

JAVA WEB APPLICATION DEVELOPMENT



Session: 2014 –2015

A Industrial Training Report Submitted To

Pratap University, Jaipur

Towards Partial Fulfillment of Degree of

Bachelor of Technology

In

Computer Science and Engineering

Submitted To:

Head of Department

Submitted By:

ANURAG GAUTAM

Roll: CS110008

Enr. No: PU11100447

CERTIFICATE



This is to certify that the Industrial Training report entitled “Java Web Application Development” submitted by ANURAG GAUTAM of final year **B.Tech** in the year 2014-15 of **Computer Science and Engineering** department of this institute is a satisfactory account of his Industrial Training work based on syllabus which is approved for the award of degree of Bachelor of Technology.

Internal Examiner

External Examiner

Acknowledgement

I express my sincere gratitude to my guide Mr Nitin Purohit, Pratap University, for his excellent guidance throughout the course of this investigation.

And extremely thanks to the colleagues and trainers in LINUXWORLD, JAIPUR for providing necessary tools and modules.

I am extremely grateful to entire faculty community of the Department, for their invaluable suggestion

I sincerely thanks the authorities of Institute Library.

My sincere thanks for providing invaluable and uninterrupted Computing facility support in computing lab. Thanks to Mr. Sanjay Negi, Lab Assistant, for his co-operation in various computing and printing facilities.

My gratitude to my colleague and friend throughout the project. My heartiest love to my parents, brother and sister.

TABLE OF CONTENT

1. Abstract

2. Chapter-I

- About the Company

3. Technologies

- Java
- Java Bean
- JDBC
- Servlet
- JSP

4. Chapter-II

- Object
- Users
- Modules
- Technologies

- *Future Scopes*

5. *Reference*

LIST OF FIGURES

1. *Virtual Machine*
2. *JDBC DRIVER 1*
3. *JDBC DRIVER 2*
4. *JDBC DRIVER 3*
5. *JDBC DRIVER 4*
6. *CGI*
7. *SERVLET*

8. JSP
9. User Registration
10. User Login
11. Services to user
12. Online Query
13. Administrator Login
14. Services to Admin
15. Sessions

CHAPTER-1

COMPANY PROFILE

LinuxWorld ('LW') is a fast growing ISO 9001:2008 Certified Organisation; fully governed by young and energetic Technocrats, dedicated to Open Source technologies and Linux promotion. Since its inception in the year 2005, LW have achieved the status of centre of excellence wherein there is latest technology, innovative developing methodology, state of the art infrastructure and individual needs of employees are identified and executed professionally, efficiently & ethically. Red Hat rewarded LinuxWorld as "The Most Promising Partner"

We are the Red Hat Partner; Today Red Hat is the world's most trusted provider of Linux and open source technology. The most recognized Linux brand in the world. Red Hat serves global enterprises through technology and services made possible by the open source model, include Red Hat Enterprise Linux operating platforms and features RHCE, the global standard Linux certification.

In the field of Security, We are EC-Council Accredited Partner for Training & we also serve as Examination Centre.

LW is committed to nation building through extending its high end technical support services to MNC's and organisations.

Academic institutions are the natural channel to introduce Linux knowledge and skills to a diverse student population. As the Linux distributor and service provider, LW continues its long-standing partnership with the education market by providing the world's only 100% hands-on Linux curriculum designed on a competency based framework that includes live- system testing measurements. Add further quality & diversity to your IT program and/or teach the most thorough, relevant and up-to- date Linux skills & certification.

High end Business & Training Services (i.e. Training & Development Centre):

LW provides training by in-house experts or certified trainers or corporate developers in areas related to System & Network Administration, Programming Languages, Applications/Software Packages, and Server Administration etc. depending on the skills required by the customer.

LW is structured around its Customers, in which customers are the focus of the organisation. Customers lie at the heart of our strategy. We encourage our customers to self-manage, identifying for themselves where their needs lie. Following are the few courses available in various areas like security, certificate courses, development, database management, etc.

- Ethical Hacking (Basic & Advance Module)
- Web Application Development with PHP
- RHCE, RHCSS, RHCA
- CCENT, CCNA, CCNP, CCIE and many more in the list

Technical Support Services:

With a passionate technical team, fully committed to the development and progress of Open Source, LW offers its clients high end support solutions every time. LINUXWORLD provides support for all major Open Source applications

Research & Development Centre:

VISION: "To come out with the best operating system i.e. a globally acceptable product this would be different, new and useful to the entire world."

The main objective of LW R&D's team is to work dedicate on the loopholes present in the existing Linux Operating System, to ensure that the same are removed and plug-in the new development which does not exist in Linux OS at present. In other words, it follows Push - Pull Strategy i.e. Pull out the loopholes and Push in the new developments.

Many companies' combines various software available in the communities/platforms and edits the existing ones to some extent, to give a new look to the same. As it is rightly said "Old wine in the New bottle" whereas LW's R&D Centre develops its own networking tools, system monitoring tools and many other software's and LINUX related tools with their own and new ideas and not with the help of the existing ones.

LinuxWorld R&D centre is developing, maintaining & promoting its own special tools based on open source technology & softwares.

According to survey conducted:-

"LW is the only organisation in India having a wholly and solely dedicated R&D Centre"

TECHNOLOGY

1. Introduction to Java

History

Java is a programming language created by James Gosling from Sun Microsystems (Sun) in 1991. The first publicly available version of Java (Java 1.0) was released in 1995.

Sun Microsystems was acquired by the Oracle Corporation in 2010. Oracle has now the steermanship for Java.

Over time new enhanced versions of Java have been released. The current version of Java is Java 1.7 which is also known as *Java 7*.

From the Java programming language the *Java platform* evolved. The Java platform allows software developers to write program code in other languages than the Java programming language which still runs on the Java virtual machine. The *Java platform* is usually associated with the *Java virtual machine* and the *Java core libraries*.

Java and Open Source

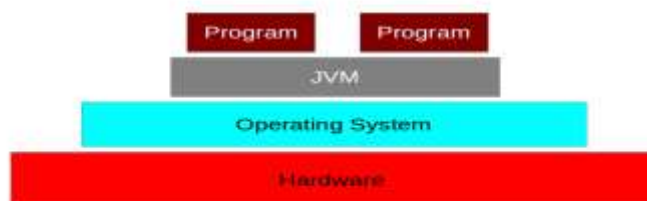
In 2006 Sun started to make Java available under the GNU General Public License (GPL). Oracle continues this project called *OpenJDK*.

Java virtual machine

The Java virtual machine (JVM) is a software implementation of a computer that executes programs like a real machine.

The Java virtual machine is written specifically for a specific operating system, e.g., for Linux a special implementation is required as well as for Windows.

Java Virtual Machine (VM)



Java programs are compiled by the Java compiler into *bytecode*. The Java virtual machine interprets this *bytecode* and executes the Java program.

Java Runtime Environment vs. Java Development Kit

A Java distribution typically comes in two flavors, the *Java Runtime Environment* (JRE) and the *Java Development Kit* (JDK).

The Java runtime environment (JRE) consists of the JVM and the Java class libraries. Those contain the necessary functionality to start Java programs.

The JDK additionally contains the development tools necessary to create Java programs. The JDK therefore consists of a Java compiler, the Java virtual machine and the Java class libraries.

Characteristics of Java

The target of Java is to write a program once and then run this program on multiple operating systems.

Java has the following properties:

- Platform independent: Java programs use the Java virtual machine as abstraction and do not access the operating system directly. This makes Java programs highly portable. A Java program (which is standard-compliant and follows certain rules) can run unmodified on all supported platforms, e.g., Windows or Linux.
- Object-orientated programming language: Except the primitive data types, all elements in Java are objects.
- Strongly-typed programming language: Java is strongly-typed, e.g., the types of the used variables must be pre-defined and conversion to other objects is relatively strict, e.g., must be done in most cases by the programmer.
- Interpreted and compiled language: Java source code is transferred into the bytecode format which does not depend on the target platform. These bytecode

instructions will be interpreted by the Java Virtual machine (JVM). The JVM contains a so called Hotspot-Compiler which translates performance critical bytecode instructions into native code instructions.

- Automatic memory management: Java manages the memory allocation and de-allocation for creating new objects. The program does not have direct access to the memory. The so-called garbage collector automatically deletes objects to which no active pointer exists.

The Java syntax is similar to C++. Java is case-sensitive, e.g., variables called `myValue` and `myvalue` are treated as different variables.

Development Process with Java

Java source files are written as plain text documents. The programmer typically writes Java source code in an *Integrated Development Environment* (IDE) for programming. An IDE supports the programmer in the task of writing code, e.g., it provides auto-formatting of the source code, highlighting of the important keywords, etc.

At some point the programmer (or the IDE) calls the Java compiler (*javac*). The Java compiler creates the *bytecode* instructions. These instructions are stored in *.class* files and can be executed by the Java Virtual Machine.

Garbage collector

The JVM automatically re-collects the memory which is not referred to by other objects. The Java *garbage collector* checks all object references and finds the objects which can be automatically released.

While the garbage collector relieves the programmer from the need to explicitly manage memory, the programmer still need to ensure that he does not keep unneeded object references, otherwise the garbage collector cannot release the associated memory. Keeping unneeded object references are typically called *memory leaks*.

Classpath

The *classpath* defines where the Java compiler and Java runtime look for *.class* files to load. These instructions can be used in the Java program.

For example, if you want to use an external Java library you have to add this library to your classpath to use it in your program.

2. Java Bean

A Java Bean is a java class that should follow following conventions:

- It should have a no-arg constructor.
- It should be Serializable.
- It should provide methods to set and get the values of the properties, known as getter and setter methods.

Why use Java Bean?

According to Java white paper, it is a reusable software component. A bean encapsulates many objects into one object, so we can access this object from multiple places.

Moreover, it provides the easy maintenance.

Simple example of java bean class

```
1. //Employee.java
2.
3. package mypack;
4. public class Employee implements java.io.Serializable{
5.     private int id;
6.     private String name;
7.
8.     public Employee(){
9.
10.    public void setId(int id){this.id=id;}
11.
12.    public int getId(){return id;}
13.
14.    public void setName(String name){this.name=name;}
15.
16.    public String getName(){return name;}
17.
18. }
```

How to access the java bean class?

To access the java bean class, we should use getter and setter methods.

```
1. package mypack;
2. public class Test{
3.     public static void main(String args[]){
4.
5.     Employee e=new Employee();//object is created
6.
7.     e.setName("Arjun");//setting value to the object
8.
9.     System.out.println(e.getName());
```

10.
11.}}

3. JDBC

JDBC stands for **Java Database Connectivity**, which is a standard Java API for database-independent connectivity between the Java programming language and a wide range of databases.

The JDBC library includes APIs for each of the tasks commonly associated with database usage:

- Making a connection to a database
- Creating SQL or MySQL statements
- Executing that SQL or MySQL queries in the database
- Viewing & Modifying the resulting records

JDBC Driver Types

JDBC drivers are divided into four types or levels. The different types of jdbc drivers are:

Type 1: JDBC-ODBC Bridge driver (Bridge)

Type 2: Native-API/partly Java driver (Native)

Type 3: AllJava/Net-protocol driver (Middleware)

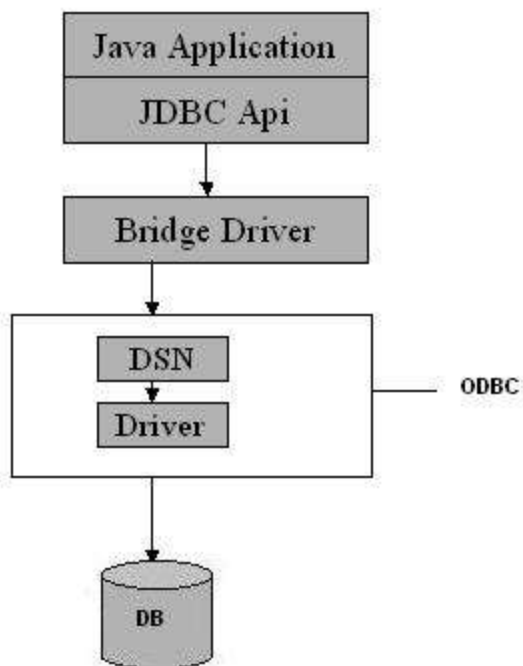
Type 4: All Java/Native-protocol driver (Pure)

4 types of jdbc drivers are elaborated in detail as shown below:

Type 1 JDBC Driver

JDBC-ODBC Bridge driver

The Type 1 driver translates all JDBC calls into ODBC calls and sends them to the ODBC driver. ODBC is a generic API. The JDBC-ODBC Bridge driver is recommended only for experimental use or when no other alternative is available.



Type 1: JDBC-ODBC Bridge

Advantage

The JDBC-ODBC Bridge allows access to almost any database, since the database's ODBC drivers are already available.

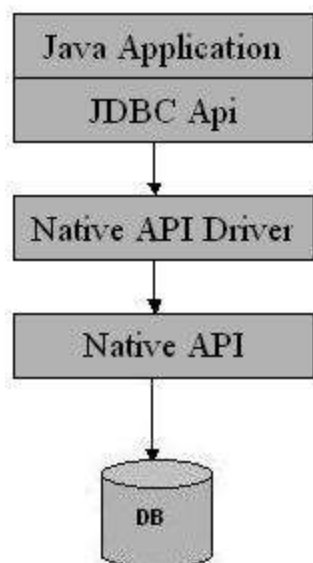
Disadvantages

1. Since the Bridge driver is not written fully in Java, Type 1 drivers are not portable.
2. A performance issue is seen as a JDBC call goes through the bridge to the ODBC driver, then to the database, and this applies even in the reverse process. They are the slowest of all driver types.
3. The client system requires the ODBC Installation to use the driver.
4. Not good for the Web.

Type 2 JDBC Driver

Native-API/partly Java driver

The distinctive characteristic of type 2 jdbc drivers are that Type 2 drivers convert JDBC calls into database-specific calls i.e. this driver is specific to a particular database. Some distinctive characteristic of type 2 jdbc drivers are shown below. Example: Oracle will have oracle native api.



Type 2: Native api/ Partly Java Driver

Advantage

The distinctive characteristic of type 2 jdbc drivers are that they are typically offer better performance than the JDBC-ODBC Bridge as the layers of communication (tiers) are less than that of Type

1 and also it uses Native api which is Database specific.

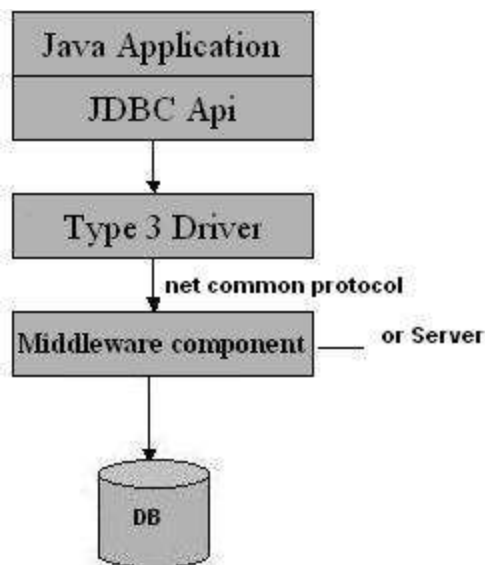
Disadvantage

1. Native API must be installed in the Client System and hence type 2 drivers cannot be used for the Internet.
2. Like Type 1 drivers, it's not written in Java Language which forms a portability issue.
3. If we change the Database we have to change the native api as it is specific to a database
4. Mostly obsolete now
5. Usually not thread safe.

Type 3 JDBC Driver

All Java/Net-protocol driver

Type 3 database requests are passed through the network to the middle-tier server. The middle-tier then translates the request to the database. If the middle-tier server can in turn use Type1, Type 2 or Type 4 drivers.



Type 3: All Java/ Net-Protocol Driver

Advantage

1. This driver is server-based, so there is no need for any vendor database library to be present on client machines.
2. This driver is fully written in Java and hence Portable. It is suitable for the web.
3. There are many opportunities to optimize portability, performance, and scalability.
4. The net protocol can be designed to make the client JDBC driver very small and fast to load.
5. The type 3 driver typically provides support for features such as caching (connections, query results, and so on), load balancing, and advanced system administration such as logging and auditing.
6. This driver is very flexible allows access to multiple databases using one driver.
7. They are the most efficient amongst all driver types.

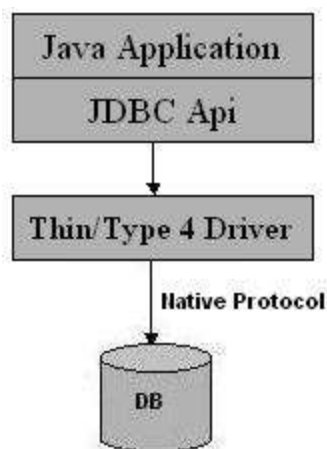
Disadvantage

It requires another server application to install and maintain. Traversing the recordset may take longer, since the data comes through the backend server.

Type 4 JDBC Driver

Native-protocol/all-Java driver

The Type 4 uses java networking libraries to communicate directly with the database server.



Type 4: Native-protocol/all-Java driver

Advantage

1. The major benefit of using a type 4 jdbc drivers are that they are completely written in Java to achieve platform independence and eliminate deployment administration issues. It is most suitable for the web.
2. Number of translation layers is very less i.e. type 4 JDBC drivers don't have to translate database requests to ODBC or a native connectivity interface or to pass the request on to another server, performance is typically quite good.
3. You don't need to install special software on the client or server. Further, these drivers can be downloaded dynamically.

Disadvantage

With type 4 drivers, the user needs a different driver for each database.

4. Servlet

Servlet technology is used to create web application (resides at server side and generates dynamic web page).

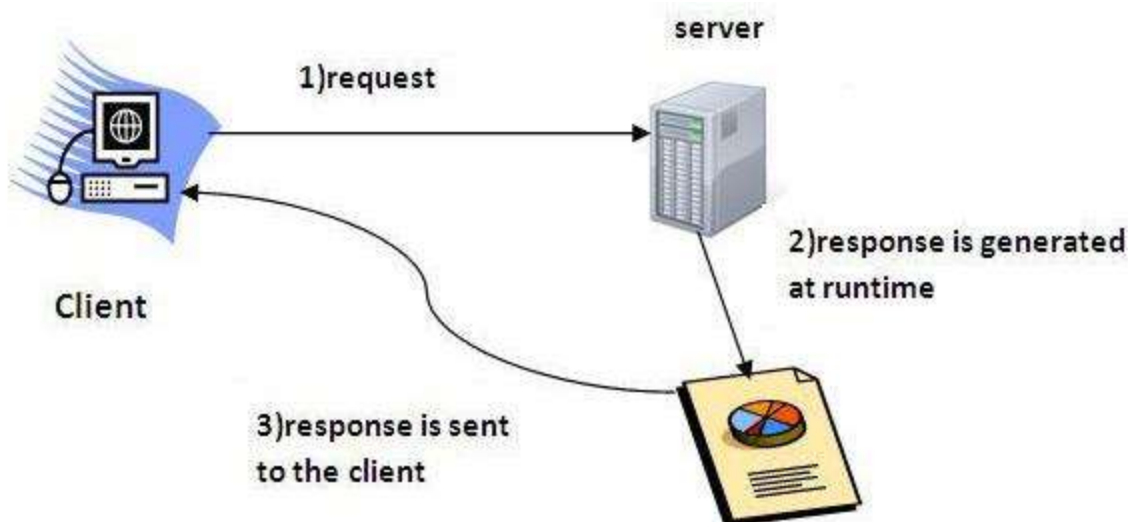
Servlet technology is robust and scalable as it uses the java language. Before Servlet, CGI (Common Gateway Interface) scripting language was used as a server-side programming language. But there were many disadvantages of this technology. We have discussed these disadvantages below.

There are many interfaces and classes in the servlet API such as Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse etc.

What is a Servlet?

Servlet can be described in many ways, depending on the context.

- Servlet is a technology i.e. used to create web application.
- Servlet is an API that provides many interfaces and classes including documentations.
- Servlet is an interface that must be implemented for creating any servlet.
- Servlet is a class that extend the capabilities of the servers and respond to the incoming request. It can respond to any type of requests.
- Servlet is a web component that is deployed on the server to create dynamic web page.

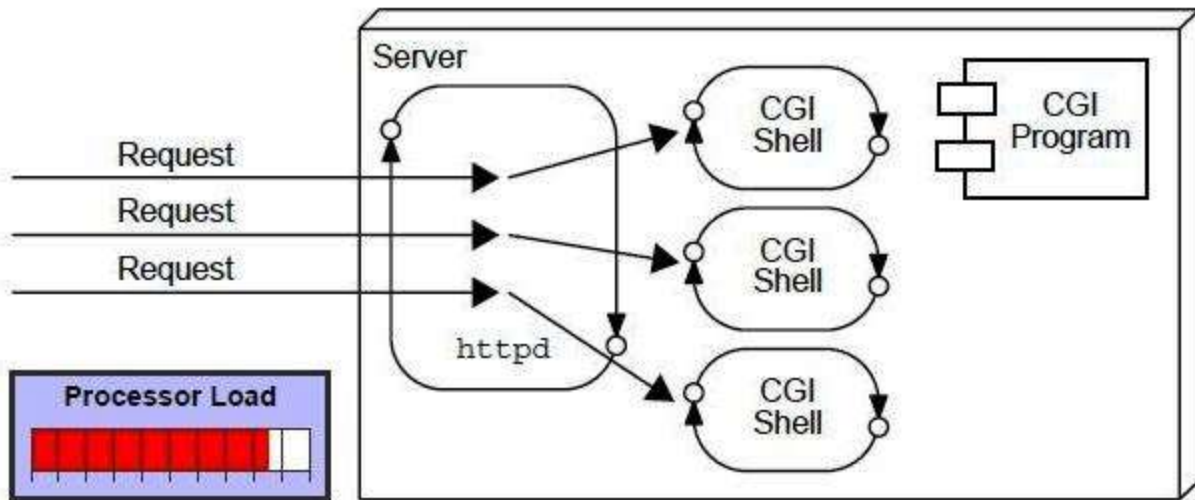


What is web application?

A web application is an application accessible from the web. A web application is composed of web components like Servlet, JSP, Filter etc. and other components such as HTML. The web components typically execute in Web Server and respond to HTTP request.

CGI(Common Gateway Interface)

CGI technology enables the web server to call an external program and pass HTTP request information to the external program to process the request. For each request, it starts a new process.

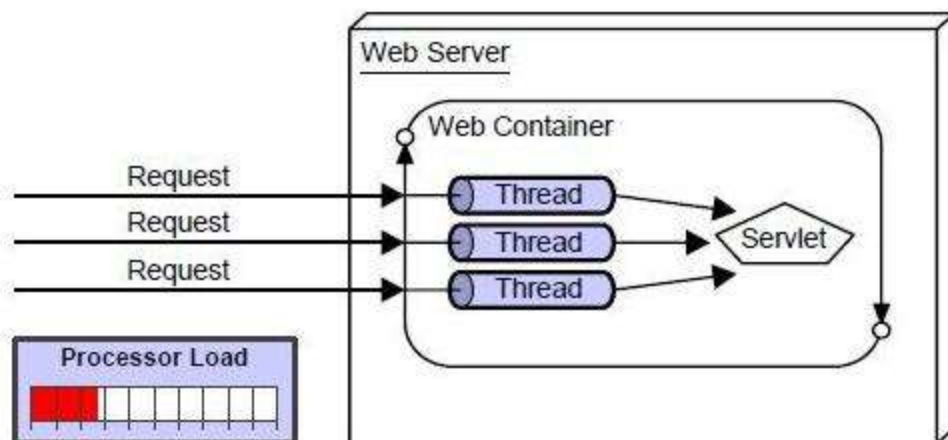


Disadvantages of CGI

There are many problems in CGI technology:

1. If number of clients increases, it takes more time for sending response.
2. For each request, it starts a process and Web server is limited to start processes.
3. It uses platform dependent language e.g. C, C++, perl.

Advantage of Servlet



There are many advantages of Servlet over CGI. The web container creates threads for handling the multiple requests to the servlet. Threads have a lot of benefits over the Processes such as they share a common memory area, lightweight, cost of communication between the threads are low. The basic benefits of servlet are as follows:

1. **better performance:** because it creates a thread for each request not process.
2. **Portability:** because it uses java language.
3. **Robust:** Servlets are managed by JVM so no need to worry about momory leak, garbage collection etc.
4. **Secure:** because it uses java language..

5. JSP

JSP technology is used to create web application just like Servlet technology. It can be thought of as an extension to servlet because it provides more functionality than servlet such as expression language, jstl etc.

A JSP page consists of HTML tags and JSP tags. The jsp pages are easier to maintain than servlet because we can separate designing and development. It provides some additional features such as Expression Language, Custom Tag etc.

Advantage of JSP over Servlet

There are many advantages of JSP over servlet. They are as follows:

1) Extension to Servlet

JSP technology is the extension to servlet technology. We can use all the features of servlet in JSP. In addition to, we can use implicit objects, predefined tags, expression language and Custom tags in JSP, that makes JSP development easy.

2) Easy to maintain

JSP can be easily managed because we can easily separate our business logic with presentation logic. In servlet technology, we mix our business logic with the presentation logic.

3) Fast Development: No need to recompile and redeploy

If JSP page is modified, we don't need to recompile and redeploy the project. The servlet code needs to be updated and recompiled if we have to change the look and feel of the application.

4) Less code than Servlet

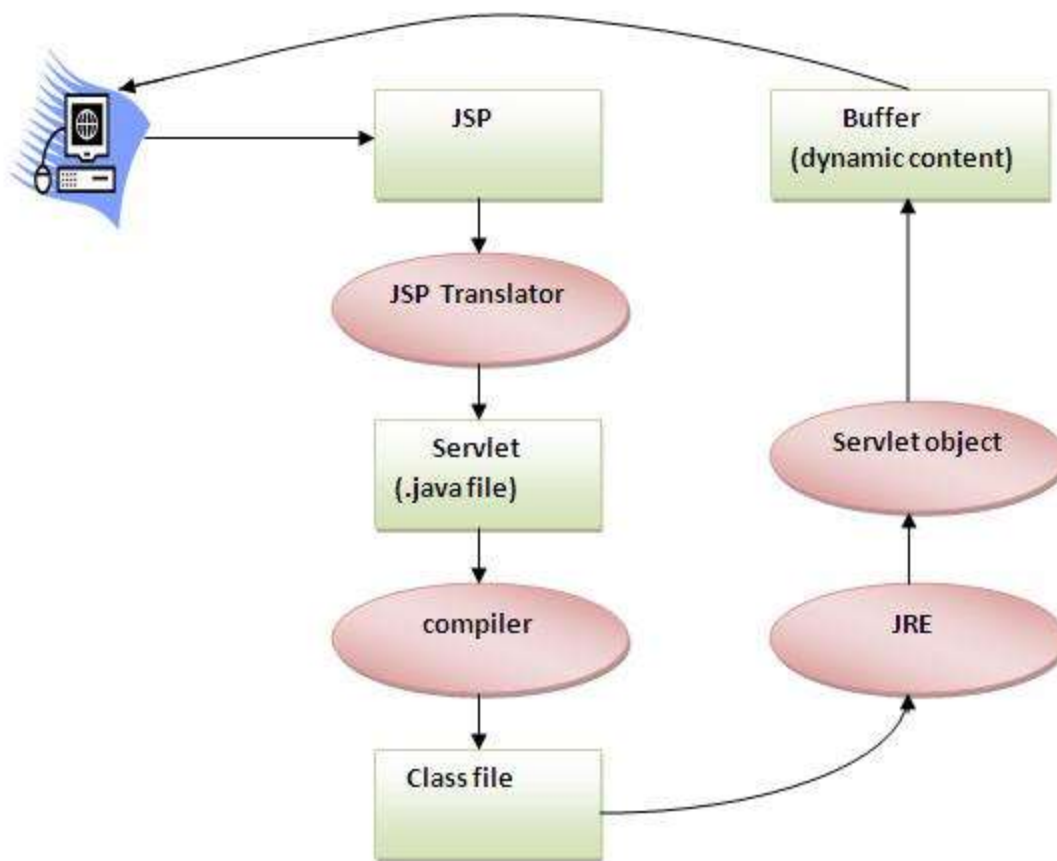
In JSP, we can use a lot of tags such as action tags, jstl, custom tags etc. that reduces the code. Moreover, we can use EL, implicit objects etc.

Life cycle of a JSP Page

The JSP pages follows these phases:

- Translation of JSP Page
- Compilation of JSP Page
- Classloading (class file is loaded by the classloader)
- Instantiation (Object of the Generated Servlet is created).
- Initialization (`jspInit()` method is invoked by the container).
- Request processing (`_jspService()` method is invoked by the container).
- Destroy (`jspDestroy()` method is invoked by the container).

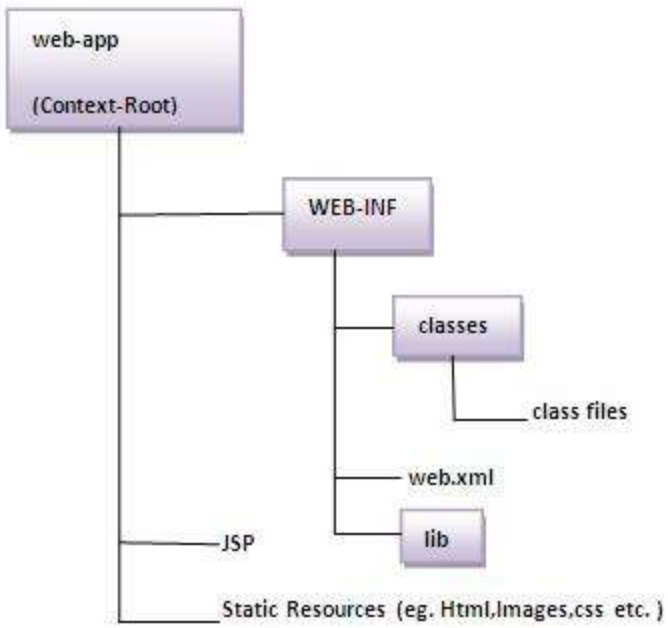
Note: `jspInit()`, `_jspService()` and `jspDestroy()` are the life cycle methods of JSP.



As depicted in the above diagram, JSP page is translated into servlet by the help of JSP translator. The JSP translator is a part of webserver that is responsible to translate the JSP page into servlet. After that Servlet page is compiled by the compiler and gets converted into the class file. Moreover, all the processes that happens in servlet is performed on JSP later like initialization, committing response to the browser and destroy.

Directory structure of JSP

The directory structure of JSP page is same as servlet. We contains the jsp page outside the WEB-INF folder or in any directory.



CHAPTER-II

PROJECT

Objective: Online Recipe Portal is a web application which provide Recipe Search in various views like recipe by categories, occasion and demand. It also provides enquiry services to the user about any recipes.

Users: There are 3 basic users in the portal:

- Administrator
- Registered User
- Un-Registered User

Scope:

- An Administrator has a right to allow the registration of a user. Insert, update and delete any registration detail of a user.
- A Registered User has a right to browse recipe of different occasion, category. He/She has a right to post a recipe and join the Online Recipe forum.
- An Un-Registered User has only a right to ask questions from the administrator.

Modules:

- Login for Administrator and User.
- Registration for User.
- Online Recipe Forum.

Technologies Used:

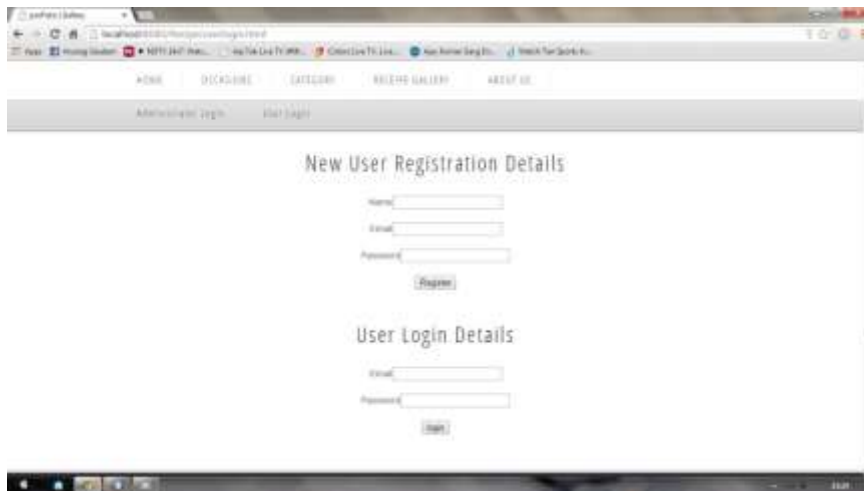
- Dreamweaver 8.0
- Netbean 8.0
- MySql
- JSP
- JDBC
- J2EE
- CSS
- JS

- Glassfish Server

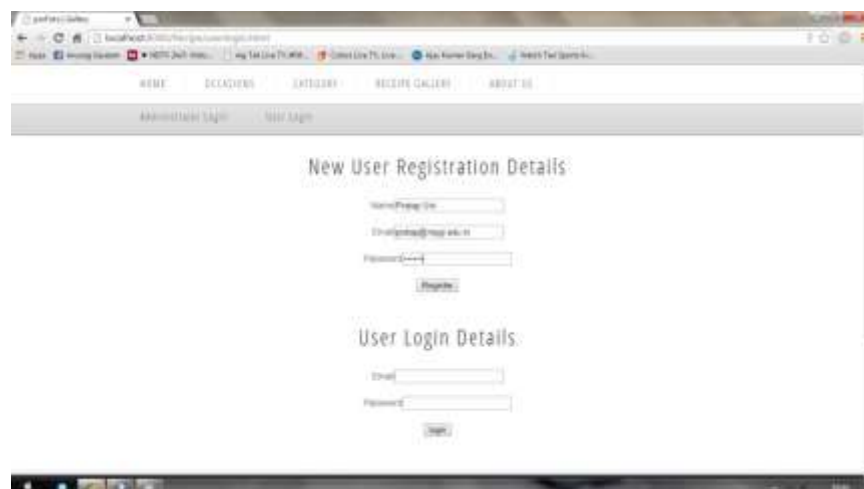
MODULES OF PROJECT

1. Registration of User

In this module a user can register himself. When he/she clicks user login, login page will appear and a portion for registration is shown as below:

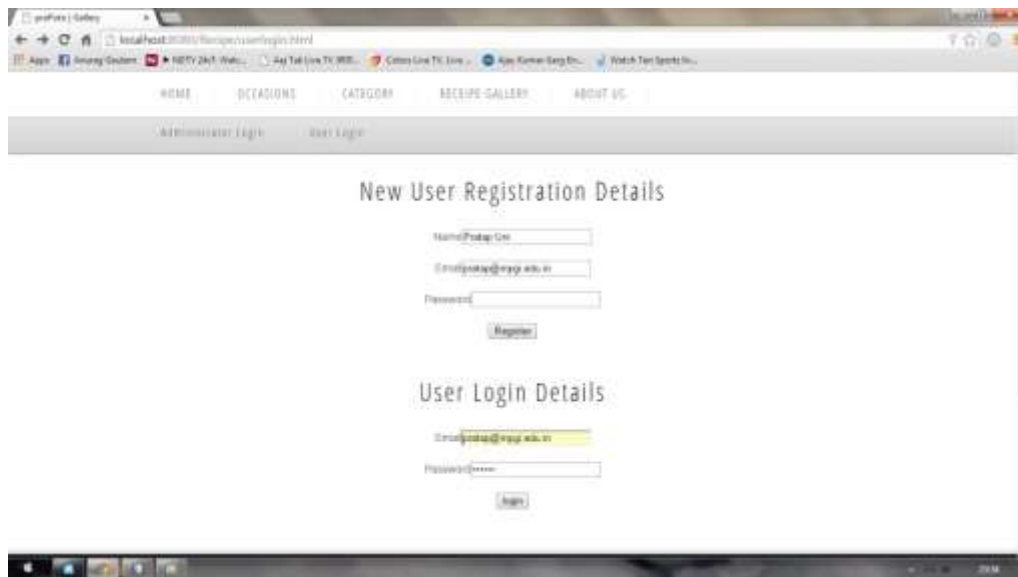
A screenshot of a web browser showing a registration form. The form is titled "New User Registration Details" and contains three input fields: "Name", "Email", and "Password". Below the "Password" field is a "Register" button. Below the registration form is a "User Login Details" section with "Email" and "Password" input fields and a "Login" button. The browser's address bar shows a URL starting with "http://localhost:8080/". The browser's tabs and address bar are visible at the top of the window.

A user can register himself/herself by entering his/her Name, Email and password.

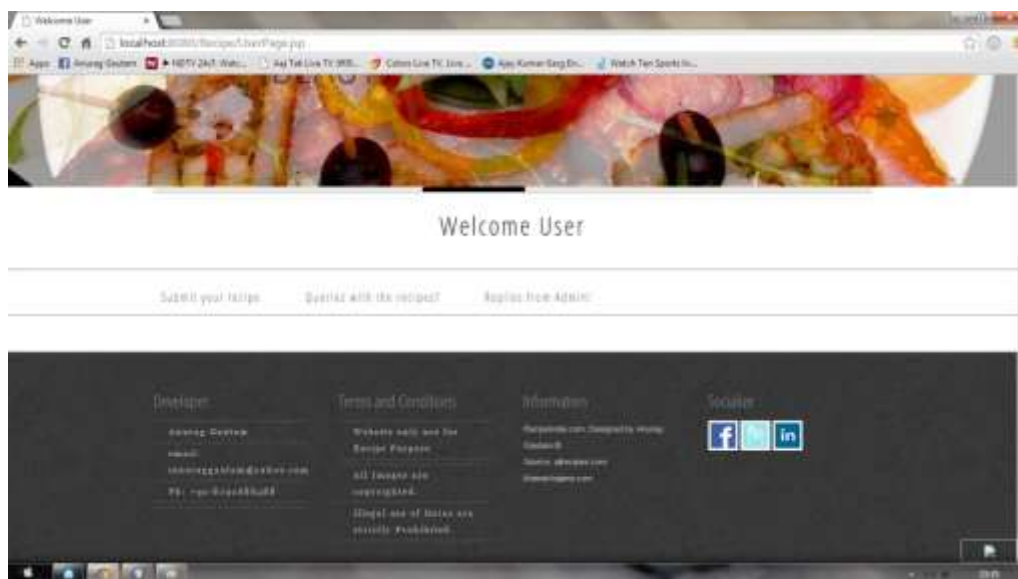
A screenshot of the same web browser showing the registration form, but now with data entered. The "Name" field contains "Name/Pragna/Dev", the "Email" field contains "Pragna@gmail.com", and the "Password" field contains "12345678". The "Register" button is still visible. The "User Login Details" section is also visible below it. The browser's address bar and tabs are visible at the top of the window.

2. Login of User

After Registration in the previous module, a user can login into the website using this module as shown:

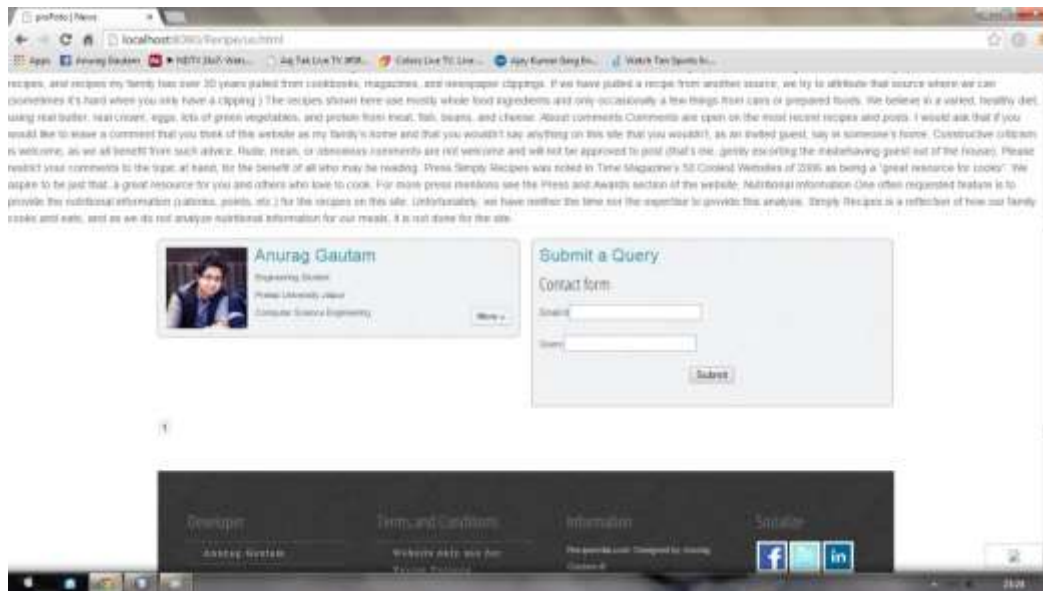


3. Services for a User

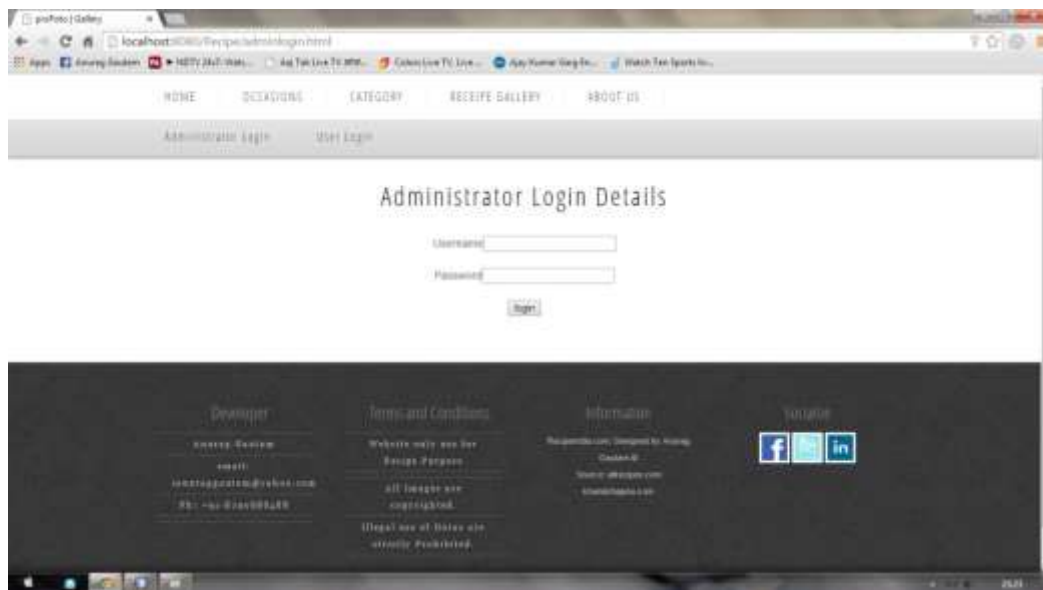


A user can post new recipe, ask question to Administrator, and checks replies from the admin.

4. Online query

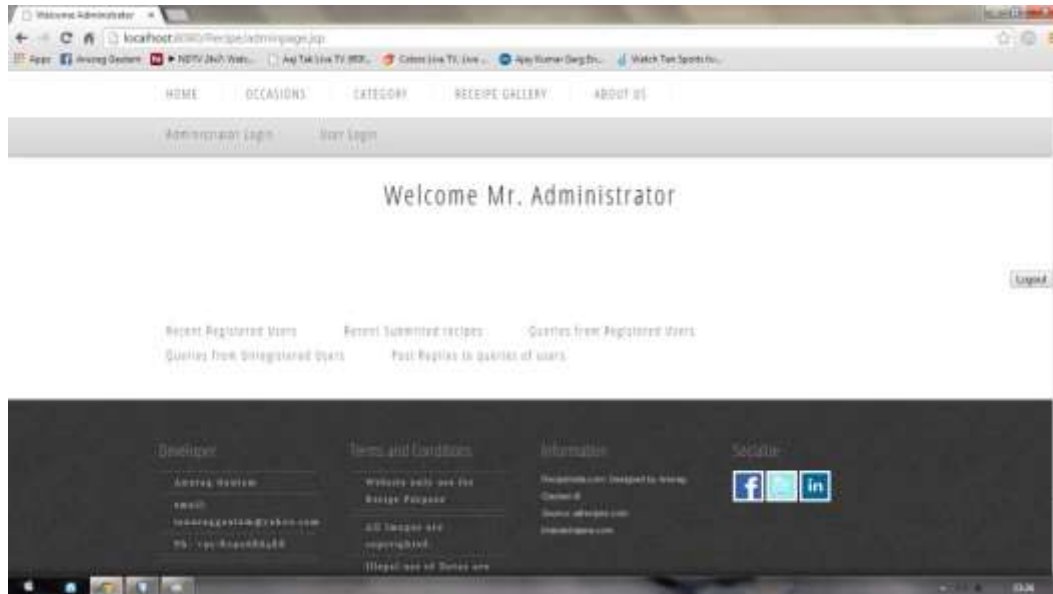


5. Administrator Login



In this module, an admin enters his/her details to enter admin module.

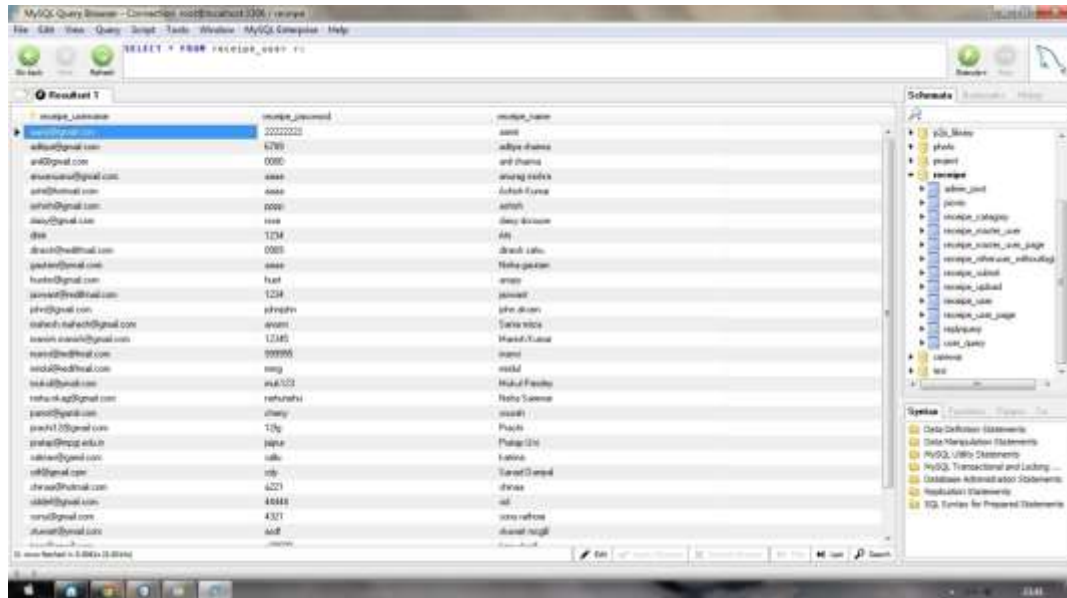
6. Services for Administrator



An administrator receives services like, he can get all the registration detail of user, posted recipes and queries. He/She also has a right to Insert, Update and Delete the entries.



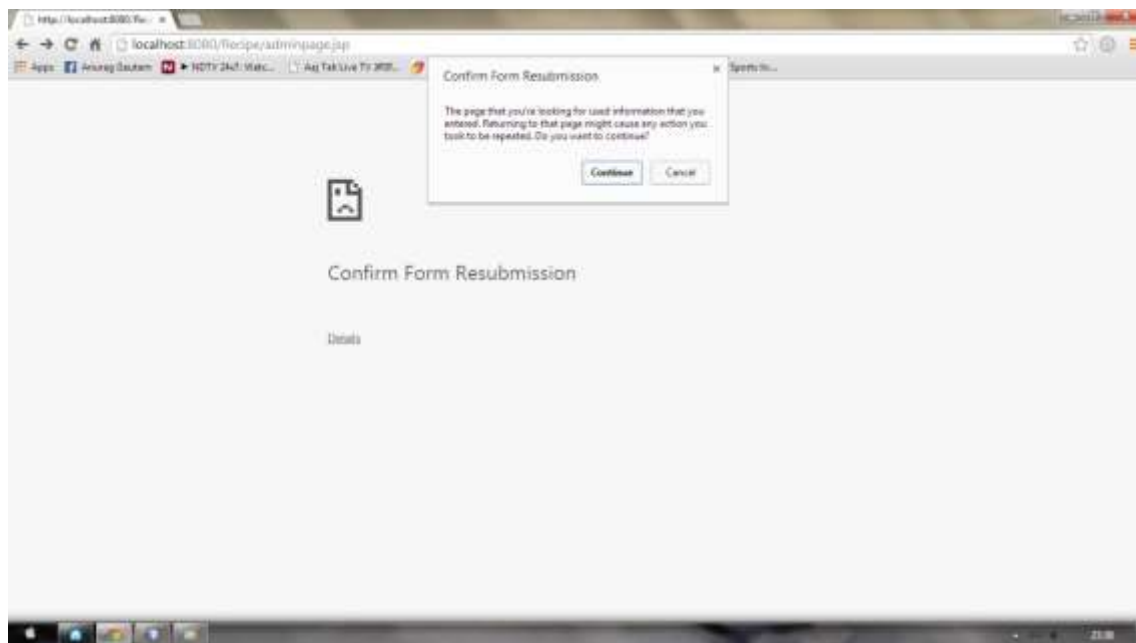
These are the list of details fetched from the database.



Administrator has rights to Insert, Update and Delete records







“Session Expired”

FUTURE SCOPES

This application can be easily implemented under various situations. We can add new features as and when we require. Reusability is possible as and when require in this application. There is flexibility in all the modules.

SOFTWARE SCOPE:

- **Extensibility** : This software is extendable in ways that its original developers may not expect. The following principles enhances extensibility like hide data structure, avoid traversing multiple links or methods, avoid case statements on object type and distinguish public and private operations.

Reusability : Reusability is possible as and when require in this application. We can update it next version. Reusable software reduces design, coding and testing cost by amortizing effort over several designs. Reducing the amount of code also simplifies understanding, which increases the likelihood that the code is correct. We follow up both

types of reusability: Sharing of newly written code within a project and reuse of previously written code on new projects.

Understandability: A method is understandable if someone other than the creator of the method can understand the code (as well as the creator after a time lapse). We use the method, which small and coherent helps to accomplish this.

• **Cost-effectiveness:** Its cost is under the budget and make within given time period. It is desirable to aim for a system with a minimum cost subject to the condition that it must satisfy the entire requirement. Scope of this document is to put down the requirements, clearly identifying the information needed by the user, the source of the information and outputs expected from the system.

REFERENCE

1. allrecipes.com
2. KhanaKhajana.com
3. Google.com
4. allindiafood.in