Activities  Tab | Get some tabs here | | Enter Context Users search keywords | Pass |

Table 4 user modules

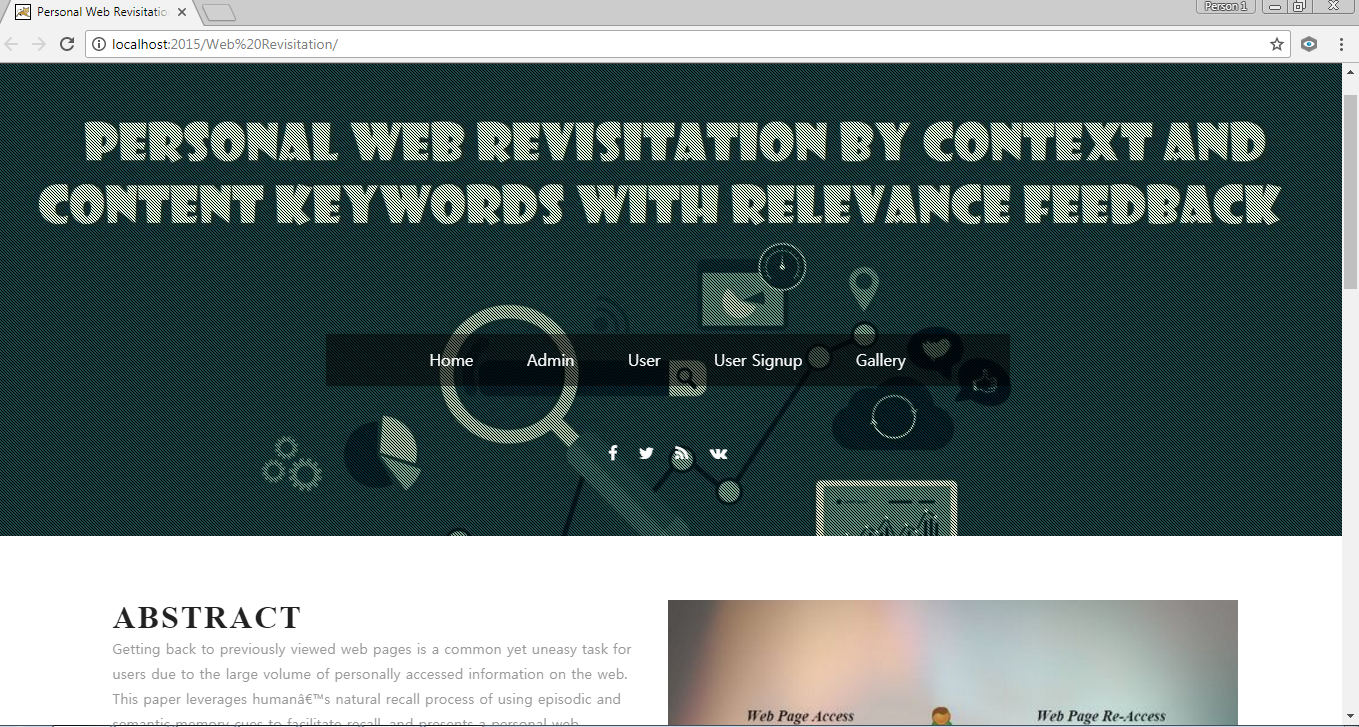
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** #5 | | **Test Case Description** - Update / Delete restaurant | | | |
| **S#** | **Prerequisites** | | **S#** | **Test Data Requirement** | |
| 1 | Logged in user | | 1 | Revisitation | |
| **Test Condition** | | | | | |
| Crawl data page | | | | | |
| **Step #** | **Step Details** | **Expected Results** | | **Actual Results** | **Pass/Fail/Not Executed/Suspended** |
| 1 | Revisitation | Enter Context Keyworsds | | Calculate | Fail |

Table 5 Revisitation

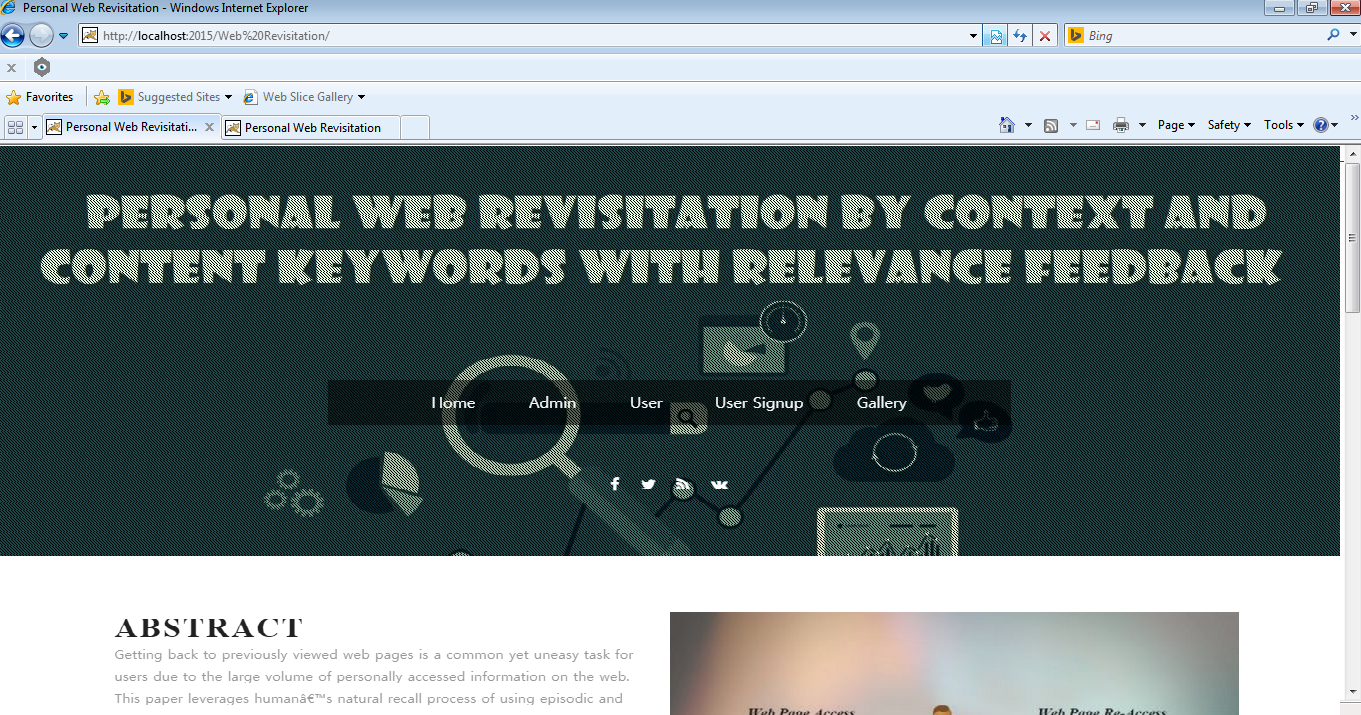
**Browser compatibility testing to project application**



**Result of My Application on opera browser**

****

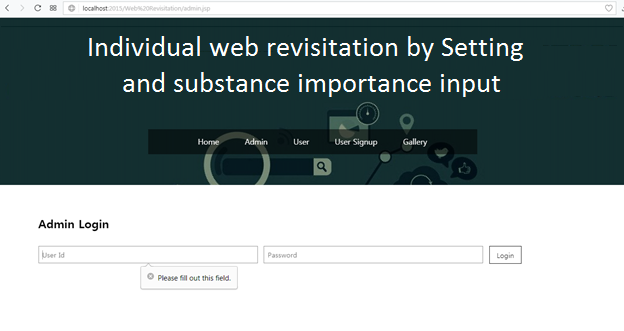
**Result of my Project in chrome**

****

**Result of my Project in Inter net Explore**

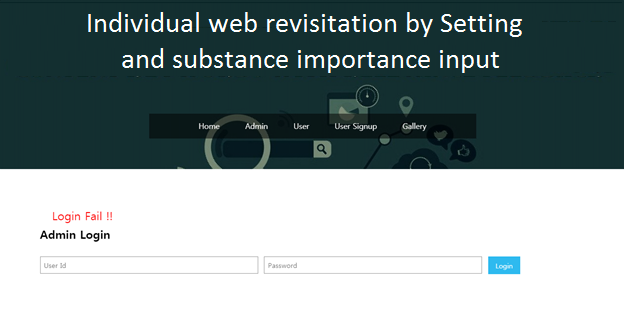
**Validation testing for project application**

**Admin Login with “NULL Value Validation”**

****

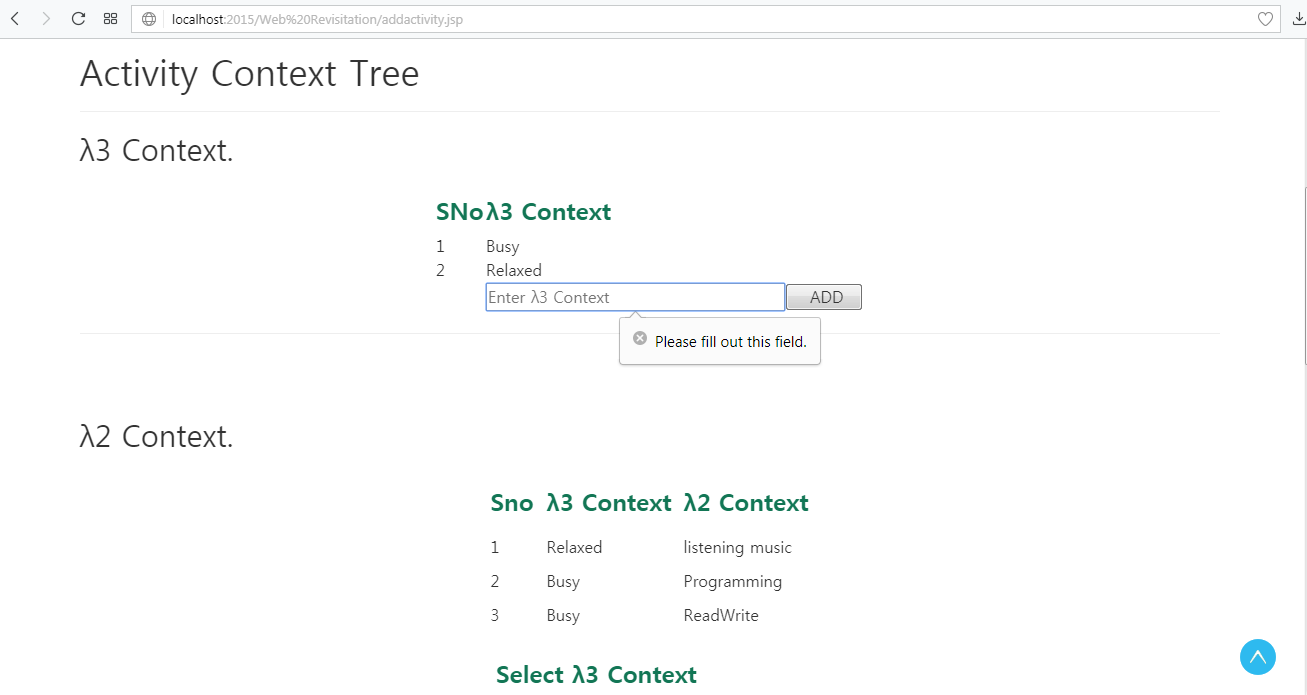
In admin login scree when we are trying to login with null values its giving a error message like please fill out the field

**Admin invalid credentials**

****

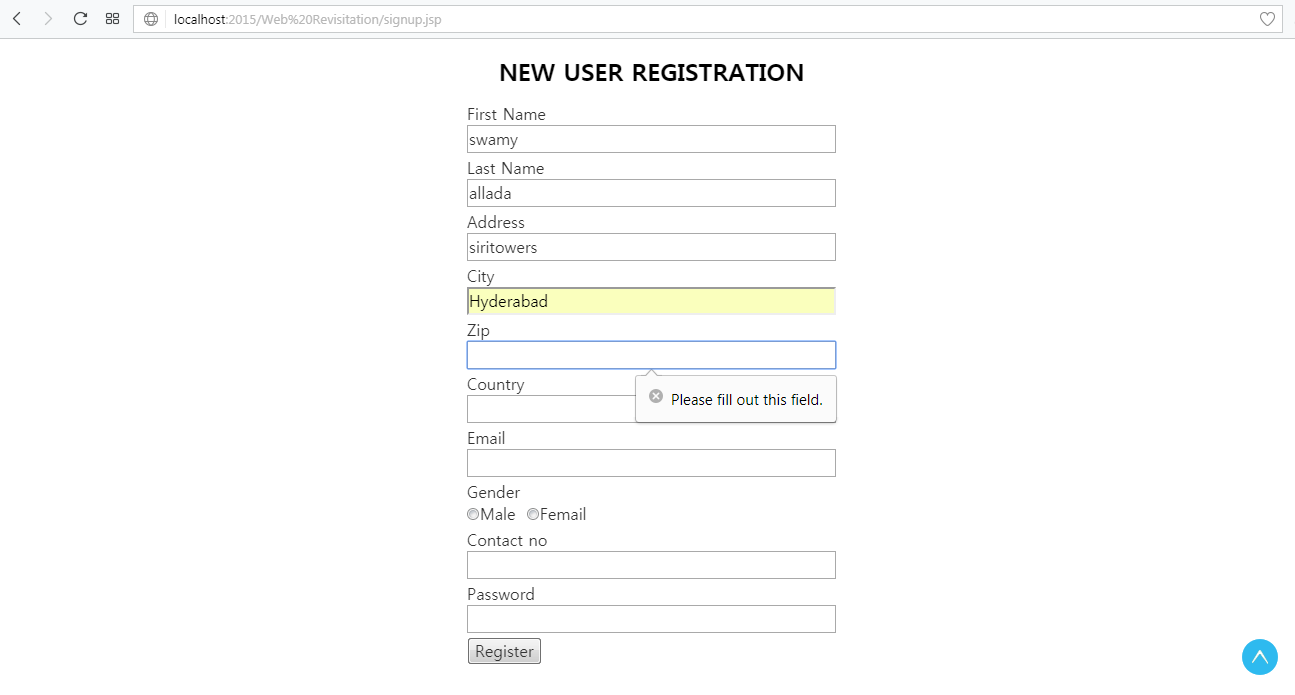
If we give in valid credentials and try to login it gives you an error message like Login fail

**Activity Context Tree Input**

****

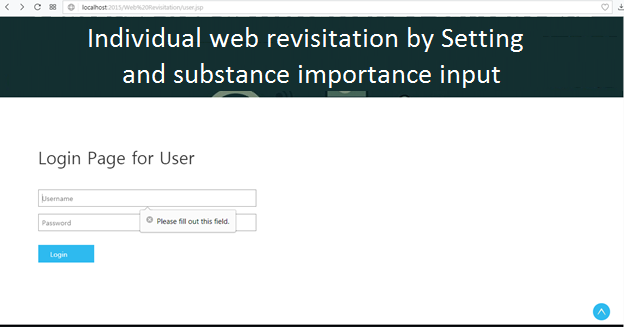
When admin try to add some Activity Context in our application if try with null values it give a validation warning like please fill out this field

**Application Users Registration with Some Null values**

****

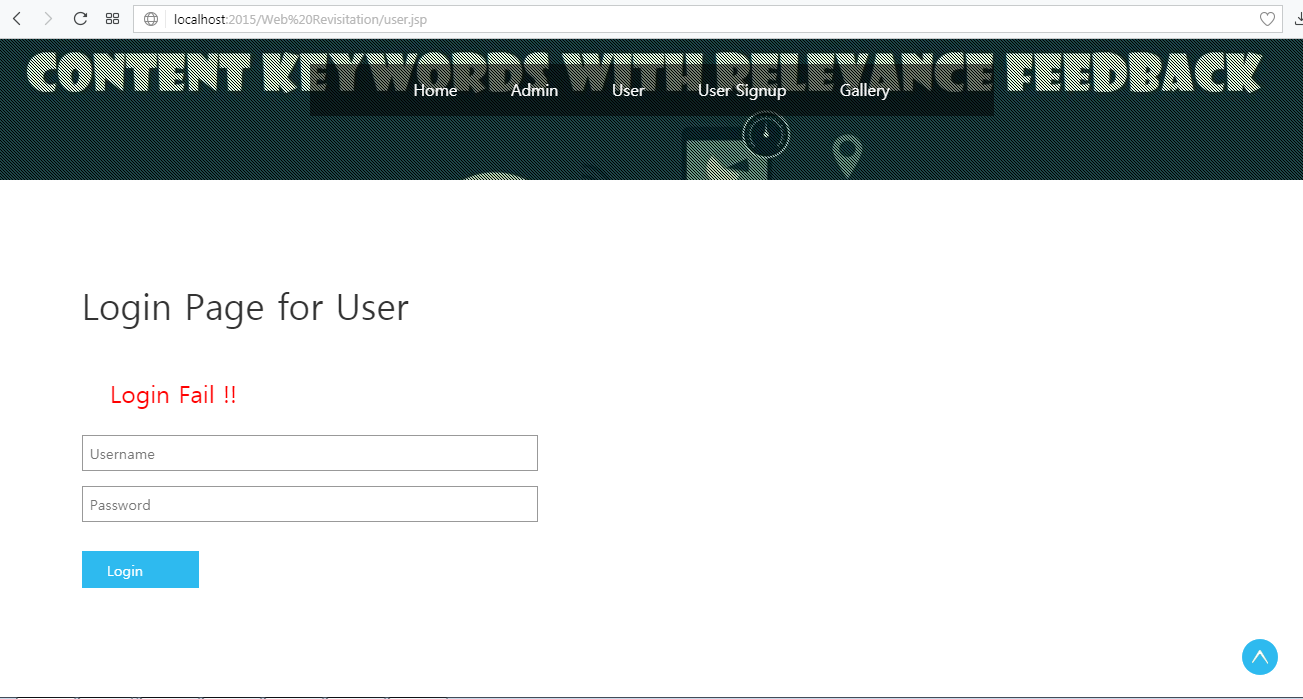
**When user dot give all the Mandatory fields Its give a Warning message like please fill out the field**

**Application Users try to login with Null values**

****

**When users try to login with null values it gives an validation message that pleas fill out this field**

**User Give Invalid user id or Password**

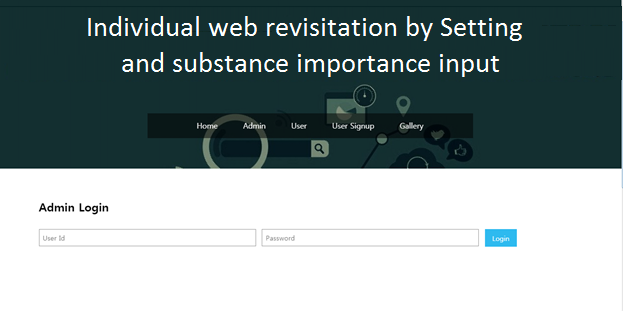
****

**If user gives an invalid user name or password it gives an error message like Login fails**

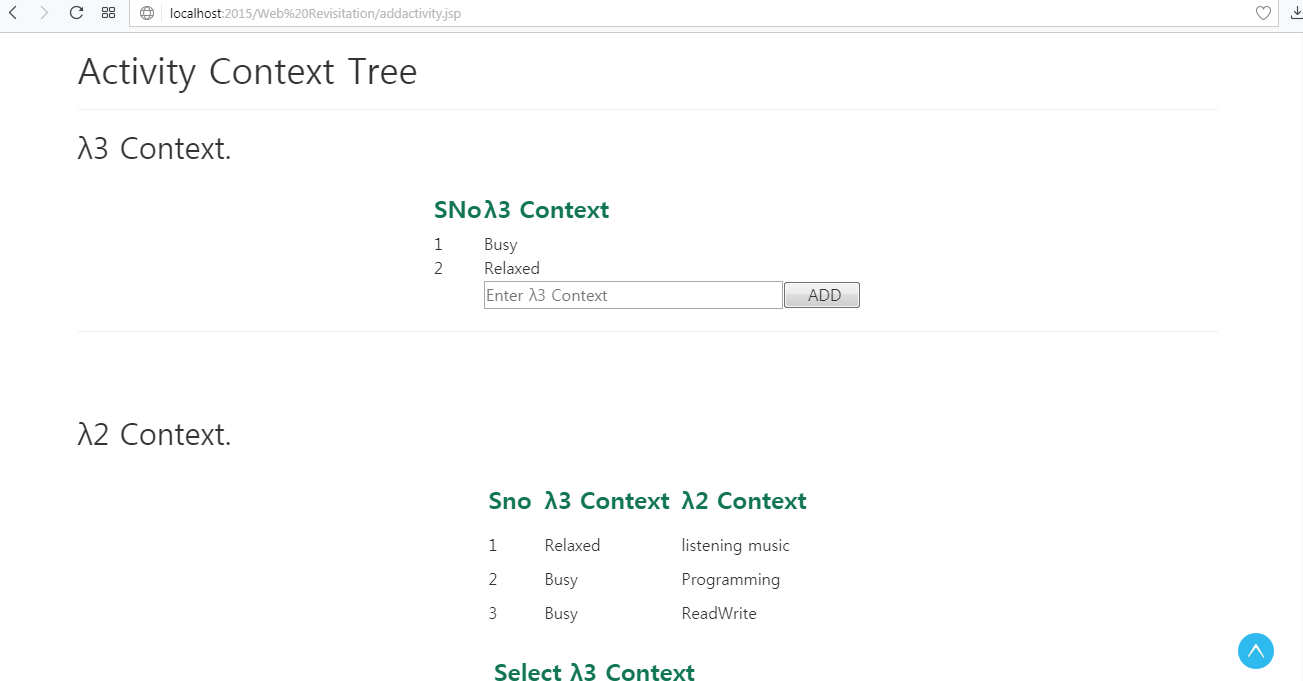
**7 .Results & graph**



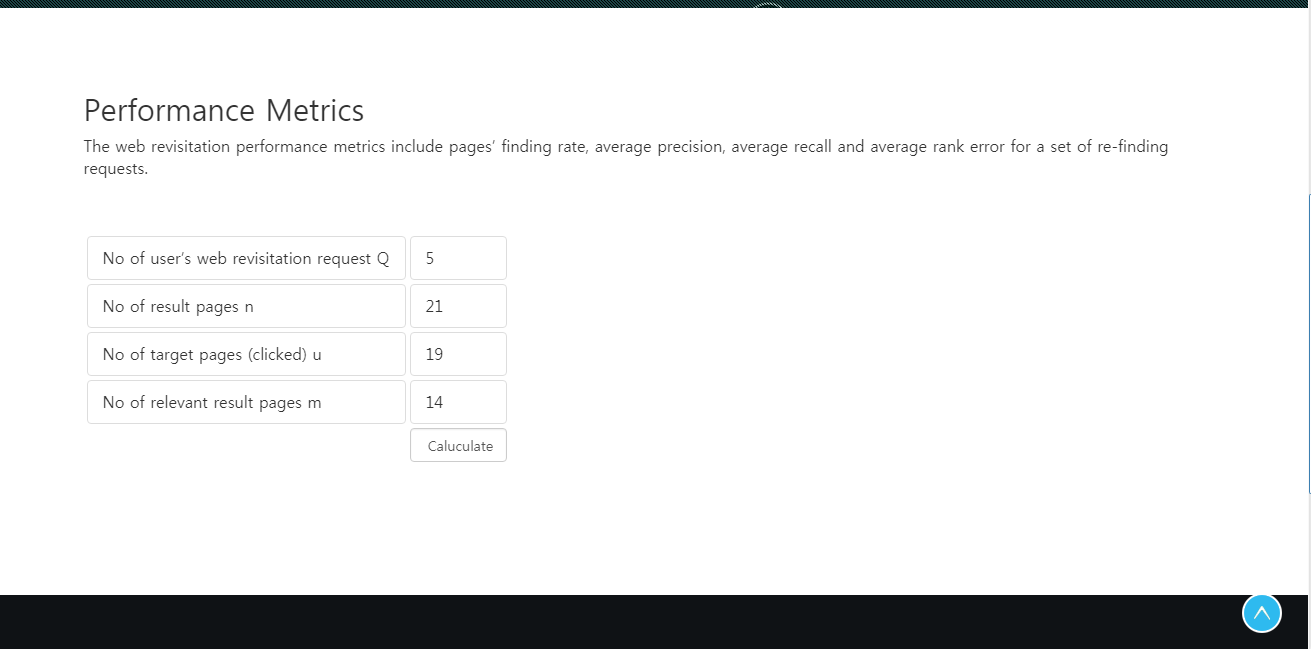
Project Home Page



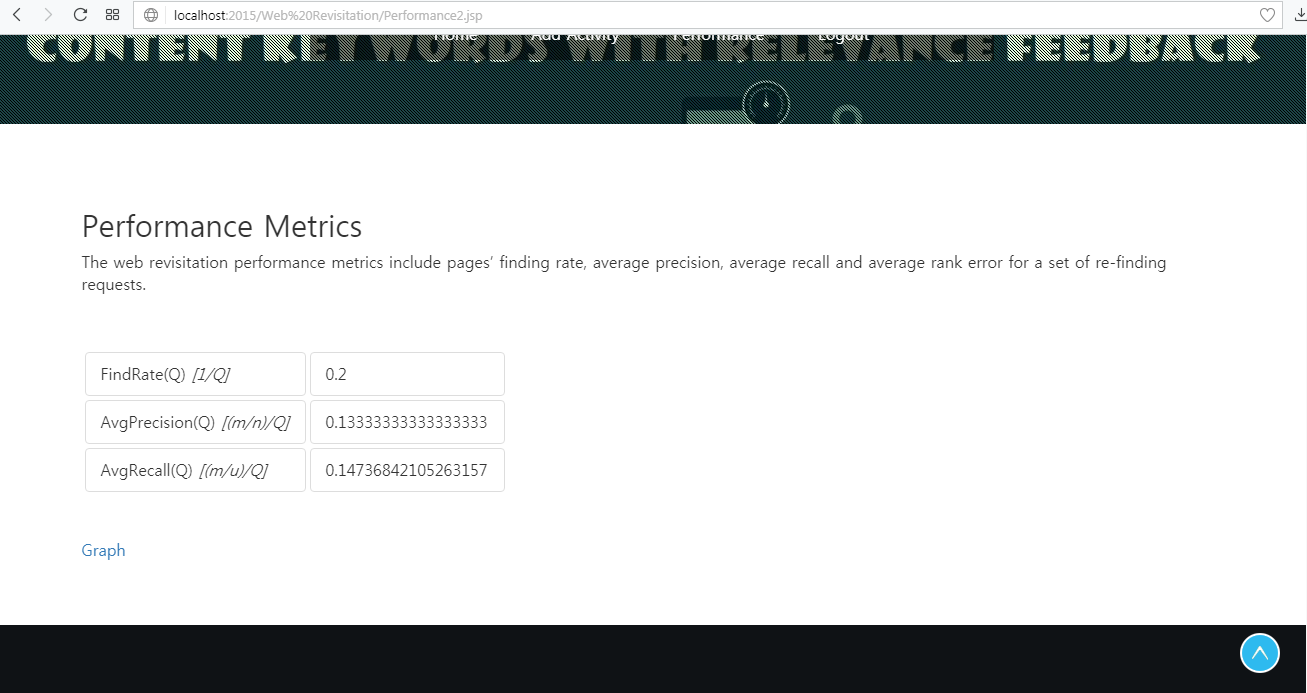
Admin Login Page



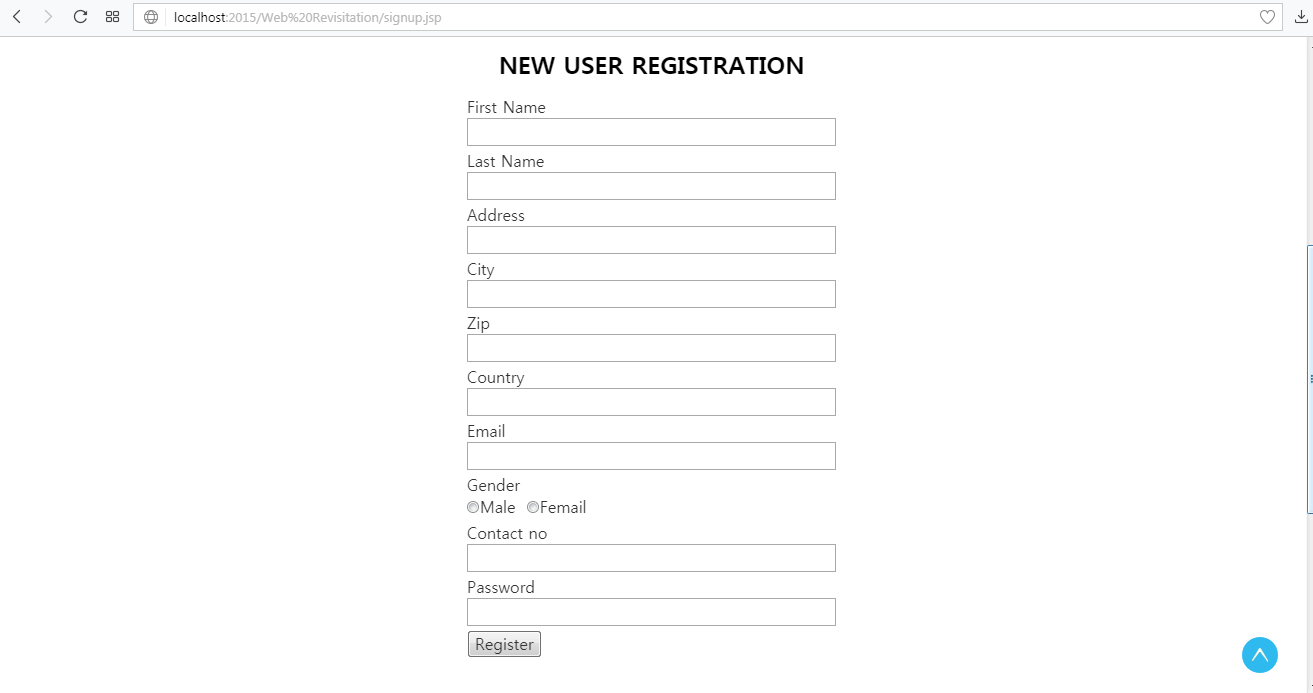
Activity Context Tree Making Page



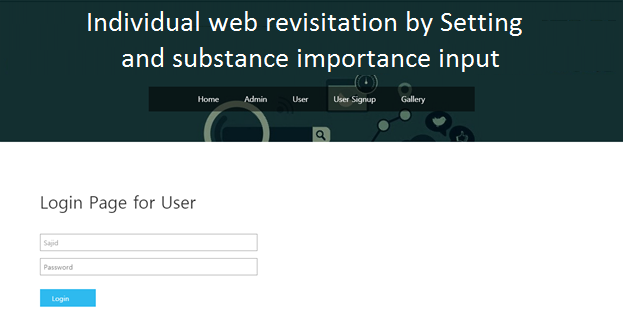
Performance Metrics Page



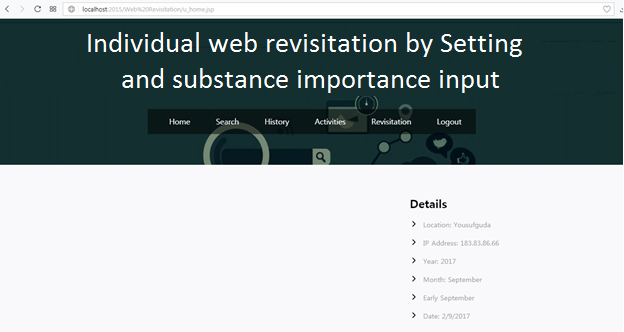
Performance Metrics Calculation Result



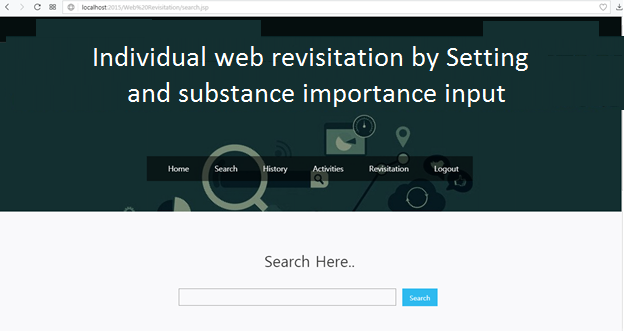
New user Registration page



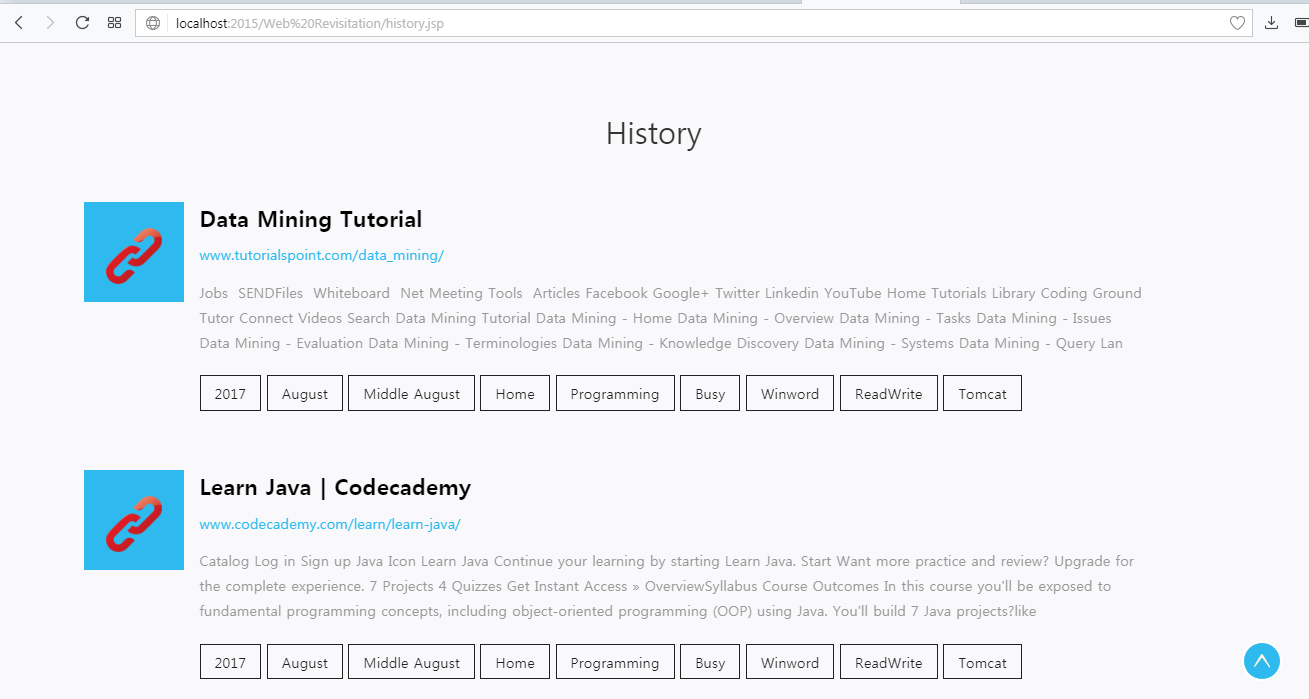
User Login Page



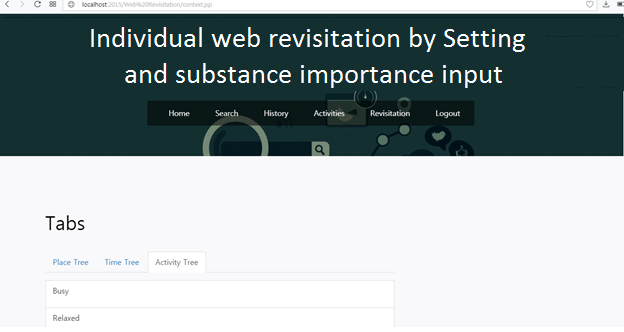
User Home Page



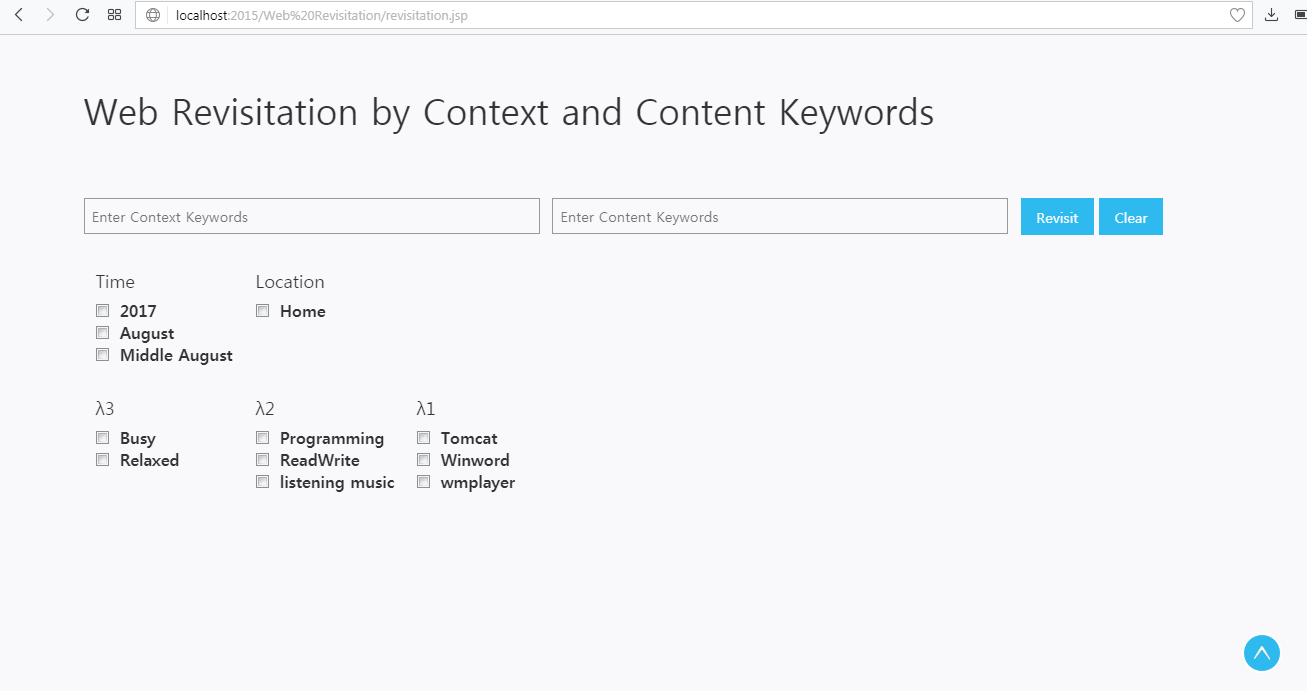
Users Search Page



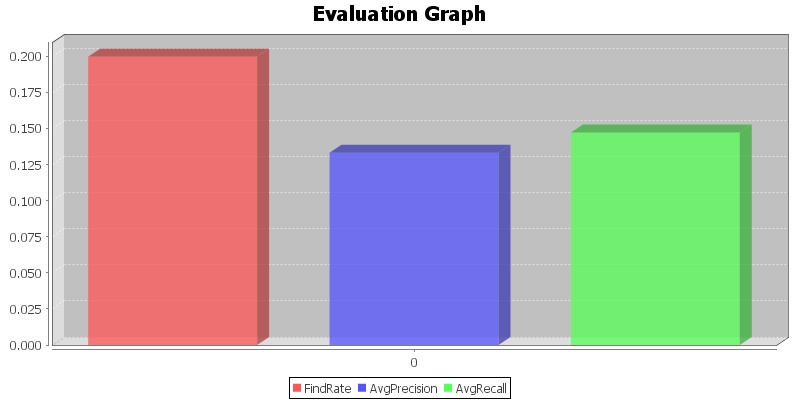
Users History Display Page



Activities of User like Place tree, Time Tree, Activity tree



Web Revisitation by Setting and substance Users search keywords Page



Graph Showing Performance in discovery rate, Performance in average precision, Performance in and average recall

**8. Conclusions**

Our Project Practical results presentation the efficacy and applicability of the suggest technique. Drawing on the attribute of human encephalon recollection in organizing and exploiting episodic events and semantic words in information recollect, in our project we present an Individual web revisitation technique predicated on setting and substance users search keywords. Our upcoming work will be on presage of end users’ revisitation, elongating the technique to fortify to end-users’ equivocal re-discovery requests, and incorporating convivial context factors in information re-discovery. The logic suggests by the toolbar. Context occasions and page content are separately sorted out as probabilistic setting trees and probabilistic term records, which progressively advance by debasement and support with pertinence consequentiality input.

**9 References**

[1] A. Cockburn, S. Greenberg, S. Jones, B. Mckenzie, and M. Moyle. Improving web page revisitation: analysis, design and evaluation. IT & Society, 1(3):159–183, 2003.

[2] L. Tauscher and S. Greenberg. How people revisit web pages: empirical discoverys and implications for the design of history systems. International Journal of Human Computer Studies, 47(1):97– 137, 1997.

[3] J. Teevan, E. Adar, R. Jones, and M. Potts. Information re-retrieval: repeat queries in yahoo’s logs. In SIGIR, pages 151–158, 2007.

[4] M. Mayer. Web history tools and revisitation support: a survey of existing approaches and directions. Foundations and Trends in HCI, 2(3):173–278, 2009.

[5] L. C. Wiggs, J. Weisberg, and A. Martin. Neural correlates of semantic and episodic memory retrieval. Neuropsychologia, pages 103–118, 1999.

[6] M. Lamming and M. Flynn. ”forget-me-not”: intimate computing in support of human memory. In FRIEND21 Intl. Symposium on Next Generation Human Interface, 1994.

[7] E. Tulving. What is episodic memory? Current Directions intheoreticals Science, 2(3):67–70, 1993.

[8] C. E. Kulkarni, S. Raju, and R. Udupa. Memento: unifying content and context to aid webpage re-visitation. In UIST, pages 435–436, 2010.

[9] J. Hailpern, N. Jitkoff, A. Warr, K. Karahalios, R. Sesek, and N. Shkrob. Youpivot: improving recall with contextual search. In CHI, pages 1521–1530, 2011.

[10] T. Deng, L. Zhao, H. Wang, Q. Liu, and L. Feng. Refinder: a context-based information re-discovery system. IEEE TKDE, 25(9):2119–2132, 2013.

[11] T. Deng, L. Zhao, and L. Feng. Enhancing web revisitation by contextual users search keywords. In ICWE, pages 323–337, 2013.

[12] H. Takano and T. Winograd. Dynamic bookmarks for the WWW. In HYPERTEXT, pages 297–298, 1998.

[13] S. Kaasten and S. Greenberg. Integrating back, history and bookmarks in web browsers. In HCI, pages 379–380, 2001.

[14] J. A. Gamez, J. L. Mateo, and J. M. Puerta. Improving revisitation browsers capability by using a dynamic bookmarks personal toolbar. In WISE, pages 643–652, 2007.

[15] R. Kawase, G. Papadakis, E. Herder, and W. Nejdl. Beyond the usual suspects: contextaware revisitation support. In HT, pages 27–36, 2011.