TERM PAPER

On

E-SMART SOLUTION

Submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology

In

Computer Science and Engineering

By

Rohit Raj

A7605212092

Under the guidance of

Mr.Puneet Sharma — Mr.Mohsin Anshari

Lecturer — Trainer

Department of Computer Science & Engineering

Amity School of Engineering and Technology

Amity University Lucknow Campus , 2015

AMITY UNIVERSITY

Department of Computer Science & Engineering

CERTIFICATE

I hereby certify that

a. Rohit Raj, En.no. A7605212092, student of Bachelor of Technology in Computer Science and Engineering (Batch-2012-2016) at Amity School Of Engineering and Technology, Amity University Uttar Pradesh has completed the Project Report on
“E-Smart Solution”, during Summer Break under my supervision.

b. The presented work embodies original research work carried out by the student as per the guidelines given in University Regulations.

c. The research and writing embodied in the thesis are those of the candidate except where due reference is made in the text.

d. I am satisfied that the above candidate’s prima facie, is worthy of examination both in terms of its content and its technical presentations relative to the standards recognised by the university as appropriate for examination.

e. I certify that in accordance with NTCC guidelines, the report does not exceed the prescribed maximum word limit; or prior approval has been sought to go beyond the word limit.

f. Wherever work from other source has been used, all debts (for words, data, arguments and ideas) have been appropriately acknowledged and referenced in accordance with the requirements of NTCC Regulations and Guidelines.

Signature

Mr. Puneet Sharma
Lecturer,
ASET, Amity University

Abstract

The Title of the project is **E-SMART SOLUTION**.

This project was a good way to get into Web Development as Internet is the only form communication that's truly universal and once you get a hold of how to be a part of the World Wide Web, it gets easier to do pretty much anything, be it making money, communicating with someone or just sharing something you've built with the whole world.

This project was just like any other. It required a blueprint and brief planning before diving into the construction and it was easily accomplished with a pencil and a piece of paper. After getting a hold on the web development language, all that was needed was to go step by step and implement that blueprint on a text editor and watch the results screen on a web browser.

This project helped me understand quite a lot about how a web page works, which I wasn't even briefly aware of before. It also developed my interest into the World Wide Web and I'd love to explore it more now that I'm done with this project.

I completed this project at CMC,Lucknow,details of which are given below.

Tata CMC Ltd.

Address with Email & Contact Number of Company: Ground Floor, Shah Tower, Near Chintals House, Station Road, Hussainganj, Lucknow.

Email:- cmcltd.lko@gmail.com

Acknowledgment

Before I proceed further, I would like to spend some time in expressing my gratitude to all those who have been involved in guiding me out during the entire curriculum.

I wish to expresss my most sincere and profound gratitude to **Dr. Deepak Arora**, HOD, computer Science and Engineering department, ASET, Amity University Lucknow, for giving me inspiration and requisite facility by giving me a chance to show my capabilities and also for making me feel comfortable in the strictly professional environment of the college premises.

I am also thankful To **Mr.Puneet Sharma**, Lecturer, Computer Science and Engineering Department, ASET, Amity University Lucknow, for his/her guidance throughout the work with her/his help and suggestions. I am also grateful to her/his for cooperation in providing with all required resources.

I extend special thanks to my friends and family for their constant support .

This work cannot be completed without the help of above mentioned people.

Rohit Raj

Date:29 sept 2015

Table of Contents

CHAPTER 1
Introduction

E-Smart Solution project is the creation of a Website which provides support to users. It is a solutions website with a mission to provide innovative IT services and solutions to satisfy customer needs such as:- Hardware and Software Maintenance.

- Networking Installation, Managing and Troubleshooting.
- Set up, Installation, and Configuration.
- Downloading Drivers and Softwares.

This project has been made by making use of following:

- Java
- Servlets
- JSP
- HTML
- Java Beans
- CSS
- MY SQL

Methodology/ Planning of work:

The Project is divided into 4 main Modules:-

- ➢ Administrator Module
- ➢ Supervisor Module
- ➢ Engineer Module
- ➢ User Module

Software Requiements

- Operating System: Windows xp/7 /8/8.1/10
- Language: Java
- Database: MY SQL
- Tools: Eclipse IDE/Netbeans IDE 8.0
- Technologies used: HTML, XML.
- Server: Apache Tomcat 7.0/Glassfish 4.1

Hardware Requirements

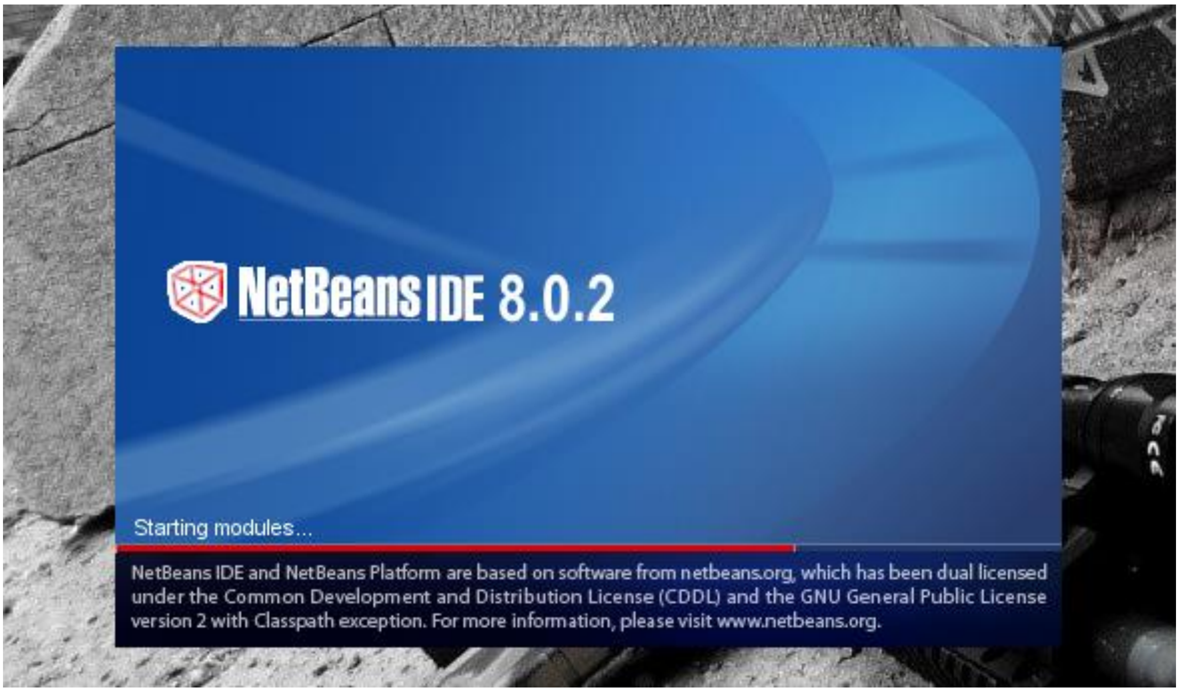
- Processor: Dual Core/i3 or higher
- RAM: 2GB
- Minimum space to execute: 5.0MB

CHAPTER 2

Getting started with Netbeans IDE

NetBeans is a software development platform written in Java. The NetBeans Platform allows applications to be developed from a set of modular software components called *modules.* **NetBeans IDE** is an open-source integrated development environment. NetBeans IDE supports development of all Java application types (Java SE (including JavaFX), Java ME, web, EJB and mobile applications) out of the box.

- Java Development Kit or JDK 1.7 .
- Glassfish 4.1/Apache Tomcat 7.0



Starting modules...

NetBeans IDE and NetBeans Platform are based on software from netbeans.org, which has been dual licensed under the Common Development and Distribution License (CDDL) and the GNU General Public License version 2 with Classpath exception. For more information, please visit www.netbeans.org.

Fig 1.1

CHAPTER 3

INTRODUCTION TO JAVA

3.1 JAVA

Java is a general-purpose computer programming language that is concurrent, class-based, object-oriented and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of computer architecture. As of 2015, Java is one of the most popular programming languages in use, particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by **James Gosling** at Sun Microsystems (which has since been acquired by Oracle Corporation) and released in 1995 as a core component of Sun Microsystems Java platform.

3.2.1 Basic concepts of OOP:

The object oriented programming has been developed with a view to overcome the drawbacks of conventional programming approaches. The OOP approach is based on certain drawbacks that help it attain its goal of overcoming the drawbacks or shortcomings of conventional programming approaches. The general concepts of OOP are given below:

- Data abstraction
- Data encapsulation
- Modularity
- Inheritance
- Polymorphism

Characteristics of Java-

The important characteristics of java are as under:

- Write Once Run Anywhere(WORA)
- Light Weight Code
- Robust
- Security
- Supports Multimedia
- Platform Independent

- **Object Oriented :** In java everything is an Object. Java can be easily extended since it is based on the Object model.

- **Platform independent:** Unlike many other programming languages including C and C++ when Java is compiled, it is not compiled into platform specific machine, rather into platform independent byte code. This byte code is distributed over the web and interpreted by virtual Machine (JVM) on whichever platform it is being run.

- **Simple :**Java is designed to be easy to learn. If you understand the basic concept of OOP java would be easy to master.

- **Secure :** With Java's secure feature it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption.

- **Architectural- neutral :**Java compiler generates an architecture-neutral object file format which makes the compiled code to be executable on many processors, with the presence Java runtime system.

- **Portable :**being architectural neutral and having no implementation dependent aspects of the specification makes Java portable. Compiler and Java is written in ANSI C with a clean portability boundary which is a POSIX subset.

- **Robust :**Java makes an effort to eliminate error prone situations by emphasizing mainly on compile time error checking and runtime checking.

- **Multi-threaded :** With Java's multi-threaded feature it is possible to write programs that can do many tasks simultaneously. This design feature allows developers to construct smoothly running interactive application.

Java Basic Syntax :-

About Java programs, it is very important to keep in mind the following points.

- **Case Sensitivity -** Java is case sensitive which means identifier **Hello** and **hello** would have different meaning in Java.

- **Class Names -** For all class names the first letter should be in Upper Case.

  If several words are used to form a name of the class each inner words first letter should be in Upper Case.

  Example *class MyFirstJavaClass*
- **Method Names -** All method names should start with a Lower Case letter.

  If several words are used to form the name of the method, then each inner word's first letter should be in Upper Case.

  Example *public void myMethodName()*
- **Program FileName -** Name of the program file should exactly match the class name.

- When saving the file you should save it using the class name (Remember java is case sensitive) and append '.java' to the end of the name. (if the file name and the class name do not match your program will not compile).

Example : Assume 'MyFirstJavaProgram' is the class name. Then the file should be saved as*'MyFirstJavaProgram.java'*

- **public static void main(String args[]) -** java program processing starts from the main() method which is a mandatory part of every java program.

First Java Program :-

Let us look at a simple code that would print the words *Hello World*.

```
public class MyFirstJavaProgram {

  /* This is my first java program.

   * This will print 'Hello World' as the output

   */    public static void main(String []args) {

      System.out.println("Hello World"); // prints Hello World

} }
```

Java Objects and Classes :-

Objects in Java:

Let us now look deep into what are objects. If we consider the real-world we can find many objects around us, Cars, Dogs, Humans etc. All these objects have a state and behavior.If we consider a dog then its state is - name, breed, color, and the behavior is - barking, wagging, runningIf you compare the software object with a real world object, they have very similar characteristics.

Software objects also have a state and behavior. A software object's state is stored in fields and behavior is shown via methods.

So in software development methods operate on the internal state of an object and the object-to-object communication is done via methods.

Classes in Java:

A class is a blue print from which individual objects are created.A class can contain any of the following variable types.

- **Local variables .** variables defined inside methods, constructors or blocks are called local variables. The variable will be declared and initialized within the method and the variable will be destroyed when the method has completed.
- **Instance variables .** Instance variables are variables within a class but outside any method. These variables are instantiated when the class is loaded. Instance variables can be accessed from inside any method, constructor or blocks of that particular class.
- **Class variables .** Class variables are variables declared with in a class, outside any method, with the static keyword.

A class can have any number of methods to access the value of various kind of methods. In the above example, barking(), hungry() and sleeping() are methods.

Constructors :- When discussing about classes one of the most important sub topic would be constructors. Every class has a constructor. If we do not explicitly write a constructor for a class the java compiler builds a default constructor for that class.Each time a new object is created at least one constructor will be invoked. The main rule of constructors is that they should have the same name as the class. A class can have more than one constructor.

Java Basic Data types :- Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory.Based on the data type of a variable, the operating system allocates memory and decides what can be stored in the reserved memory. Therefore, by assigning different data types to variables, you can store integers, decimals, or characters in these variables.

There are two data types available in Java:

- Primitive Data Types
- Reference/Object Data Types

Primitive Data Types:

There are eight primitive data types supported by Java. Primitive data types are predefined by the language and named by a key word. Let us now look into detail about the eight primitive data types.

Byte,Short,int,long,float,double,char,Boolean.

Java Variable types :-

In Java, all variables must be declared before they can be used. The basic form of a variable declaration is shown here:

The *type* is one of Java's datatypes. The *identifier* is the name of the variable. To declare more than one variable of the specified type, use a comma-separated list.

Access Modifiers : default, public , protected, private

Non-access Modifiers : final

Java Basic Operators :- Java provides a rich set of operators to manipulate variables. We can divide all the Java operators into the following groups:

- Arithmetic Operators
- Relational Operators
- Bitwise Operators
- Logical Operators
- Assignment Operators
- Misc Operators

Misc Operators

There are few other operators supported by Java Language.

Conditional Operator (? :):

Conditional operator is also known as the ternary operator. This operator consists of three operands and is used to evaluate boolean expressions. The goal of the operator is to decide which value should be assigned to the variable. The operator is written as :

```
variable x = (expression) ? value if true : value if false
```

Java Loop Control :-

There may be a sitution when we need to execute a block of code several number of times, and is often referred to as a loop.

Java has very flexible three looping mechanisms. You can use one of the following three loops:

- while Loop
- do...while Loop
- for Loop

The break Keyword:

The *break* keyword is used to stop the entire loop. The break keyword must be used inside any loop or a switch statement.

The break keyword will stop the execution of the innermost loop and start executing the next line of code after the block.

The continue Keyword:

The *continue* keyword can be used in any of the loop control structures. It causes the loop to immediately jump to the next iteration of the loop.

- In a for loop, the continue keyword causes flow of control to immediately jump to the update statement.
- In a while loop or do/while loop, flow of control immediately jumps to the Boolean expression.

Java Decision Making :-

There are two types of decision making statements in Java. They are:

- if statements
- switch statements

The if Statement:

An if statement consists of a Boolean expression followed by one or more statements.

The if...else Statement:

An if statement can be followed by an optional *else* statement, which executes when the Boolean expression is false.

The switch Statement:

A *switch* statement allows a variable to be tested for equality against a list of values. Each value is called a case, and the variable being switched on is checked for each case.

Java Strings :-

Strings, which are widely used in Java programming, are a sequence of characters. In the Java programming language, strings are objects.

The Java platform provides the String class to create and manipulate strings.

Creating Strings:

The most direct way to create a string is to write:

```
String greeting = "Hello world!";
```

Whenever it encounters a string literal in your code, the compiler creates a String object with its valuein this case, "Hello world!'.

As with any other object, you can create String objects by using the new keyword and a constructor. The String class has eleven constructors that allow you to provide the initial value of the string using different sources, such as an array of characters.

Java Arrays :-

Java provides a data structure, the **array**, which stores a fixed-size sequential collection of elements of the same type. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables of the same type.

Instead of declaring individual variables, such as number0, number1, ..., and number99, you declare one array variable such as numbers and use numbers[0], numbers[1], and ..., numbers[99] to represent individual variables.

Java Methods :-

A Java method is a collection of statements that are grouped together to perform an operation. When you call the System.out.println method, for example, the system actually executes several statements in order to display a message on the console.

Objects in Java:

Let us now look deep into what are objects. If we consider the real-world we can find many objects around us, Cars, Dogs, Humans etc. All these objects have a state and behavior.If we consider a dog then its state is - name, breed, color, and the behavior is - barking, wagging, runningIf you compare the software object with a real world object, they have very similar characteristics.

Software objects also have a state and behavior. A software object's state is stored in fields and behavior is shown via methods.

So in software development methods operate on the internal state of an object and the object-to-object communication is done via methods.

Classes in Java:

A class is a blue print from which individual objects are created.A class can contain any of the following variable types.

- **Local variables .** variables defined inside methods, constructors or blocks are called local variables. The variable will be declared and initialized within the method and the variable will be destroyed when the method has completed.
- **Instance variables .** Instance variables are variables within a class but outside any method. These variables are instantiated when the class is loaded. Instance variables can be accessed from inside any method, constructor or blocks of that particular class.
- **Class variables .** Class variables are variables declared with in a class, outside any method, with the static keyword.

A class can have any number of methods to access the value of various kind of methods. In the above example, barking(), hungry() and sleeping() are methods.

Constructors :- When discussing about classes one of the most important sub topic would be constructors. Every class has a constructor. If we do not explicitly write a constructor for a class the java compiler builds a default constructor for that class.Each time a new object is created at least one constructor will be invoked. The main rule of constructors is that they should have the same name as the class. A class can have more than one constructor.

Java Basic Data types :- Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory.Based on the data type of a variable, the operating system allocates memory and decides what can be stored in the reserved memory. Therefore, by assigning different data types to variables, you can store integers, decimals, or characters in these variables.

There are two data types available in Java:

- Primitive Data Types
- Reference/Object Data Types

Primitive Data Types:

There are eight primitive data types supported by Java. Primitive data types are predefined by the language and named by a key word. Let us now look into detail about the eight primitive data types.

Byte,Short,int,long,float,double,char,Boolean.

Java Variable types :-

In Java, all variables must be declared before they can be used. The basic form of a variable declaration is shown here:

The *type* is one of Java's datatypes. The *identifier* is the name of the variable. To declare more than one variable of the specified type, use a comma-separated list.

Access Modifiers : default, public , protected, private

Non-access Modifiers : final

Java Basic Operators :- Java provides a rich set of operators to manipulate variables. We can divide all the Java operators into the following groups:

- Arithmetic Operators
- Relational Operators
- Bitwise Operators
- Logical Operators
- Assignment Operators
- Misc Operators

Misc Operators

There are few other operators supported by Java Language.

Conditional Operator (? :):

Conditional operator is also known as the ternary operator. This operator consists of three operands and is used to evaluate boolean expressions. The goal of the operator is to decide which value should be assigned to the variable. The operator is written as :

```
variable x = (expression) ? value if true : value if false
```

Java Loop Control :-

There may be a sitution when we need to execute a block of code several number of times, and is often referred to as a loop.

Java has very flexible three looping mechanisms. You can use one of the following three loops:

- while Loop
- do...while Loop
- for Loop

The break Keyword:

The *break* keyword is used to stop the entire loop. The break keyword must be used inside any loop or a switch statement.

The break keyword will stop the execution of the innermost loop and start executing the next line of code after the block.

The continue Keyword:

The *continue* keyword can be used in any of the loop control structures. It causes the loop to immediately jump to the next iteration of the loop.

- In a for loop, the continue keyword causes flow of control to immediately jump to the update statement.
- In a while loop or do/while loop, flow of control immediately jumps to the Boolean expression.

Java Decision Making :-

There are two types of decision making statements in Java. They are:

- if statements
- switch statements

The if Statement:

An if statement consists of a Boolean expression followed by one or more statements.

The if...else Statement:

An if statement can be followed by an optional *else* statement, which executes when the Boolean expression is false.

The switch Statement:

A *switch* statement allows a variable to be tested for equality against a list of values. Each value is called a case, and the variable being switched on is checked for each case.

Java Strings :-

Strings, which are widely used in Java programming, are a sequence of characters. In the Java programming language, strings are objects.

The Java platform provides the String class to create and manipulate strings.

Creating Strings:

The most direct way to create a string is to write:

```
String greeting = "Hello world!";
```

Whenever it encounters a string literal in your code, the compiler creates a String object with its valuein this case, "Hello world!'.

As with any other object, you can create String objects by using the new keyword and a constructor. The String class has eleven constructors that allow you to provide the initial value of the string using different sources, such as an array of characters.

Java Arrays :-

Java provides a data structure, the **array**, which stores a fixed-size sequential collection of elements of the same type. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables of the same type.

Instead of declaring individual variables, such as number0, number1, ..., and number99, you declare one array variable such as numbers and use numbers[0], numbers[1], and ..., numbers[99] to represent individual variables.

Java Methods :-

A Java method is a collection of statements that are grouped together to perform an operation. When you call the System.out.println method, for example, the system actually executes several statements in order to display a message on the console.

INTRODUCTION TO SERVLET

3.2 Servlet

A **servlet** is a Java programming language class that is used to extend the capabilities of servers that host applications accessed by means of a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by web servers. For such applications, Java Servlet technology defines HTTP-specific servlet classes.

- 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.

3.2 Advantages of Servlet

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

3.2 Life Cycle of Servlet

- Servlet class is loaded.
- Servlet instance is created.
- init method is invoked.
- service method is invoked.
- destroy method is invoked.

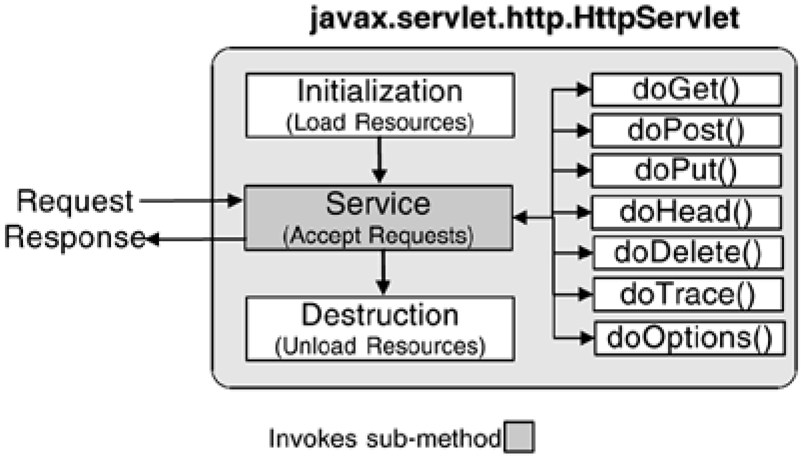


Fig.1.2

3.2 Servlet API

The javax.servlet and javax.servlet.http packages represent interfaces and classes for servlet api.

The **javax.servlet** package contains many interfaces and classes that are used by the servlet or web container. These are not specific to any protocol.

Interface in javax.servlet Package

There are many interfaces in javax.servlet.http package. They are as follows:

- HttpServletRequest
- HttpServletResponse
- HttpSession
- HttpSessionListener
- HttpSessionAttributeListener
- HttpSessionBindingListener
- HttpSessionActivationListener
- HttpSessionContext

Classes in javax.servlet.http package

There are many classes in javax.servlet.http package. They are as follows:

- HttpServlet
- Cookie
- HttpServletRequestWrapper
- HttpServletResponseWrapper
- HttpSessionEvent

- HttpSessionBindingEvent
- HttpUtils

Servlet Types

There are two main servlet types, generic and HTTP:

- **Generic servlets**

GenericServlet class implements **Servlet**, **ServletConfig** and**Serializable** interfaces. It provides the implementaion of all the methods of these interfaces except the service method.GenericServlet class can handle any type of request so it is protocol-independent.

- Extend javax.servlet.GenericServlet.
- Are protocol independent. They contain no inherent HTTP support or any other transport protocol.

Methods of GenericServlet class

There are many methods in GenericServlet class. They are as follows:

- **public void init(ServletConfig config)** is used to initialize the servlet.
- **public abstract void service(ServletRequest request, ServletResponse response)** provides service for the incoming request. It is invoked at each time when user requests for a servlet.
- **public void destroy()** is invoked only once throughout the life cycle and indicates that servlet is being destroyed.
- **public ServletConfig getServletConfig()** returns the object of ServletConfig.
- **public String getServletInfo()** returns information about servlet such as writer, copyright, version etc.
- **public void init()** it is a convenient method for the servlet programmers, now there is no need to call super.init(config)
- **public ServletContext getServletContext()** returns the object of ServletContext.

```
import java.io.*;
import javax.servlet.*;
public class First extends GenericServlet{
public void service(ServletRequest req,ServletResponse res)
throws IOException,ServletException{

res.setContentType("text/html");

PrintWriter out=res.getWriter();
out.print("<html><body>");
out.print("<b>hello generic servlet</b>");
out.print("</body></html>");

}
}
```

- **HTTP Servlets**

The HttpServlet class extends the GenericServlet class and implements Serializable interface. It provides http specific methods such as doGet, doPost, doHead, doTrace etc

- Extend javax.servlet.HttpServlet.
- Have built-in HTTP protocol support and are more useful in a Sun Java System Web Server environment.

For both servlet types, you implement the constructor method init() and the destructor method destroy() to initialize or deallocate resources.

All servlets must implement a service() method, which is responsible for handling servlet requests. For generic servlets, simply override the service method to provide routines for handling requests. HTTP servlets provide a service method that automatically routes the request to another method in the servlet based on which HTTP transfer method is used. So, for HTTP servlets, overridedoPost() to process POST requests, doGet() to process GET requests, and so on.

Methods of HttpServlet class

There are many methods in HttpServlet class. They are as follows:

- **public void service(ServletRequest req,ServletResponse res)** dispatches the request to the protected service method by converting the request and response object into http type.
- **protected void service(HttpServletRequest req, HttpServletResponse res)** receives the request from the service method, and dispatches the request to the doXXX() method depending on the incoming http request type.
- **protected void doGet(HttpServletRequest req, HttpServletResponse res)** handles the GET request. It is invoked by the web container.
- **protected void doPost(HttpServletRequest req, HttpServletResponse res)** handles the POST request. It is invoked by the web container.
- **protected void doHead(HttpServletRequest req, HttpServletResponse res)** handles the HEAD request. It is invoked by the web container.
- **protected void doOptions(HttpServletRequest req, HttpServletResponse res)** handles the OPTIONS request. It is invoked by the web container.
- **protected void doPut(HttpServletRequest req, HttpServletResponse res)** handles the PUT request. It is invoked by the web container.
- **protected void doTrace(HttpServletRequest req, HttpServletResponse res)** handles the TRACE request. It is invoked by the web container.
- **protected void doDelete(HttpServletRequest req, HttpServletResponse res)** handles the DELETE request. It is invoked by the web container.

```
import javax.servlet.http.*;
import javax.servlet.*;
import java.io.*;
public class DemoServlet extends HttpServlet{
public void doGet(HttpServletRequest req,HttpServletResponse res)
throws ServletException,IOException
{
res.setContentType("text/html");
PrintWriter pw=res.getWriter();


pw.println("<html><body>");
pw.println("Welcome to servlet");
pw.println("</body></html>");

pw.close();
}}
```

INTRODUCTION TO JSP

3.2JSP

JavaServer Pages (JSP) technology allows you to easily create web content that has both static and dynamic components. JSP technology makes available all the dynamic capabilities of Java Servlet technology but provides a more natural approach to creating static content

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.

3.2 Advantages of JSP

- **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.

- **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.

- **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.

- **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.

3.2 Life Cycle of JSP

- 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).
- Reqeust processing (_jspService() method is invoked by the container).
- Destroy (jspDestroy() method is invoked by the container)

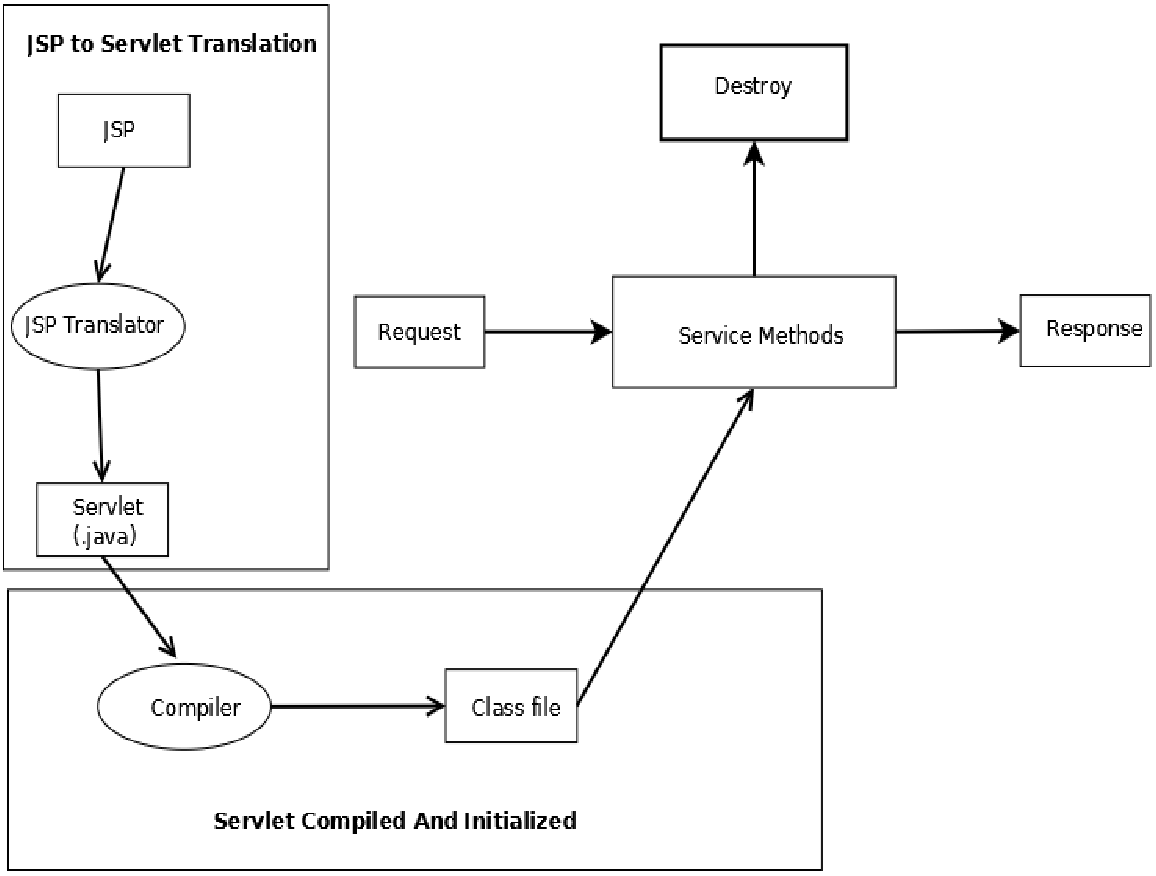
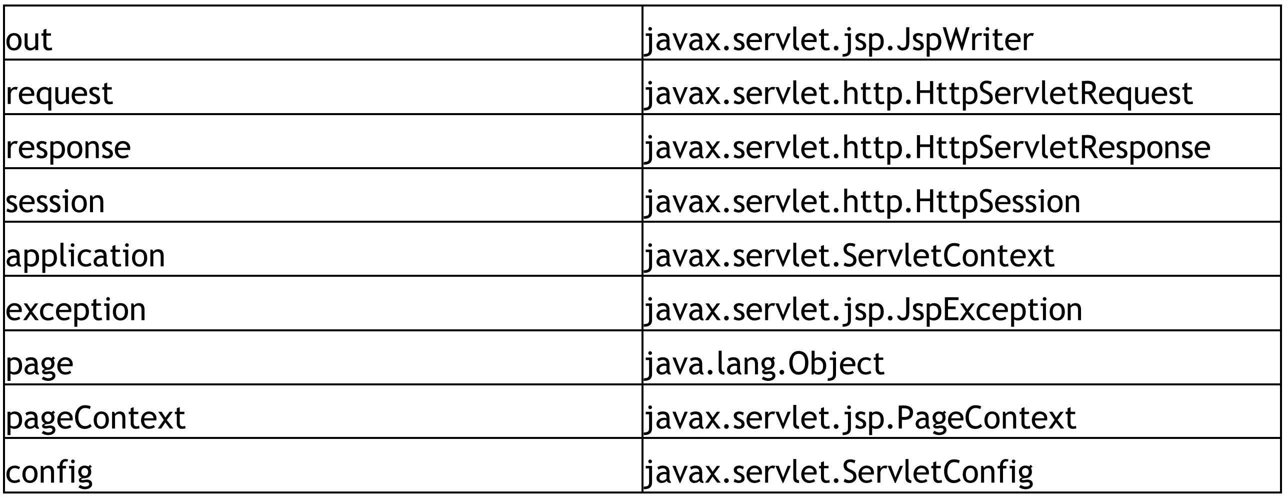


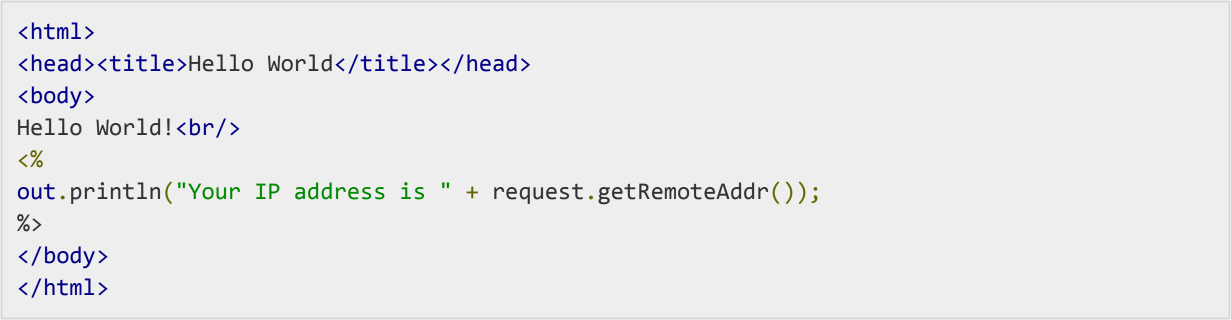
Fig 1.3

JSP Implicit Objects



out	javax.servlet.jsp.JspWriter
request	javax.servlet.http.HttpServletRequest
response	javax.servlet.http.HttpServletResponse
session	javax.servlet.http.HttpSession
application	javax.servlet.ServletContext
exception	javax.servlet.jsp.JspException
page	java.lang.Object
pageContext	javax.servlet.jsp.PageContext
config	javax.servlet.ServletConfig

Fig 1.4



```
<html>
<head><title>Hello World</title></head>
<body>
Hello World!<br/>
<%
out.println("Your IP address is " + request.getRemoteAddr());
%>
</body>
</html>
```

Chapter 4

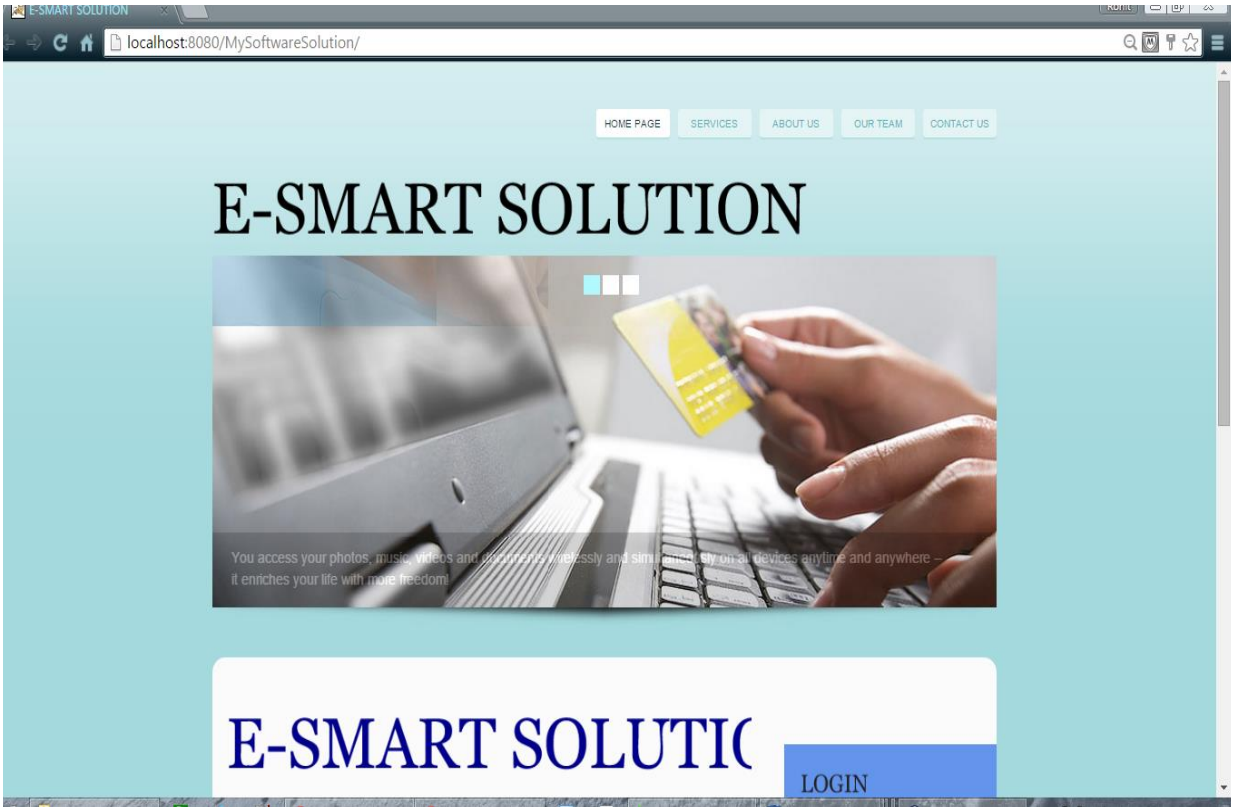
E-SMART SOLUTION

E-Smart Solution project is the creation of a Website which provides support to users. It is a solutions website with a mission to provide innovative IT services and solutions to satisfy customer needs such as:- Hardware and Software Maintenance.

- ➢ **E-SMART SOLUTION home page:**

The project is a website that provides free software and services to the clients.

- It is created on a jsp page .
- Java Servlets is used for coding purposes.
    - HOMEPAGE:
        - ➢ Login form
            - ✓ User
            - ✓ Engineer
            - ✓ Supervisor
            - ✓ Administrator
            - ✓ Text boxes and Buttons
        - ➢ Registration
        - ➢ Details of Services and Support Provided



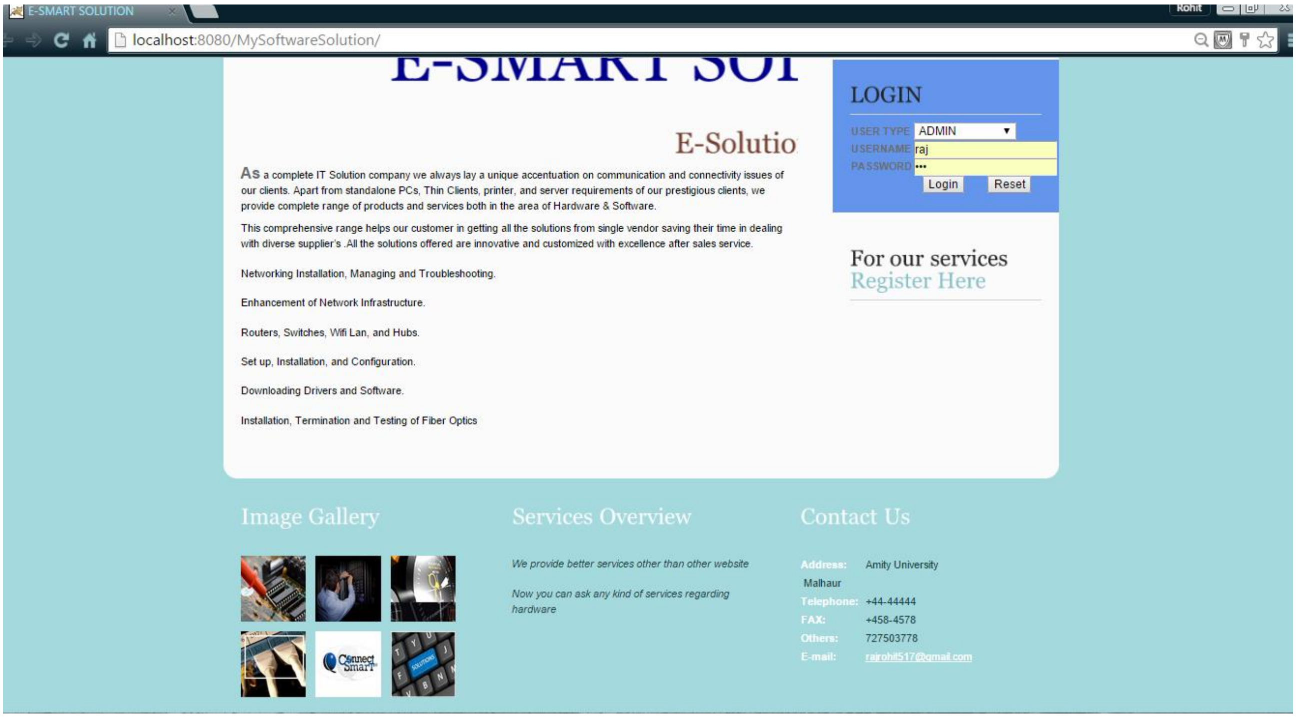
E-SMART SOLUTION

You access your photos, music, videos and documents wirelessly and simultaneously on all devices anytime and anywhere – it enriches your life with more freedom!

E-SMART SOLUTI(

LOGIN

Fig 1.5



E-SMART SOL

E-Solutio

As a complete IT Solution company we always lay a unique accentuation on communication and connectivity issues of our clients. Apart from standalone PCs, Thin Clients, printer, and server requirements of our prestigious clients, we provide complete range of products and services both in the area of Hardware & Software.

This comprehensive range helps our customer in getting all the solutions from single vendor saving their time in dealing with diverse supplier's .All the solutions offered are innovative and customized with excellence after sales service.

Networking Installation, Managing and Troubleshooting.

Enhancement of Network Infrastructure.

Routers, Switches, Wifi Lan, and Hubs.

Set up, Installation, and Configuration.

Downloading Drivers and Software.

Installation, Termination and Testing of Fiber Optics

LOGIN

USER TYPE ADMIN ▼
USERNAME raj
PASSWORD •••
Login Reset

For our services
Register Here

Image Gallery

Services Overview

We provide better services other than other website

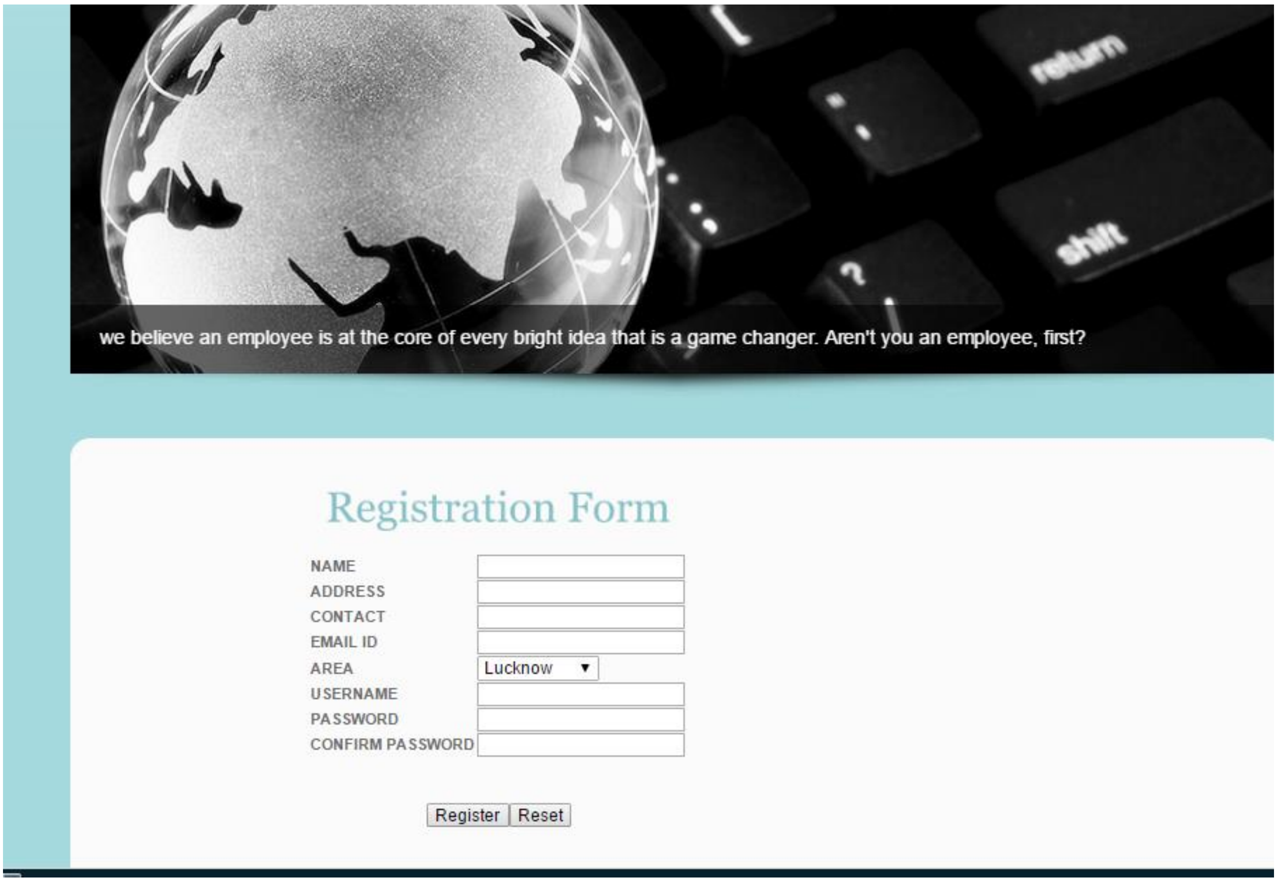
Now you can ask any kind of services regarding hardware

Contact Us

Address: Amity University
Malhaur
Telephone: +44-44444
FAX: +458-4578
Others: 727503778
E-mail: rajrohit517@gmail.com

Fig 1.6

- For new user a **Registration form** is created. The details will be stored in a database and user will be provided with a verification code which will be sent on the email provided by the user. After verification user can have access to our services.



we believe an employee is at the core of every bright idea that is a game changer. Aren't you an employee, first?

Registration Form

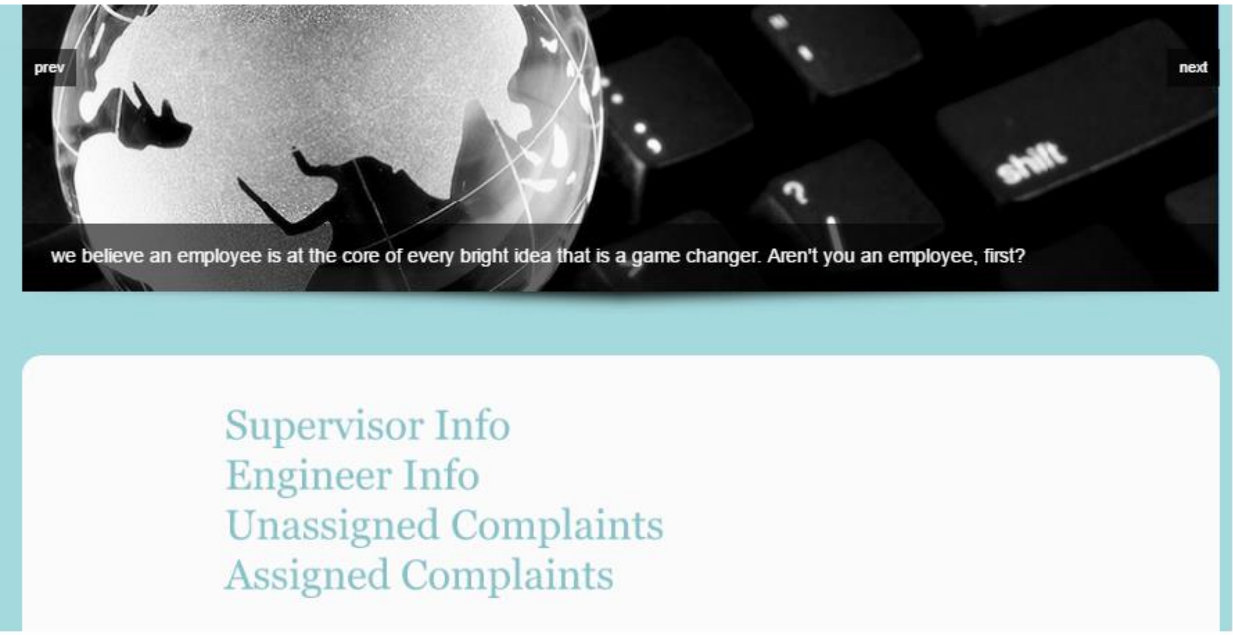
NAME	
ADDRESS	
CONTACT	
EMAIL ID	
AREA	Lucknow ▾
USERNAME	
PASSWORD	
CONFIRM PASSWORD	

Fig 1.6

Module :1

➢ ADMIN MODULE

- SUPERVISOR INFORMATION
- ADD SUPERVISOR
- UPDATE DETAILS OF SUPERVISOR
- ENGINNER INFORMATION
- ADD ENGINEER
- UPDATE DETAILS OF ENGINEER
- CAN SEE AND ASSIGN COMPLAINTS
- HAVE FULL ACCESS TO ALL UNASSIGNED AND ASSIGNED COMPLAINTS
- CAN SEE USER DETAILS

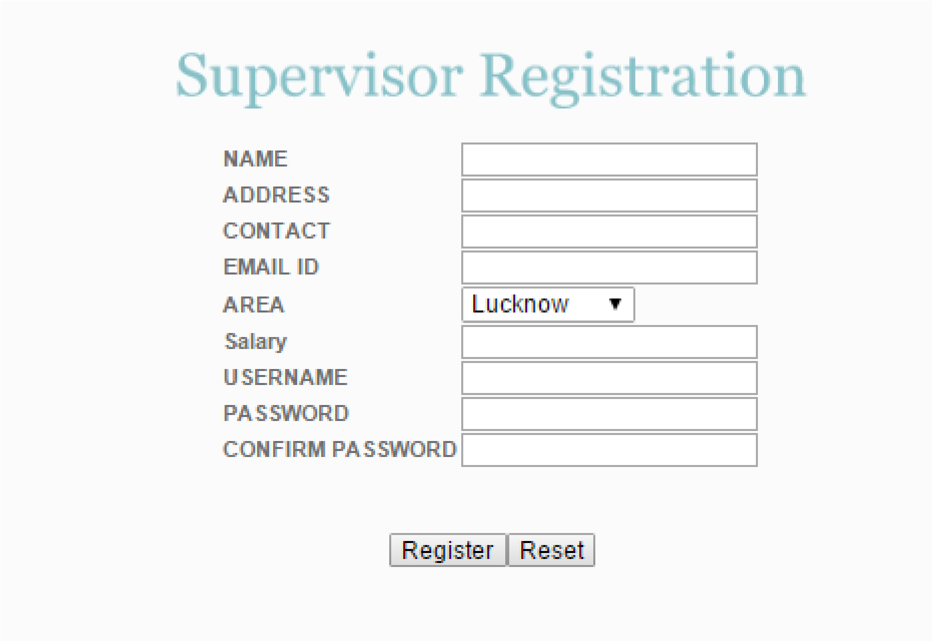


we believe an employee is at the core of every bright idea that is a game changer. Aren't you an employee, first?

Supervisor Info
Engineer Info
Unassigned Complaints
Assigned Complaints

Fig 1.7

➢ SUPERVISOR Registration



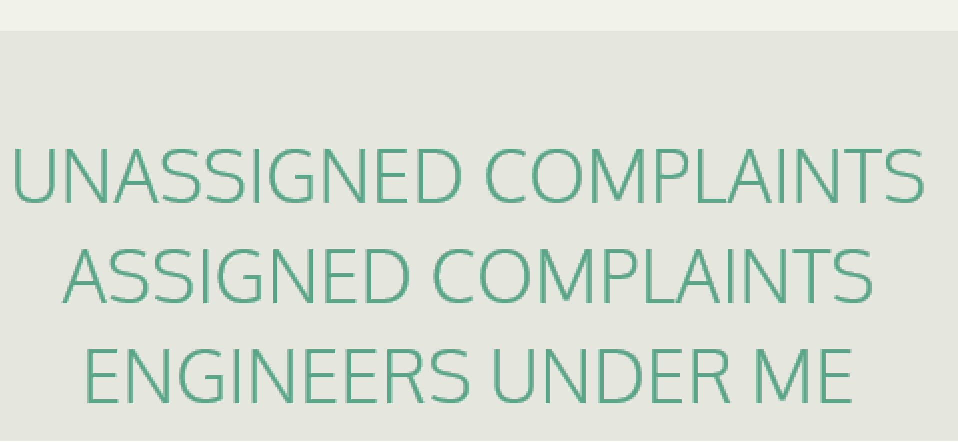
Supervisor Registration

NAME	
ADDRESS	
CONTACT	
EMAIL ID	
AREA	Lucknow ▼
Salary	
USERNAME	
PASSWORD	
CONFIRM PASSWORD	

Fig 1.8

Module :2

- **SUPERVISORHOME**
  - Engineer under me
  - Unassigned complaints
  - Assigned complaints



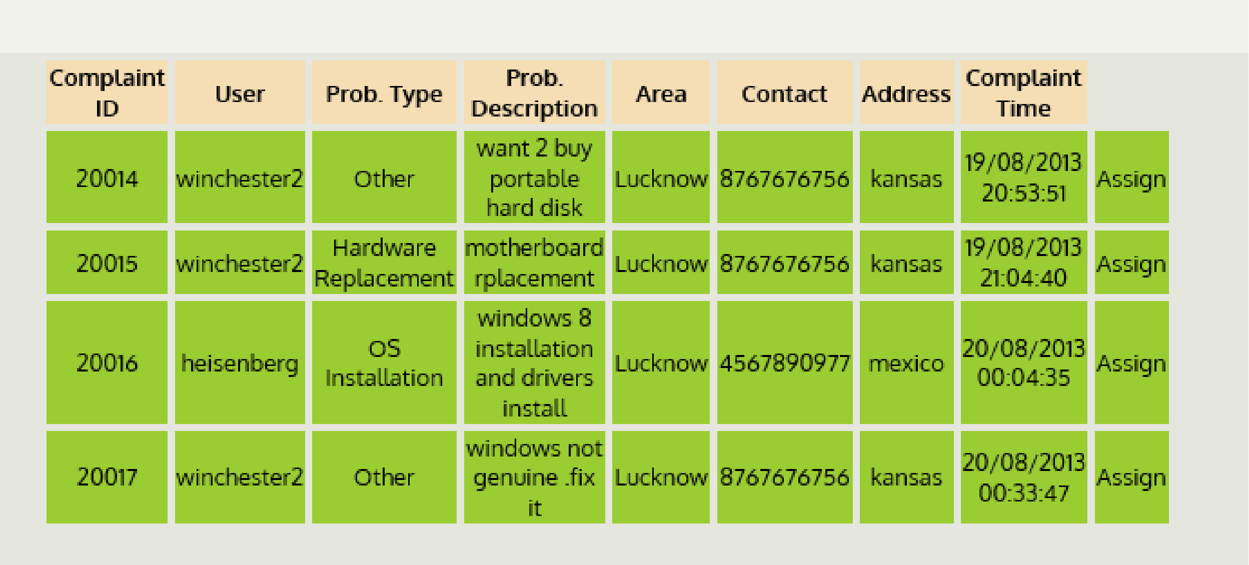
UNASSIGNED COMPLAINTS

ASSIGNED COMPLAINTS

ENGINEERS UNDER ME

Fig 1.

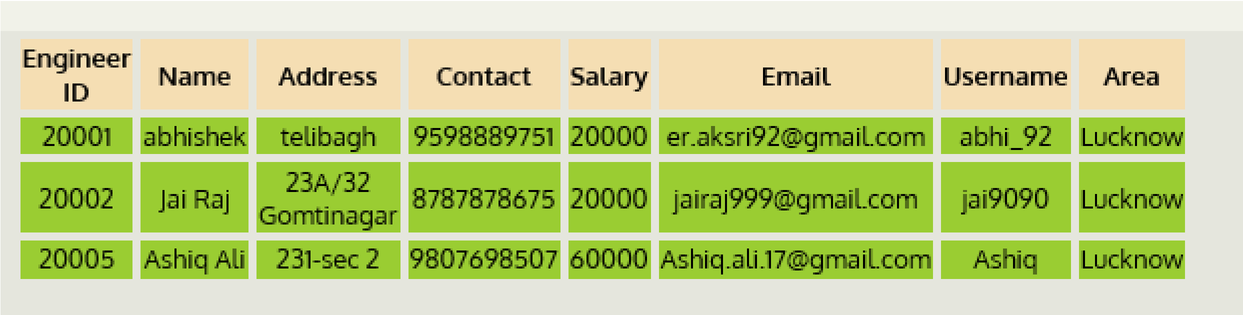
- Supervisor can see the details of the engineers,edit ,update them.Any modification,updation required can also be done by the supervisor.
- Supervisor can assign tasks to engineers under him.
- Can also see the details of the complaint which includes user details like area,name,date of complaint, status of the complaint.



Complaint ID	User	Prob. Type	Prob. Description	Area	Contact	Address	Complaint Time	
20014	winchester2	Other	want 2 buy portable hard disk	Lucknow	8767676756	kansas	19/08/2013 20:53:51	Assign
20015	winchester2	Hardware Replacement	motherboard rplacement	Lucknow	8767676756	kansas	19/08/2013 21:04:40	Assign
20016	heisenberg	OS Installation	windows 8 installation and drivers install	Lucknow	4567890977	mexico	20/08/2013 00:04:35	Assign
20017	winchester2	Other	windows not genuine .fix it	Lucknow	8767676756	kansas	20/08/2013 00:33:47	Assign

Fig 1.10

Unassigned complaints.



Engineer ID	Name	Address	Contact	Salary	Email	Username	Area
20001	abhishek	telibagh	9598889751	20000	er.aksri92@gmail.com	abhi_92	Lucknow
20002	Jai Raj	23A/32 Gomtinagar	8787878675	20000	jairaj999@gmail.com	jai9090	Lucknow
20005	Ashiq Ali	231-sec 2	9807698507	60000	Ashiq.ali.17@gmail.com	Ashiq	Lucknow

Fig 1.11

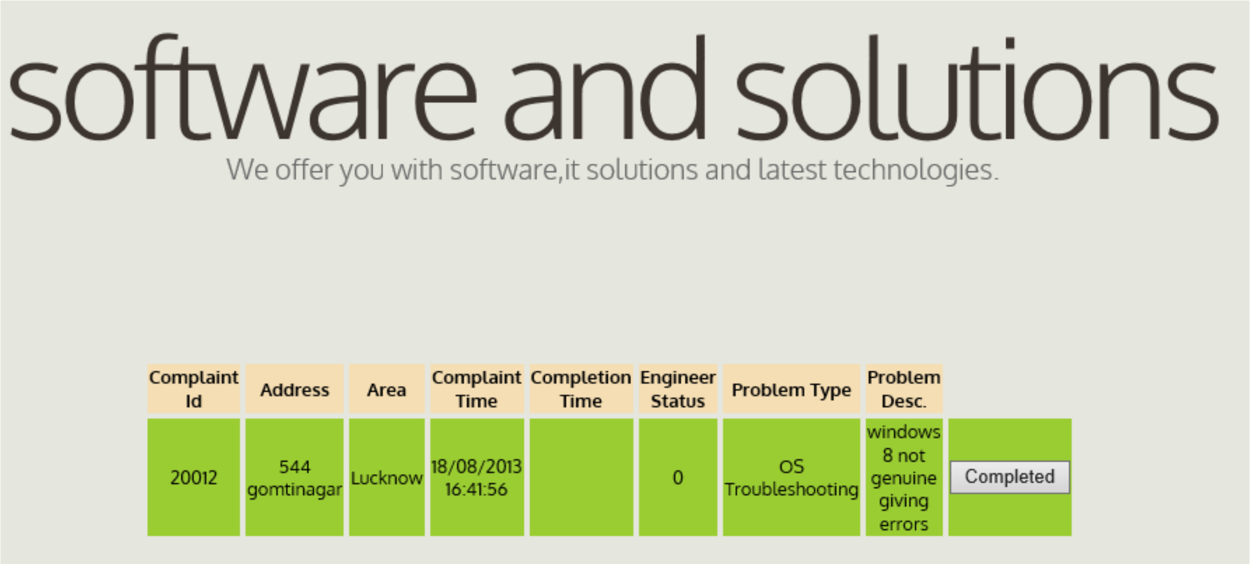
Details of engineers under supervisor.

.

Module :3

➢ ENGINEER_HOME

- CONTENTS
- MY ASSIGNMENTS
- In my assignment page the tasks given to the engineer are shown.
- COMPLETED ASSIGNMENTS
- It shows the details of the assignments that are completed by the engineer.

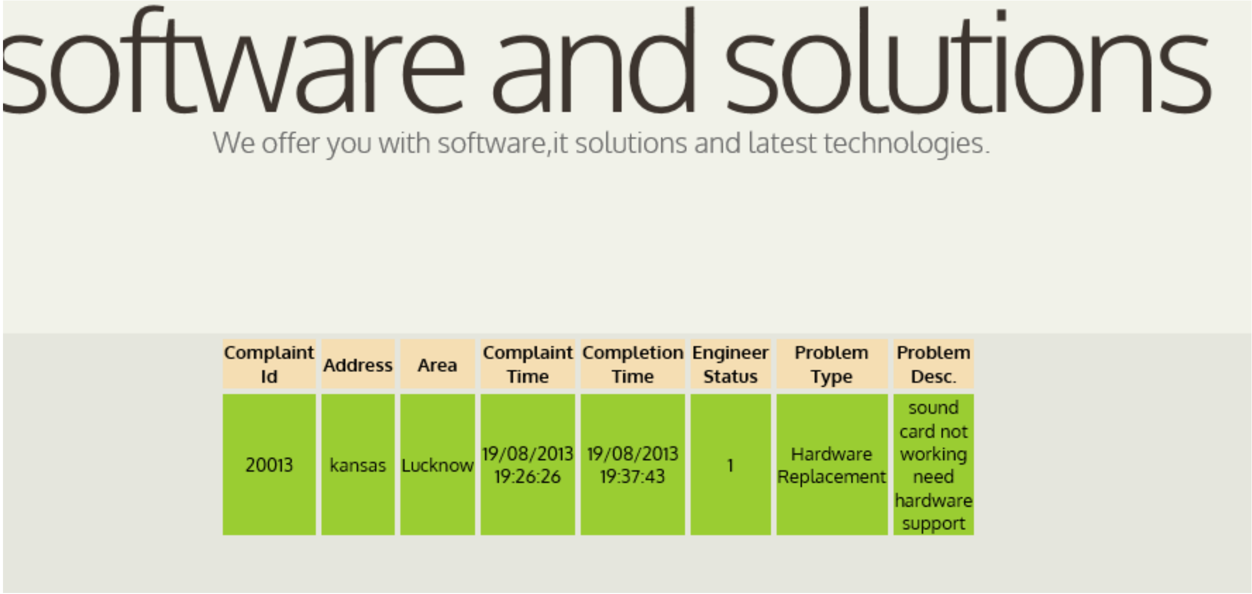


software and solutions

We offer you with software,it solutions and latest technologies.

Complaint Id	Address	Area	Complaint Time	Completion Time	Engineer Status	Problem Type	Problem Desc.	
20012	544 gomtinagar	Lucknow	18/08/2013 16:41:56		0	OS Troubleshooting	windows 8 not genuine giving errors	Completed

Fig 1.12



software and solutions

We offer you with software,it solutions and latest technologies.

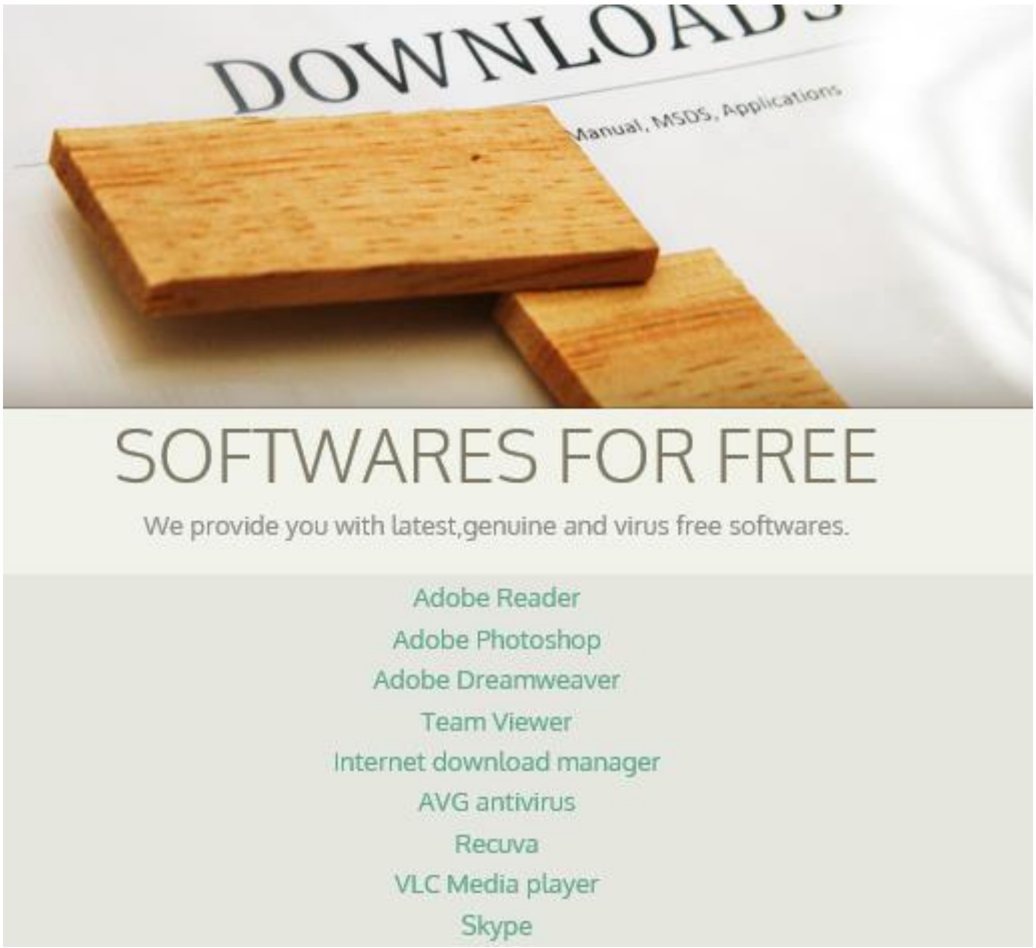
Complaint Id	Address	Area	Complaint Time	Completion Time	Engineer Status	Problem Type	Problem Desc.
20013	kansas	Lucknow	19/08/2013 19:26:26	19/08/2013 19:37:43	1	Hardware Replacement	sound card not working need hardware support

Fig 1.13

Module :4

➢USERHOME

- FREE SOFTWARES FOR USER
- CAN MAKE COMPLAINTS
- TECHNICAL SUPPORT PROVIDED

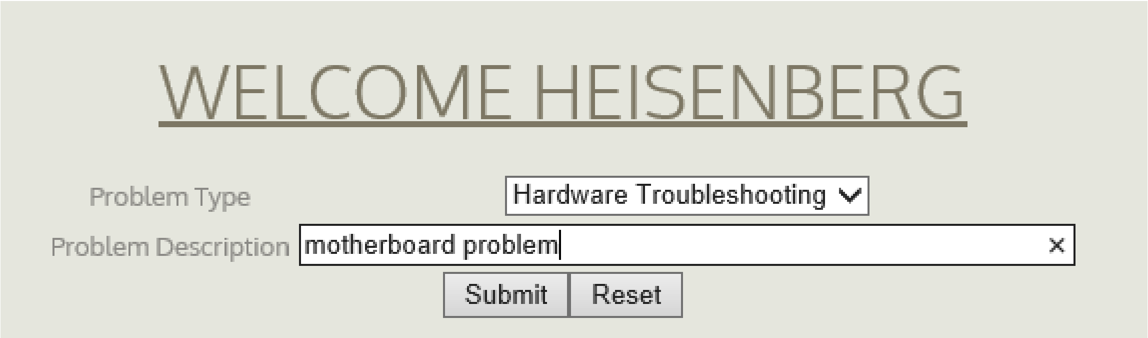


SOFTWARES FOR FREE

We provide you with latest,genuine and virus free softwares.

Adobe Reader
Adobe Photoshop
Adobe Dreamweaver
Team Viewer
Internet download manager
AVG antivirus
Recuva
VLC Media player
Skype

Fig.1.14



WELCOME HEISENBERG

Problem Type: Hardware Troubleshooting

Problem Description: motherboard problem

Fig.1.15

Chapter 5

Conclusion

The project titled “E-SMART SOLUTION” is completed with the help of my trainer and internal guide and a lot of hard work. The web application is running effectively without any problems but it can be used .By studying java, i gained a lot of knowledge that is very helpful for me and it will be helpful for me in the future. As java is a very good language but it is not easy to learn so I have to learn a lot of things, techniques that I do not know and I will try to learn and use them too.

Summary of project:

1. WebSite is running well

REFERENCES

➢ Books Referred

- Head First Java - Bert Bates, Kathy Sierra
- Head First Servlets and JSP--Kathy Sierra

➢ Websites referred

- www.javatpoint.com
- www.w3school.com
- www.wikipedia.com/Servlets
- www.wikipedia.com/jsp