



**globsyn**   
*Taking People To The Next Level*

Umesh Kumar Sharma

Globsyn Skills, (



alt Lake, Kolkata – 700091

Document sign date : Jul 11, 2015

# MNREGA Employment Management

## Group Members:

AMIT KUMAR TIWARI, GOVT COLLEGE OF ENGINEERING AND CERAMIC TECHNOLOGY, 121130110043 OF 2012-13  
SUBHAM GUPTA, GOVT COLLEGE OF ENGINEERING AND CERAMIC TECHNOLOGY, 121130110078 OF 2012-13  
BHUMIKA MEHTA, HERITAGE INSTITUTE OF TECHNOLOGY, 131260110139 OF 2013-14  
DIPTA DAS CHOWDHURY, HERITAGE INSTITUTE OF TECHNOLOGY, 131260110141 OF 2013-14  
SNIGDHA JAIN, HERITAGE INSTITUTE OF TECHNOLOGY, 131260110178 OF 2013-14

Umesh Kumar Sharma



## Table of Contents

Acknowledgement.....	4
Project Objective .....	5
Project Scope.....	6
Requirement Specification .....	7
Database Design.....	8
Application Work Flow .....	9
Screenshots .....	10
Future Scope of Improvements.....	11
Code.....	12
Certificate.....	13

Umesh Kumar Sharma



## Acknowledgement

I take this opportunity to express my profound gratitude and deep regards to my faculty Mr.Umesh Kumar Sharma for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him/her time to time shall carry me a long way in the journey of life on which I am about to embark.

I am obliged to my project team members for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

**AMIT KUMAR TIWARI  
BHUMIKA MEHTA  
DIPTA DAS CHOWDHURY  
SUBHAM GUPTA  
SNIGDHA JAIN**

Umesh Kumar Sharma



# PROJECT OBJECTIVE

Main objective of this project is to manage employments offered through this scheme .

There are two users of this system : a) BDO (Block Development Officer) b) Gram Panchayat member  
System works in the following way:

A project is created by BDO . Project is where people are assigned to . Before assigning anybody to a project(done by Gram Panchayat member) his/her data must be inserted into the system .

## Functions of BDO

- Login into their account
- Create a project
- See List of project
- See details of a project
- See list of people working on that project

## Function of Gram Panchayat member

- Login into their account
- Create a new employee
- Assign employee to a project
- View total number of days employee worked in a project



# Project Scope

- **Security and privacy-** The sheer volume of information that governments maintain on individuals makes privacy issues extremely important. Since NREGA application requires strong authentication processes, possibly involving multiple government agencies, it is imperative to offer authentication as a government-managed service in the NREGA infrastructure.
- **Properly layered structure-** Software is in accordance with guidelines of scheme which follows a layered approach. Each layer is identified and managed according to the stakeholders needs so that proper management and analysis can be done.
- **Capturing Data at grass root level-** Software facilitates entry of Works and workers at Gram Panchayat level.
- **User Friendly** - Software is designed according to requirement of each stakeholder, and prerequisite for using this software is that a person must know Computer.

Umesh Kumar Sharma



# Requirement Specification

- Java-

Java is a pure object-oriented language. Java was designed to be easy for professional programmer to learn and use efficiently. Java inherits the C/C++ syntax and the object-oriented features of C++. Java can be used to create two types of programs- applications and applets.

- Java Server Pages

Java Server Pages (JSP) is a technology based on the Java language and enables the development of dynamic web sites. JSP was developed by Sun. JSP files are HTML files with special Tags containing Java source code that provide the dynamic content.

- SCRIPLETS

Many developers believe custom tags, rather than scriptlets or expressions, should be used in JSP pages. The rationale is:  
Scriptlets mix logic with presentation.  
Scriptlets break the separation of roles.

- WILDFLY- **WildFly**, formerly known as JBoss AS, or simply JBoss, is an application **server** authored by JBoss, now developed by Red Hat. **WildFly** is written in Java,

  
Umesh Kumar Sharma



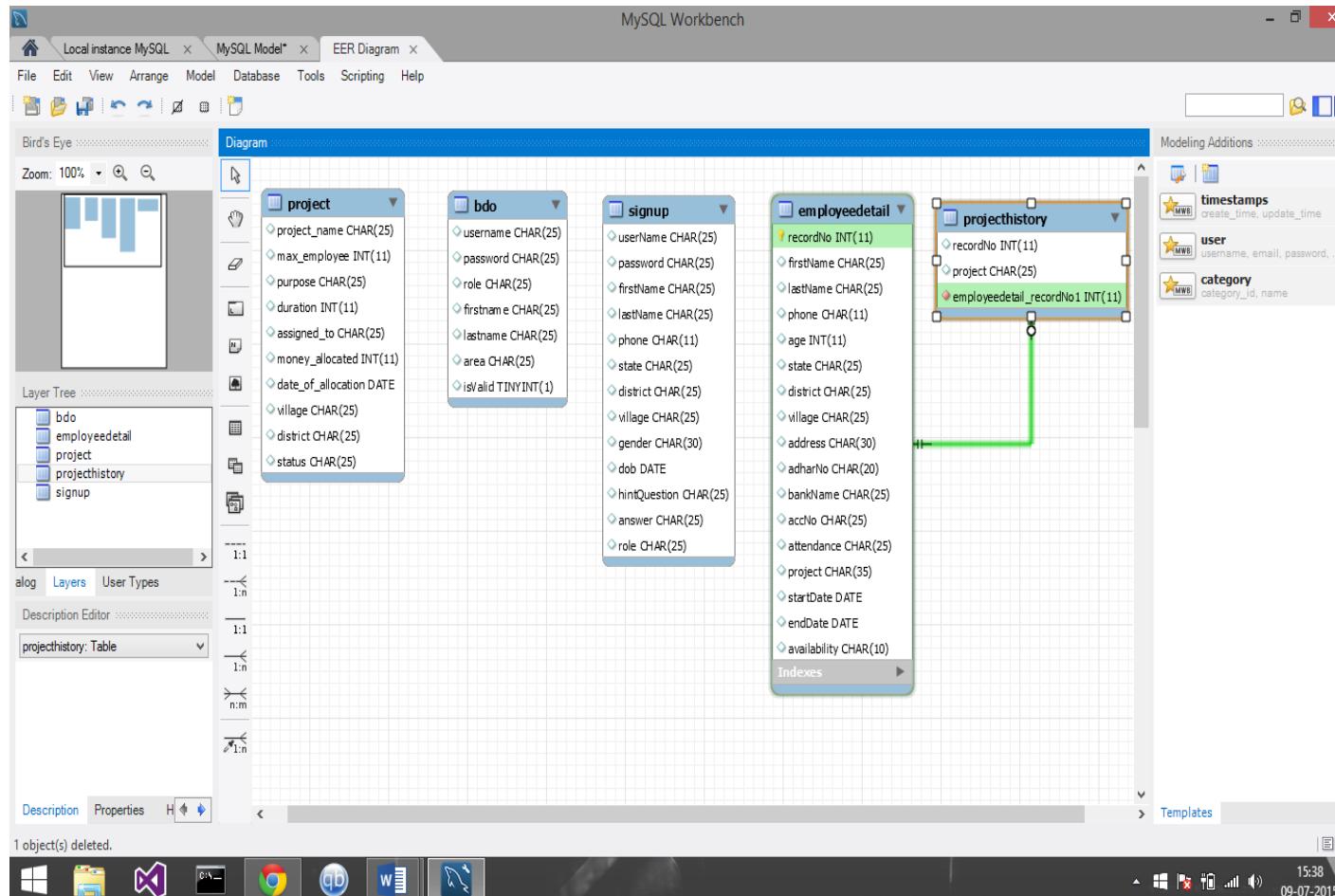
and implements the Java Platform, Enterprise Edition (Java EE) specification.

- **MYSQL-** is a freely available open source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL). SQL is the most popular language for adding, accessing and managing content in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use.

## Database Design

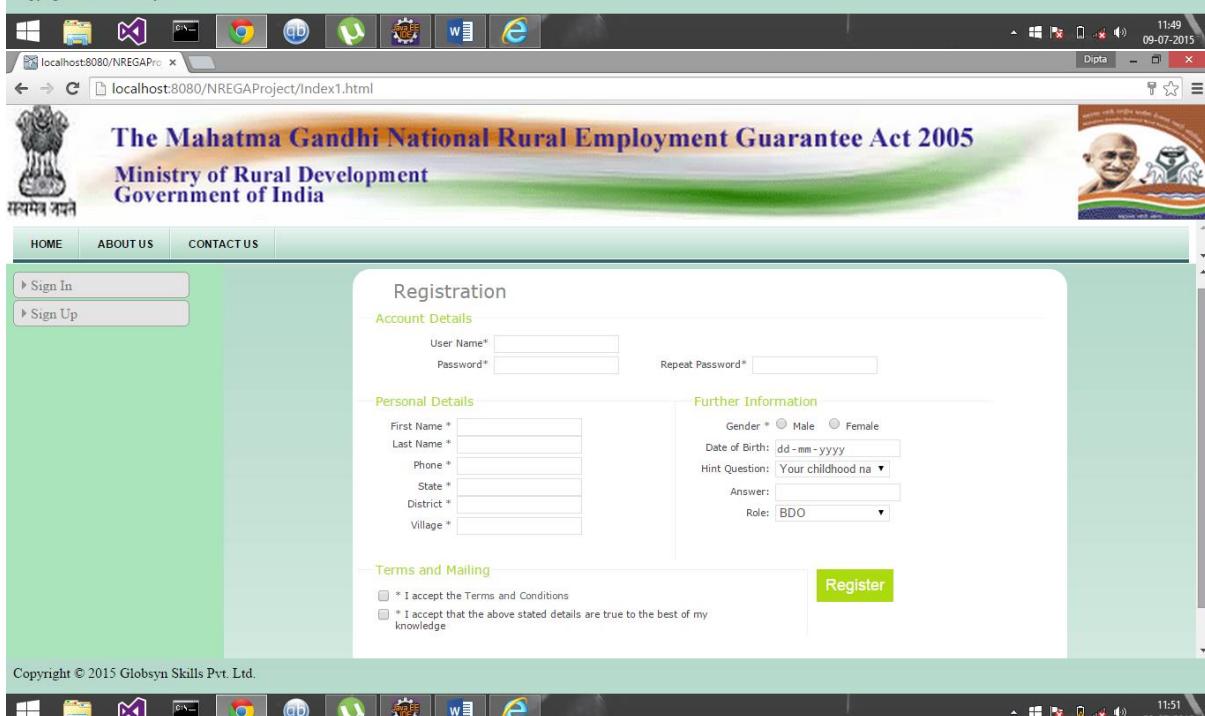
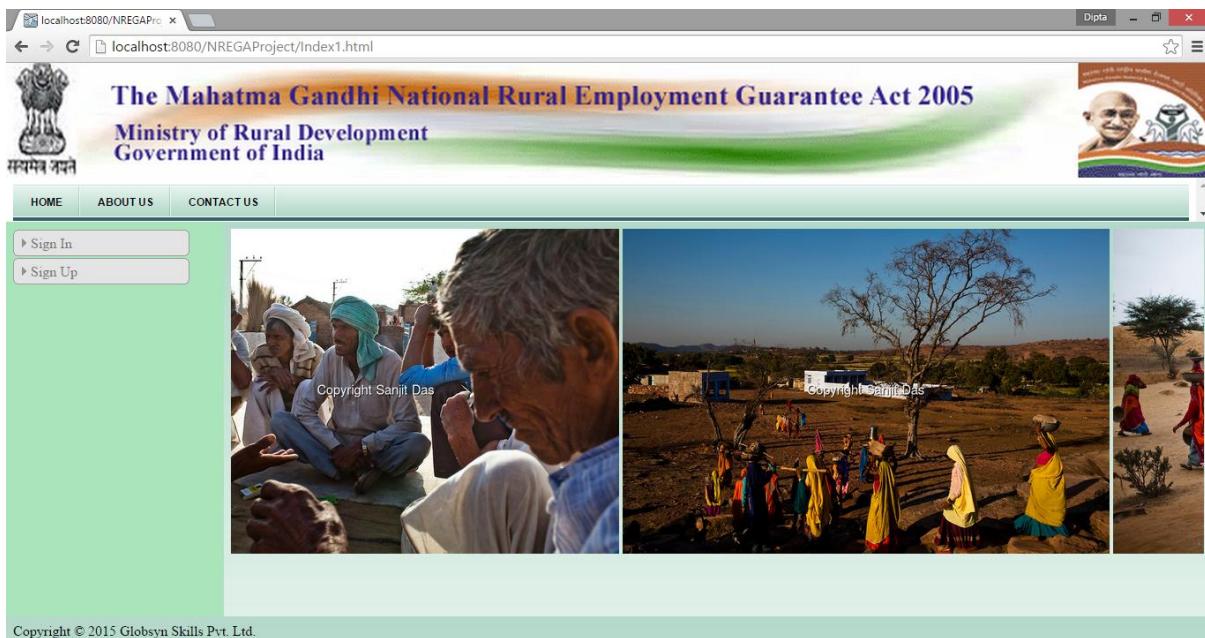
Umesh Kumar Sharma



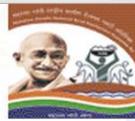




# Screenshots



The Mahatma Gandhi National Rural Employment Guarantee Act 2005  
 Ministry of Rural Development  
 Government of India



HOME ABOUT US CONTACT US

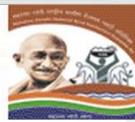
Sign In Sign Up

Contact Us

District MGNREGA Cell,  
 Room No. 5, 2nd Floor,  
 Collectorate Building ,  
 Kolkata-700101

Phone: 03561-224826,  
 Phone/FAX: 03561-222334,  
 Email: nrega.kol@gmail.com

The Mahatma Gandhi National Rural Employment Guarantee Act 2005  
 Ministry of Rural Development  
 Government of India



HOME ABOUT US CONTACT US

Sign In Sign Up

My Profile Create New Employee Assign Employees to project Mark Attendance Employee Profile

Your Profile:

First Name:	Dipta
Last Name:	Das Chowdhury
Role:	Gram Panchayat
Phone No.:	9007847306
Date of Birth:	2012-05-26
State:	West Bengal
District:	Birbhum
Village:	Santiniketan

Logout

Copyright © 2015 Globsyn Skills Pvt. Ltd.

Umesh Kumar Sharma



Document sign date : Jul 11, 2015

localhost:8080/NREGAPro x Dipta ← → C localhost:8080/NREGAProject/Index1.html

**The Mahatma Gandhi National Rural Employment Guarantee Act 2005**

Ministry of Rural Development  
Government of India

Sign In Sign Up My Profile Create New Employee Assign Employees to Project Mark Attendance Employee Profile

Search For Employee:

Employee name:  Search

Logout

Employee ID:	First Name:	Last Name:	Phone:	Age:	State:	District:	Village:	Address:	Aadhar No.:	Bank Name:	Account No.:	Attendance:	Projects:	Availability:
1	Snigdha	Jain	9851833396	20	MANIPUR	Kolkata	Kalikapur	ruby	420	State Bank Of India	9043	2	abc	null

Copyright © 2015 Globsyn Skills Pvt. Ltd.

localhost:8080/NREGAPro x Dipta ← → C localhost:8080/NREGAProject/Index1.html

**The Mahatma Gandhi National Rural Employment Guarantee Act 2005**

Ministry of Rural Development  
Government of India

Sign In Sign Up My Profile Create New Employee Assign Employees to Project Mark Attendance Employee Profile

Project Assignment:

Employee Id:   
Project name:   
Start Date:   
End Date:

Assign Project

Employee ID:	First Name:	Last Name:	Phone:	Age:	State:	District:	Village:	Address:	Aadhar No.:	Bank Name:	Account No.:	Attendance:	Projects:	Availability:
1	Snigdha	Jain	9851833396	20	MANIPUR	Kolkata	Kalikapur	ruby	420	State Bank Of India	9043	2	abc	null

Copyright © 2015 Globsyn Skills Pvt. Ltd.



localhost:8080/NREGAProject/Index1.html

## The Mahatma Gandhi National Rural Employment Guarantee Act 2005

Ministry of Rural Development  
Government of India

Sign In | Sign Up | My Profile | Create a Project | See list of Projects | See details of Projects | People working in a Projects

Project Creation

Project Name *	<input type="text"/>	Maximum Employees *	<input type="text"/>
Estimated Duration *	<input type="text"/>	Purpose *	<input type="text"/>
Assigned to *	<input type="text"/>	Allocated Money *	<input type="text"/>
Allocated Date *	<input type="text"/> dd-mm-yyyy	Village *	<input type="text"/>
District *	<input type="text"/>	<b>Register</b>	

Logout

Copyright © 2015 Globsyn Skills Dev. Ltd.  
localhost:8080/NREGAProject/createProject.jsp

localhost:8080/NREGAProject/Index1.html

## The Mahatma Gandhi National Rural Employment Guarantee Act 2005

Ministry of Rural Development  
Government of India

Sign In | Sign Up | My Profile | Create New Employee | Assign Employees | Mark Attendance | Employee Profile | Project History

1	Snigdha	Jain	9851833396	20	MANIPUR	Kolkata	Kalikapur	ruby	420	State Bank Of India	9043	2	Rasta banao	null
2	Dipto	Das	12344556	20	JHARKHAND	fghjk	sdfghj	fghj	cvbn	Union Bank	xdcfvgbhn	0	null	null

Project History

- Project1
- abc
- abc
- pqr
- abc
- abc
- Rasta banao

Copyright © 2015 Globsyn Skills Dev. Ltd.  
localhost:8080/NREGAProject/EmployeController/historyRecordNo=1




localhost:8080/NREGAProject/Index1.html

The Mahatma Gandhi National Rural Employment Guarantee Act 2005

Ministry of Rural Development  
Government of India

Sign In  
Sign Up  
My Profile  
Create a Project  
See list of Projects  
See details of Projects  
People working in a Projects

Available Projects  
Select a Project to view Employees working on it  
abc  
pqr

Logout

Copyright © 2015 Globsyn Skills Dev. Ltd.  
localhost:8080/NREGAProject/projectList.jsp

DHT: Waiting to log in D: 0.0 kB/s T: 52.2 kB U: 0.0 kB/s T: 1.6 kB 15:00 09-07-2015

Umesh Kumar Sharma



# Future Scope of Improvements

- We can have a method to stop / check multiple enrollment of same person in the system.
- WE can have a method to monitor that after enrolled person get employment within 15 days as per the Act. A method to keep check on the proxy workers.

Umesh Kumar Sharma



# Code

## Controller

### 1.Attendance controller

Java EE - NREGAProject/src/controller/AttendanceController.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Index.html leftBar.jsp projectDetails.jsp projectList.jsp AttendanceController.java

1 package controller;
2
3 import java.io.IOException;
4 import java.util.ArrayList;
5 import javax.servlet.RequestDispatcher;
6 import javax.servlet.ServletException;
7 import javax.servlet.annotation.WebServlet;
8 import javax.servlet.http.HttpServlet;
9 import javax.servlet.http.HttpServletRequest;
10 import javax.servlet.http.HttpServletResponse;
11 import javax.servlet.http.HttpSession;
12 import dao.AttendanceDAO;
13 import dao.AttendanceDAOImpl;
14
15 @WebServlet("/AttendanceController")
16 public class AttendanceController extends HttpServlet {
17     private static final long serialVersionUID = 1L;
18     String dbStatus;
19     AttendanceDAO ad = new AttendanceDAOImpl();
20
21     protected void doGet(HttpServletRequest request,
22             HttpServletResponse response) throws ServletException, IOException {
23         //***** Display Employee List *****/
24         String btnDisplay = request.getParameter("btnAttendance");
25         if (btnDisplay != null) {
26             String project = request.getParameter("project");
27             ArrayList<String> alist = ad.displayEmployee(project);
28             if (alist != null) {
29                 getServletContext().setAttribute("alist", alist);
30                 RequestDispatcher rd = request
31                     .getRequestDispatcher("attendance.jsp");
32                 rd.forward(request, response);
33             }
34         }
35         //***** Display Project List *****/
36         String btnVillageSub = request.getParameter("btnVillageSub");
37         if (btnVillageSub != null) {
38             if (btnVillageSub != null) {
39                 String village = request.getParameter("village");
40                 if (village != null) {
41                     getServletContext().setAttribute("village", village);
42                     RequestDispatcher rd = request
43                         .getRequestDispatcher("attendance.jsp");
44                     rd.forward(request, response);
45                 }
46             }
47             //***** Display Village List *****/
48             String btnDistrictSub = request.getParameter("btnDistrictSub");
49             if (btnDistrictSub != null) {
50                 String district = request.getParameter("district");
51                 System.out.println(district);
52                 if (district != null) {
53                     getServletContext().setAttribute("district", district);
54                     RequestDispatcher rd = request
55                         .getRequestDispatcher("attendance.jsp");
56                     rd.forward(request, response);
57                 }
58             }
59             //***** Attendance Updating *****/
60             String att[] = request.getParameterValues("attendance");
61             if (att != null) {
62                 for (int i = 0; i < att.length; i++) {
63                     dbStatus = ad.updateAttendance(Integer.parseInt(att[i]));
64                 }
65                 if (!dbStatus.equals(null)) {
66                     HttpSession session = request.getSession();
67                     session.setAttribute("dbStatus", dbStatus);
68                     response.sendRedirect("attendance.jsp");
69                 }
70             }
71         }
72     }
73 }
```

Writable Smart Insert 13:30 Sign in to Google...

Java EE - NREGAProject/src/controller/AttendanceController.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Index.html leftBar.jsp projectDetails.jsp projectList.jsp AttendanceController.java

37     if (btnVillageSub != null) {
38         String village = request.getParameter("village");
39         if (village != null) {
40             getServletContext().setAttribute("village", village);
41             RequestDispatcher rd = request
42                 .getRequestDispatcher("attendance.jsp");
43             rd.forward(request, response);
44         }
45     }
46
47     //***** Display Village List *****/
48     String btnDistrictSub = request.getParameter("btnDistrictSub");
49     if (btnDistrictSub != null) {
50         String district = request.getParameter("district");
51         System.out.println(district);
52         if (district != null) {
53             getServletContext().setAttribute("district", district);
54             RequestDispatcher rd = request
55                 .getRequestDispatcher("attendance.jsp");
56             rd.forward(request, response);
57         }
58     }
59     //***** Attendance Updating *****/
60     String att[] = request.getParameterValues("attendance");
61     if (att != null) {
62         for (int i = 0; i < att.length; i++) {
63             dbStatus = ad.updateAttendance(Integer.parseInt(att[i]));
64         }
65         if (!dbStatus.equals(null)) {
66             HttpSession session = request.getSession();
67             session.setAttribute("dbStatus", dbStatus);
68             response.sendRedirect("attendance.jsp");
69         }
70     }
71 }
```

Writable Smart Insert 13:30 Sign in to Google...

### 2.Employee Controller

Umesh Kumar Sharma



Java EE - NREGAProject/src/controller/EmployeeController.java - Eclipse

```

1 package controller;
2
3 import java.io.IOException;
4 import java.sql.Date;
5 import java.util.List;
6 import javax.servlet.RequestDispatcher;
7 import javax.servlet.ServletException;
8 import javax.servlet.annotation.WebServlet;
9 import javax.servlet.http.HttpServlet;
10 import javax.servlet.http.HttpServletRequest;
11 import javax.servlet.http.HttpServletResponse;
12 import javax.servlet.http.HttpSession;
13 import model.EmployeeBean;
14 import dao.EmployeeDAOImpl;
15
16 @WebServlet("/EmployeeController")
17 public class EmployeeController extends HttpServlet {
18     private static final long serialVersionUID = 1L;
19     String dbStatus;
20     EmployeeDAOImpl eud = new EmployeeDAOImpl();
21
22     protected void doGet(HttpServletRequest request,
23             HttpServletResponse response) throws ServletException, IOException {
24         EmployeeBean ebean = new EmployeeBean();
25         ebean.setFirstName(request.getParameter("firstName"));
26         ebean.setLastName(request.getParameter("lastName"));
27         ebean.setPhone(request.getParameter("phone"));
28         ebean.setAge(request.getParameter("age"));
29         ebean.setState(request.getParameter("state"));
30         ebean.setDistrict(request.getParameter("district"));
31         ebean.setVillage(request.getParameter("village"));
32         ebean.setAddress(request.getParameter("address"));
33         ebean.setAadharNo(request.getParameter("aadharNo"));
34         ebean.setBankName(request.getParameter("bankName"));
35         ebean.setAccNo(request.getParameter("accNo"));
36         ebean.setAccType(request.getParameter("accType"));
37         /****** SAVE ***** */
38         String btnSave = request.getParameter("btnSave");
  
```

Java EE - NREGAProject/src/controller/EmployeeController.java - Eclipse

```

39         if (btnSave != null) {
40             if (btnSave.equals("Save")) {
41                 dbStatus = eud.saveEmployee(ebean);
42                 if (!dbStatus.equals(null)) {
43                     HttpSession session = request.getSession();
44                     session.setAttribute("dbStatus", dbStatus);
45                     response.sendRedirect("createEmployee.jsp");
46                 }
47             }
48             /****** DISPLAY ***** */
49             String btnDisplay = request.getParameter("btnDisplay");
50             if (btnDisplay != null) {
51                 List<EmployeeBean> elist = eud.displayEmployee();
52                 if (elist != null) {
53                     getServletContext().setAttribute("eList", elist);
54                     RequestDispatcher rd = request
55                         .getRequestDispatcher("displayEmployee.jsp");
56                     rd.forward(request, response);
57                 }
58             }
59             /****** DISPLAY PROJECT HISTORY ***** */
60             String btnDisplay1 = request.getParameter("btnDisplay1");
61             if (btnDisplay1 != null) {
62                 List<EmployeeBean> elist2 = eud.displayEmployee();
63                 if (elist2 != null) {
64                     getServletContext().setAttribute("eList2", elist2);
65                     RequestDispatcher rd = request
66                         .getRequestDispatcher("projectHistory.jsp");
67                     rd.forward(request, response);
68                 }
69             }
70             /****** SEARCH DATA ***** */
71             String btnSearch = request.getParameter("btnSearch");
72             if (btnSearch != null) {
73                 /*http://localhost:8080/NREGAProject/searchEmployee.jsp?name=Umesh%20Kumar%20Sharma*/
  
```



 A handwritten signature in black ink that reads "Umesh Kumar Sharma".

Java EE - NREGAProject/src/controller/EmployeeController.java - Eclipse

```

 71 String btnSearch = request.getParameter("btnSearch");
 72 if (btnSearch != null) {
 73     List<EmployeeBean> eList = eud.searchEmployee(eBean.getFirstName());
 74     if (eList != null) {
 75         getServletContext().setAttribute("eList", eList);
 76         RequestDispatcher rd = request
 77             .getRequestDispatcher("assignEmployee.jsp");
 78         rd.forward(request, response);
 79     }
 80 }
 81 **** SEARCH DATA FOR PROJECT HISTORY*****
 82 String btnSearch1 = request.getParameter("btnSearch1");
 83 if (btnSearch1 != null) {
 84     List<EmployeeBean> eList2 = eud.searchEmployee(eBean.getFirstName());
 85     if (eList2 != null) {
 86         getServletContext().setAttribute("eList2", eList2);
 87         RequestDispatcher rd = request
 88             .getRequestDispatcher("projecthistory.jsp");
 89         rd.forward(request, response);
 90     }
 91 }
 92 **** UPDATE DATA ****
 93 String btnUpdate = request.getParameter("btnUpdate");
 94 if (btnUpdate != null) {
 95     EmployeeBean eBean = new EmployeeBean();
 96     eBean.setRecordNo(Integer.parseInt(request.getParameter("recordNo")));
 97     eBean.setProject(request.getParameter(" projectName"));
 98     eud.saveProjectHistory(eBean);
 99     eBean.setRecordNo(Integer.parseInt(request.getParameter("recordNo")));
100    eBean.setProject(request.getParameter(" projectName"));
101    eBean.setStartDate(Date.valueOf(request.getParameter("startDate")));
102    eBean.setEndDate(Date.valueOf(request.getParameter("endDate")));
103    eBean.setAvailability("On");
104    dbStatus = eud.updateEmployee(eBean);
105    if (!dbStatus.equals(null)) {
106        HttpSession session = request.getSession();
107        session.setAttribute("msg", "Record Updated");
108        response.sendRedirect("assignEmployee.jsp");
109    }
110 }
111 **** Employee Profile *****
112 String recordNo = request.getParameter("displayRecordNo");
113 if (recordNo != null) {
114     List<EmployeeBean> eList = eud.displayEmployeeProfile(Integer
115         .parseInt(recordNo));
116     if (eList != null) {
117         getServletContext().setAttribute("eList", eList);
118         RequestDispatcher rd = request
119             .getRequestDispatcher("employeeProfile.jsp");
120         rd.forward(request, response);
121     }
122 }
123
124 **** Project History *****
125 String recordNo1 = request.getParameter("historyRecordNo");
126 if (recordNo1 != null) {
127     List<EmployeeBean> eList1 = eud.displayProjectHistory(Integer.parseInt(recordNo1));
128     if (eList1 != null) {
129         getServletContext().setAttribute("eList1", eList1);
130         RequestDispatcher rd = request
131             .getRequestDispatcher("projectHistory.jsp");
132         rd.forward(request, response);
133     }
134 }
135
136 }
137
138 }
139
140 }
141
142 }
143
144 }
145
  
```

Java EE - NREGAProject/src/controller/EmployeeController.java - Eclipse

```

 109     response.sendRedirect("assignEmployee.jsp");
110 }
111 **** Employee Profile *****
112 String recordNo = request.getParameter("displayRecordNo");
113 if (recordNo != null) {
114     List<EmployeeBean> eList = eud.displayEmployeeProfile(Integer
115         .parseInt(recordNo));
116     if (eList != null) {
117         getServletContext().setAttribute("eList", eList);
118         RequestDispatcher rd = request
119             .getRequestDispatcher("employeeProfile.jsp");
120         rd.forward(request, response);
121     }
122 }
123
124 **** Project History *****
125 String recordNo1 = request.getParameter("historyRecordNo");
126 if (recordNo1 != null) {
127     List<EmployeeBean> eList1 = eud.displayProjectHistory(Integer.parseInt(recordNo1));
128     if (eList1 != null) {
129         getServletContext().setAttribute("eList1", eList1);
130         RequestDispatcher rd = request
131             .getRequestDispatcher("projectHistory.jsp");
132         rd.forward(request, response);
133     }
134 }
135
136 }
137
138 }
139
140 }
141
142 }
143
144 }
145
  
```

### 3. Project Controller



Java EE - NREGAProject/src/controller/ProjectController.java - Eclipse

```

1 package controller;
2
3 @WebServlet("/ProjectController")
4 public class ProjectController extends HttpServlet {
5     private static final long serialVersionUID = 1L;
6
7     protected void doGet(HttpServletRequest request,
8             HttpServletResponse response) throws ServletException, IOException {
9
10        String btnSave = request.getParameter("btnSave");
11        if (btnSave != null) {
12            System.out.println(pBean.getVillage());
13            String dbStatus;
14            pBean.setProjectName(request.getParameter("projectName"));
15            pBean.setMaxEmployee(Integer.parseInt(request.getParameter("max")));
16            pBean.setPurpose(request.getParameter("purpose"));
17            pBean.setDuration(Integer.parseInt(request.getParameter("duration")));
18            pBean.setAssignedTo(request.getParameter("assign"));
19            pBean.setHoneyAllocated(Integer.parseInt(request
20                .getParameter("money")));
21            pBean.setDate_of_allocation(Date.valueOf(request
22                .getParameter("date")));
23            pBean.setVillage(request.getParameter("village"));
24            pBean.setDistrict(request.getParameter("district"));
25            pBean.setStatus("active");
26            dbStatus = prd.save(pBean);
27            if (dbStatus != null) {
28
29                HttpSession session = request.getSession();
30                session.setAttribute("dbStatus", dbStatus);
31                response.sendRedirect("createProject.jsp");
32            }
33        }
34    }
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
  
```

Java EE - NREGAProject/src/controller/ProjectController.java - Eclipse

```

1 package controller;
2
3 @WebServlet("/ProjectController")
4 public class ProjectController extends HttpServlet {
5     private static final long serialVersionUID = 1L;
6
7     protected void doGet(HttpServletRequest request,
8             HttpServletResponse response) throws ServletException, IOException {
9
10        pBean.setProjectName(request.getParameter("projectName"));
11        pBean.setMaxEmployee(Integer.parseInt(request.getParameter("max")));
12        pBean.setPurpose(request.getParameter("purpose"));
13        pBean.setDuration(Integer.parseInt(request.getParameter("duration")));
14        pBean.setAssignedTo(request.getParameter("assign"));
15        pBean.setHoneyAllocated(Integer.parseInt(request
16            .getParameter("money")));
17        pBean.setDate_of_allocation(Date.valueOf(request
18            .getParameter("date")));
19        pBean.setVillage(request.getParameter("village"));
20        pBean.setDistrict(request.getParameter("district"));
21        pBean.setStatus("active");
22        dbStatus = prd.save(pBean);
23        if (dbStatus != null) {
24
25            HttpSession session = request.getSession();
26            session.setAttribute("dbStatus", dbStatus);
27            response.sendRedirect("createProject.jsp");
28        }
29    }
30
31
32
33
34
35
36
37
38
39
39
40
41
42
43
44
45
46
47
48
49
49
50
51
52
53
  
```

#### 4.Sign Up Controller




Java EE - NREGAProject/src/controller/SignUpController.java - Eclipse

```

 1 package controller;
 2
 3 import java.io.IOException;
 4 import javax.servlet.RequestDispatcher;
 5 import javax.servlet.ServletException;
 6 import javax.servlet.annotation.WebServlet;
 7 import javax.servlet.http.HttpServlet;
 8 import javax.servlet.http.HttpServletRequest;
 9 import javax.servlet.http.HttpServletResponse;
10 import javax.servlet.http.HttpSession;
11 import model.SignUpBean;
12 import dao.SignUpDao;
13 import dao.SignUpDaoImpl;
14 import webServlet.HttpServlet;
15 @WebServlet("/SignUpController")
16 public class SignUpController extends HttpServlet {
17     private static final long serialVersionUID = 1L;
18     SignUpBean subean = new SignUpBean();
19     SignUpDao signupDao = new SignUpDaoImpl();
20     String dbStatus;
21     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
22
23         String btnSignUp = request.getParameter("btnSignUp");
24         String btnSignIn = request.getParameter("btnSignIn");
25         if (btnSignUp != null) {
26             subean.setUserName(request.getParameter("username"));
27             subean.setPassword(request.getParameter("password"));
28             subean.setFirstName(request.getParameter("firstname"));
29             subean.setLastName(request.getParameter("lastname"));
30             subean.setPhone(request.getParameter("phone"));
31             subean.setState(request.getParameter("state"));
32             subean.setDistrict(request.getParameter("district"));
33             subean.setVillage(request.getParameter("village"));
34             subean.setGender(request.getParameter("gender"));
35             subean.setDobDate.valueOf(request.getParameter("dob"));
36             subean.setHintQuestion(request.getParameter("hintQuestion"));
37             subean.setAnswer(request.getParameter("answer"));
38
39         subBean.setRole(request.getParameter("role"));
40         if (subBean.getPassword().equals(request.getParameter("password1"))) {
41             dbStatus = signUpDao.saveSignUp(subBean);
42             request.setAttribute("dbStatus", dbStatus);
43             RequestDispatcher rd = request
44                     .getRequestDispatcher("signUp.jsp");
45             rd.forward(request, response);
46         } else {
47             request.setAttribute("dbStatus", "Password not matching....");
48             RequestDispatcher rd = request.getRequestDispatcher("signUp.jsp");
49             rd.forward(request, response);
50         }
51         } else if (btnSignIn != null) {
52             try {
53                 subBean.setUserName(request.getParameter("username"));
54                 subBean.setPassword(request.getParameter("password"));
55                 subBean = signUpDao.login(subBean);
56                 if (subBean.isValid() && subBean.getRole().equals("Gram Panchayat")) {
57                     HttpSession session = request.getSession(true);
58                     session.setAttribute("loggedUser", subBean);
59                     request.getRequestDispatcher("gramPanchayat.jsp")
60                         .forward(request, response);
61                 } else if (subBean.isValid() && subBean.getRole().equals("BDO")) {
62                     HttpSession session = request.getSession(true);
63                     session.setAttribute("loggedUser", subBean);
64                     request.getRequestDispatcher("bdo.jsp")
65                         .forward(request, response);
66                 } else {
67                     response.sendRedirect("error.jsp");
68                 } } catch (Exception e) {
69                 e.printStackTrace();
70             }
71         }
72     }
73 }
74 
```

Java EE - NREGAProject/src/controller/SignUpController.java - Eclipse

```

 1 package controller;
 2
 3 import java.io.IOException;
 4 import javax.servlet.RequestDispatcher;
 5 import javax.servlet.ServletException;
 6 import javax.servlet.annotation.WebServlet;
 7 import javax.servlet.http.HttpServlet;
 8 import javax.servlet.http.HttpServletRequest;
 9 import javax.servlet.http.HttpServletResponse;
10 import javax.servlet.http.HttpSession;
11 import model.SignUpBean;
12 import dao.SignUpDao;
13 import dao.SignUpDaoImpl;
14 import webServlet.HttpServlet;
15 @WebServlet("/SignUpController")
16 public class SignUpController extends HttpServlet {
17     private static final long serialVersionUID = 1L;
18     SignUpBean subean = new SignUpBean();
19     SignUpDao signupDao = new SignUpDaoImpl();
20     String dbStatus;
21     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
22
23         String btnSignUp = request.getParameter("btnSignUp");
24         String btnSignIn = request.getParameter("btnSignIn");
25         if (btnSignUp != null) {
26             subbean.setUserName(request.getParameter("username"));
27             subbean.setPassword(request.getParameter("password"));
28             subbean.setFirstName(request.getParameter("firstname"));
29             subbean.setLastName(request.getParameter("lastname"));
30             subbean.setPhone(request.getParameter("phone"));
31             subbean.setState(request.getParameter("state"));
32             subbean.setDistrict(request.getParameter("district"));
33             subbean.setVillage(request.getParameter("village"));
34             subbean.setGender(request.getParameter("gender"));
35             subbean.setDobDate.valueOf(request.getParameter("dob"));
36             subbean.setHintQuestion(request.getParameter("hintQuestion"));
37             subbean.setAnswer(request.getParameter("answer"));
38
39         subBean.setRole(request.getParameter("role"));
40         if (subBean.getPassword().equals(request.getParameter("password1"))) {
41             dbStatus = signUpDao.saveSignUp(subBean);
42             request.setAttribute("dbStatus", dbStatus);
43             RequestDispatcher rd = request
44                     .getRequestDispatcher("signUp.jsp");
45             rd.forward(request, response);
46         } else {
47             request.setAttribute("dbStatus", "Password not matching....");
48             RequestDispatcher rd = request.getRequestDispatcher("signUp.jsp");
49             rd.forward(request, response);
50         }
51         } else if (btnSignIn != null) {
52             try {
53                 subBean.setUserName(request.getParameter("username"));
54                 subBean.setPassword(request.getParameter("password"));
55                 subBean = signUpDao.login(subBean);
56                 if (subBean.isValid() && subBean.getRole().equals("Gram Panchayat")) {
57                     HttpSession session = request.getSession(true);
58                     session.setAttribute("loggedUser", subBean);
59                     request.getRequestDispatcher("gramPanchayat.jsp")
60                         .forward(request, response);
61                 } else if (subBean.isValid() && subBean.getRole().equals("BDO")) {
62                     HttpSession session = request.getSession(true);
63                     session.setAttribute("loggedUser", subBean);
64                     request.getRequestDispatcher("bdo.jsp")
65                         .forward(request, response);
66                 } else {
67                     response.sendRedirect("error.jsp");
68                 } } catch (Exception e) {
69                 e.printStackTrace();
70             }
71         }
72     }
73 }
74 
```

## DAO IMPLEMENTATION

### 1.AttendanceDAOImpl



Umesh Kumar Sharma

## Java EE - NREGAProject/src/dao/AttendanceDAOImpl.java - Eclipse

```

1 package dao;
2
3 import java.sql.Connection;
4
5 public class AttendanceDAOImpl implements AttendanceDAO{
6
7     private Connection con;
8     private PreparedStatement pst;
9     private ResultSet rs1;
10    private String dbStatus;
11
12    ArrayList alist = new ArrayList();
13
14    @Override
15    public ArrayList displayEmployee(String project) {
16        con = (Connection) ConnectionFactory.getInstance().getConnection();
17        ArrayList alist = new ArrayList();
18        try {
19            pst = con.prepareStatement("select * from employeeDetail where project=?");
20            pst.setString(1,project);
21            rs1 = pst.executeQuery();
22            while(rs1.next()) {
23                AttendanceBean aBean = new AttendanceBean();
24                aBean.setRecordNo(rs1.getInt(1));
25                aBean.setFirstName(rs1.getString(2));
26                aBean.setLastName(rs1.getString(3));
27                alist.add(aBean);
28            }
29        } catch (Exception e) {
30            e.printStackTrace();
31        }
32        return alist;
33    }
34
35    @Override
36    public String updateAttendance(int recordNo) {
37        con = (Connection) ConnectionFactory.getInstance().getConnection();
38
39    }
40
41    @Override
42    public String updateAttendance(int recordNo) {
43        con = (Connection) ConnectionFactory.getInstance().getConnection();
44        try {
45            pst = con.prepareStatement("select * from employeeDetail where recordNo=?");
46            pst.setInt(1,recordNo);
47            rs1 = pst.executeQuery();
48            if(rs1.next()) {
49                int at = Integer.parseInt(rs1.getString(13));
50                at++;
51                pst = con.prepareStatement("update employeeDetail set attendance= ? where recordNo=?");
52                pst.setString(1, Integer.toString(at));
53                pst.setInt(2,recordNo);
54                int i = pst.executeUpdate();
55                if (i > 0)
56                    dbStatus = "Update Successful";
57                else {
58                    dbStatus = "Update Failed";
59                }
60            }
61        } catch (Exception e) {
62            e.printStackTrace();
63        }
64        System.out.println(dbStatus);
65        return dbStatus;
66    }
67
68 }
69
70

```

Writable Smart Insert 1:1 Sign in to Google...

11:41  
09-07-2015

## Java EE - NREGAProject/src/dao/AttendanceDAOImpl.java - Eclipse

```

34
35    } catch (Exception e) {
36        e.printStackTrace();
37    }
38    return alist;
39}
40
41@Override
42public String updateAttendance(int recordNo) {
43    con = (Connection) ConnectionFactory.getInstance().getConnection();
44    try {
45        pst = con.prepareStatement("select * from employeeDetail where recordNo=?");
46        pst.setInt(1,recordNo);
47        rs1 = pst.executeQuery();
48        if(rs1.next()) {
49            int at = Integer.parseInt(rs1.getString(13));
50            at++;
51            pst = con.prepareStatement("update employeeDetail set attendance= ? where recordNo=?");
52            pst.setString(1, Integer.toString(at));
53            pst.setInt(2,recordNo);
54            int i = pst.executeUpdate();
55            if (i > 0)
56                dbStatus = "Update Successful";
57            else {
58                dbStatus = "Update Failed";
59            }
60        }
61    } catch (Exception e) {
62        e.printStackTrace();
63    }
64    System.out.println(dbStatus);
65    return dbStatus;
66}
67
68 }
69
70

```

Writable Smart Insert 1:1 Sign in to Google...

11:41  
09-07-2015

## 2.EmployeeDAOImpl



The screenshot shows the Eclipse IDE interface with the title bar "Java EE - NREGAProject/src/dao/EmployeeDAOImpl.java - Eclipse". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help. The toolbar has various icons for file operations like Open, Save, Cut, Copy, Paste, Find, etc. The quick access bar has a "Quick Access" button and a "Java EE" icon. The code editor displays the following Java code:

```
1 package dao;
2
3 import java.sql.Connection;
4
5 public class EmployeeDAOImpl implements EmployeeDAO {
6     private Connection con;
7     private PreparedStatement ps;
8     private ResultSet rs;
9     private String dbStatus;
10
11     @Override
12     public String saveEmployee(EmployeeEbean eBean) {
13         try {
14             con = (Connection) ConnectionFactory.getInstance().getConnection();
15             String sql = "INSERT INTO employeeDetail (firstName, lastName, phone, age, state, district, village, address, adharNo, bankName, accNo, availability, attendance) values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
16             ps = con.prepareStatement(sql);
17             ps.setString(1, eBean.getFirstName());
18             ps.setString(2, eBean.getLastName());
19             ps.setString(3, eBean.getPhone());
20             ps.setString(4, eBean.getAge());
21             ps.setString(5, eBean.getState());
22             ps.setString(6, eBean.getDistrict());
23             ps.setString(7, eBean.getVillage());
24             ps.setString(8, eBean.getAddress());
25             ps.setString(9, eBean.getAdharNo());
26             ps.setString(10, eBean.getBankName());
27             ps.setString(11, eBean.getAccNo());
28             ps.setString(12, "off");
29             ps.setString(13, "0");
30             int row = ps.executeUpdate();
31             if (row > 0) {
32                 dbStatus = "Employee Created";
33             } else {
34                 dbStatus = "Creation of Employee Failed";
35             }
36         } catch (SQLException ex) {
37             dbStatus = "ERROR: " + ex.getMessage();
38         }
39     }
40 }
```

The screenshot shows the Eclipse IDE interface with the following details:

- Top Bar:** Shows icons for Windows, File Explorer, Task View, Google Drive, GitHub, and Project Explorer.
- Title Bar:** "Java EE - NREGAProject/src/dao/EmployeeDAOImpl.java - Eclipse".
- Toolbar:** Standard Eclipse toolbar with icons for file operations, search, and project management.
- Quick Access:** A button labeled "Quick Access" with a dropdown menu.
- Java EE:** A button labeled "Java EE" with a dropdown menu.
- Code Editor:** Displays the Java code for `EmployeeDAOImpl.java`. The code handles database errors and retrieves employee details from a database using JDBC.

```
45     dbStatus = "ERROR: " + ex.getMessage();
46     ex.printStackTrace();
47 } catch (Exception ex) {
48     dbStatus = "ERROR: " + ex.getMessage();
49     ex.printStackTrace();
50 }
51 return dbStatus;
52 }
53 }
54
55 @Override
56 public List<EmployeeBean> displayEmployee() {
57     con = ConnectionFactory.getInstance().getConnection();
58     List<EmployeeBean> eList = new ArrayList<EmployeeBean>();
59     try {
60         ps = con.prepareStatement("select * from employeedetail");
61         rs = ps.executeQuery();
62         while (rs.next()) {
63             EmployeeBean euBean = new EmployeeBean();
64             euBean.setRecordNo(rs.getInt(1));
65             euBean.setFirstName(rs.getString(2));
66             euBean.setLastName(rs.getString(3));
67             euBean.setPhone(rs.getString(4));
68             euBean.setAge(rs.getString(5));
69             euBean.setState(rs.getString(6));
70             euBean.setDistrict(rs.getString(7));
71             euBean.setVillage(rs.getString(8));
72             euBean.setAddress(rs.getString(9));
73             euBean.setAdharNo(rs.getString(10));
74             euBean.setBankName(rs.getString(11));
75             euBean.setAccNo(rs.getString(12));
76             euBean.setAttendance(rs.getInt(13));
77             euBean.setProject(rs.getString(14));
78
79             eList.add(euBean);
80         }
81     } catch (Exception e) {
```

Umesh Kumar Sharma



The screenshot shows the Eclipse IDE interface with the title bar "Java EE - NREGAProject/src/dao/EmployeeDAOImpl.java - Eclipse". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar contains various icons for file operations like Open, Save, Cut, Copy, Paste, Find, and Run. The left sidebar shows the package structure under "Java EE". The main editor area displays the Java code for EmployeeDAOImpl.java, which includes methods for updating employees and searching by first name. The code uses JDBC to interact with a database.

```
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java EE
 SignUpController.java AttendanceDAOImpl.java EmployeeDAOImpl.java
 83     }
 84     return eList;
 85   }
 86
 87   @Override
 88   public String updateEmployee(EmployeeBean ebean) {
 89     con = ConnectionFactory.getInstance().getConnection();
 90     try {
 91       ps = con.prepareStatement("update employeeDetail set project=? ,startDate=? ,endDate=? ,availability=? where recordNo=?");
 92       ps.setString(1, ebean.getProject());
 93       ps.setDate(2, ebean.getStartDate());
 94       ps.setDate(3, ebean.getEndDate());
 95       ps.setString(4, ebean.getAvailability());
 96       ps.setInt(5, ebean.getRecordNo());
 97       int temp = ps.executeUpdate();
 98       if (temp > 0) {
 99         dbStatus = "Update Successful";
100     } else {
101       dbStatus = "Not successful";
102     }
103   } catch (Exception e) {
104     dbStatus = "error" + e.getMessage();
105     e.printStackTrace();
106   }
107   return dbStatus;
108 }
109
110   @Override
111   public List<EmployeeBean> searchEmployee(String firstName) {
112     con = ConnectionFactory.getInstance().getConnection();
113     List<EmployeeBean> eList = new ArrayList<EmployeeBean>();
114     try {
115       ps = con.prepareStatement("select * from employeeDetail where firstName=?");
116       ps.setString(1, firstName);
117       rs = ps.executeQuery();
118       while (rs.next()) {
119         EmployeeBean euBean = new EmployeeBean();
120         euBean.setRecordNo(rs.getInt("recordNo"));
121         euBean.setProject(rs.getString("project"));
122         euBean.setFirstName(rs.getString("firstName"));
123         euBean.setLastName(rs.getString("lastName"));
124         euBean.setAddress(rs.getString("address"));
125         euBean.setCity(rs.getString("city"));
126         euBean.setState(rs.getString("state"));
127         euBean.setPincode(rs.getString("pincode"));
128         euBean.setMobile(rs.getString("mobile"));
129         euBean.setEmail(rs.getString("email"));
130         euBean.setAvailability(rs.getString("availability"));
131         euBean.setStartDate(rs.getDate("startDate"));
132         euBean.setEndDate(rs.getDate("endDate"));
133         eList.add(euBean);
134       }
135     } catch (Exception e) {
136     }
137     return eList;
138   }
139 }
```

The screenshot shows the Eclipse IDE interface with the following details:

- Top Bar:** Windows taskbar, Eclipse logo, File, Edit, Source, Refactor, Navigate, Project, Run, Window, Help, Writable, Smart Insert, 14:5, Sign in to Google.
- Toolbar:** Standard Eclipse toolbar with icons for file operations, search, project, run, etc.
- Quick Access:** A button labeled "Quick Access" with a small icon.
- Java EE:** A button labeled "Java EE" with a small icon.
- Left Sidebar:** Shows the current file is "SignUpController.java". Other tabs visible include "AttendanceDAOImpl.java" and "\*EmployeeDAOImpl.java".
- Code Editor:** Displays Java code for the `EmployeeDAOImpl.java` class, specifically the `displayWorkers` method. The code retrieves employee details from a database using JDBC statements.
- Bottom Bar:** Windows taskbar, Writable, Smart Insert, 14:5, Sign in to Google.

```
127     euBean.setVillage(rs.getString(8));
128     euBean.setAddress(rs.getString(9));
129     euBean.setAdharNo(rs.getString(10));
130     euBean.setBankName(rs.getString(11));
131     euBean.setAcNo(rs.getString(12));
132     euBean.setAttendance(rs.getInt(13));
133     euBean.setProject(rs.getString(14));
134
135     eList.add(euBean);
136 }
137 } catch (Exception e) {
138     e.printStackTrace();
139 }
140 return eList;
141 }
142
143 @Override
144 public ArrayList<EmployeeBean> displayWorkers(String projectName) {
145
146     ArrayList<EmployeeBean> eList = new ArrayList<EmployeeBean>();
147     con = ConnectionFactory.getInstance().getConnection();
148     try {
149         String sql = "select * from employeeDetail where project=?";
150         ps = con.prepareStatement(sql);
151         ps.setString(1, projectName);
152         rs = ps.executeQuery();
153         while (rs.next()) {
154             EmployeeBean euBean = new EmployeeBean();
155             euBean.setRecordNo(rs.getInt(1));
156             euBean.setFirstName(rs.getString(2));
157             euBean.setLastName(rs.getString(3));
158             euBean.setPhone(rs.getString(4));
159             euBean.setAge(rs.getString(5));
160             euBean.setState(rs.getString(6));
161             euBean.setDistrict(rs.getString(7));
162             euBean.setVillage(rs.getString(8));
163             euBean.setAddress(rs.getString(9));
164             eList.add(euBean);
165         }
166     } catch (Exception e) {
167         e.printStackTrace();
168     }
169 }
```

- Umesh Kumar Sharma



Java EE ~ NREGAProject/src/dao/EmployeeDAOImpl.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java EE

SignUpController.java AttendanceDAOImpl.java *EmployeeDAOImpl.java

167     euBean.setAttendance(rs.getInt(13));
168     euBean.setProject(rs.getString(14));
169
170     eList.add(euBean);
171 }
172
173 } catch (Exception e) {
174     e.printStackTrace();
175 }
176
177 return eList;
178 }

@Override
public ArrayList<EmployeeBean> displayEmployeeProfile(int recordno) {
    ArrayList<EmployeeBean> eList = new ArrayList<EmployeeBean>();
    con = ConnectionFactory.getInstance().getConnection();
    try {
        String sql = "select * from employeeDetail where recordno= ? ";
        ps = con.prepareStatement(sql);
        ps.setInt(1, recordno);
        ResultSet rs1 = ps.executeQuery();
        while (rs1.next()) {
            EmployeeBean euBean = new EmployeeBean();
            euBean.setRecordNo(rs1.getInt(1));
            euBean.setFirstName(rs1.getString(2));
            euBean.setLastName(rs1.getString(3));
            euBean.setPhone(rs1.getString(4));
            euBean.setAge(rs1.getString(5));
            euBean.setState(rs1.getString(6));
            euBean.setDistrict(rs1.getString(7));
            euBean.setVillage(rs1.getString(8));
            euBean.setAddress(rs1.getString(9));
            euBean.setAadharNo(rs1.getString(10));
            euBean.setBankName(rs1.getString(11));
            euBean.setAccNo(rs1.getString(12));
            euBean.setAttendance(rs1.getInt(13));
            euBean.setProject(rs1.getString(14));
        }
    } catch (SQLException ex) {
        dbStatus = "ERROR: " + ex.getMessage();
        ex.printStackTrace();
    } catch (Exception ex) {
        dbStatus = "ERROR: " + ex.getMessage();
        ex.printStackTrace();
    }
}

```

Java EE - NREGAProject/src/dao/EmployeeDAOImpl.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java EE

SignUpController.java AttendanceDAOImpl.java *EmployeeDAOImpl.java

221     ps.setInt(1, eBean1.getRecordNo());
222     ps.setString(2, eBean1.getProject());
223     ps.executeUpdate();
224
225 } catch (SQLException ex) {
226     dbStatus = "ERROR: " + ex.getMessage();
227     ex.printStackTrace();
228 } catch (Exception ex) {
229     dbStatus = "ERROR: " + ex.getMessage();
230     ex.printStackTrace();
231 }
232
233 }
234
235 @Override
public ArrayList<EmployeeBean> displayProjectHistory(int recordNo1) {
    ArrayList<EmployeeBean> eList1 = new ArrayList<EmployeeBean>();
    con = ConnectionFactory.getInstance().getConnection();
    try {
        String sql = "select * from projectHistory where recordNo= ? ";
        ps = con.prepareStatement(sql);
        ps.setInt(1, recordNo1);
        rs = ps.executeQuery();
        while (rs.next()) {
            EmployeeBean euBean = new EmployeeBean();
            euBean.setProject(rs.getString(2));
            eList1.add(euBean);
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}
255 }
256 }
257

```

### 3. ProjectDAOImpl



Umesh Kumar Sharma

The screenshot shows the Eclipse IDE interface with the title bar "Java EE - NREGAProject/src/dao/ProjectDAOImpl.java - Eclipse". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help. The toolbar has various icons for file operations like Open, Save, Cut, Copy, Paste, Find, etc. The quick access bar on the right has "Quick Access" and "Java EE". The code editor displays the following Java code:

```
1 package dao;
2
3 import java.sql.Connection;
4
5 public class ProjectDAOImpl implements ProjectDAO {
6     Connection con;
7     PreparedStatement pst;
8     String dbStatus;
9     ResultSet rs;
10
11     @Override
12     public String save(ProjectBean pBean) {
13
14         try {
15             con = ConnectionFactory.getInstance().getConnection();
16             String sql;
17             sql = "insert into project values(?,?,?,?,?,?,?,?,?,?)";
18             pst = con.prepareStatement(sql);
19             pst.setString(1, pBean.getProjectName());
20             pst.setString(2, pBean.getManager());
21             pst.setString(3, pBean.getDuration());
22             pst.setInt(4, pBean.getDuration());
23             pst.setString(5, pBean.getAssigned_to());
24             pst.setInt(6, pBean.getMoneyAllocated());
25             pst.setDate(7, pBean.getDate_of_allocation());
26             pst.setString(8, pBean.getVillage());
27             pst.setString(9, pBean.getDistrict());
28             pst.setString(10, "active");
29
30             // pst.setDate(7, pBean.getDateAllocated());
31
32             int row = pst.executeUpdate();
33             if (row > 0) {
34                 dbStatus = "project created successfully";
35             } else {
36                 dbStatus = "project not created";
37             }
38         } catch (Exception e) {
39             e.printStackTrace();
40         }
41     }
42 }
43
```

- Umesh Kumar Sharma



Java EE - NREGAProject/src/dao/ProjectDAOImpl.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java EE

1 SignUpController.java 2 AttendanceDAOImpl.java 3 *EmployeeDAOImpl.java 4 ProjectDAOImpl.java

74     return plist;
75 }
76
77 @Override
78 public ArrayList<ProjectBean> searchProject(String projectName) {
79
80     ArrayList<ProjectBean> pList = new ArrayList<ProjectBean>();
81     try {
82
83         con = ConnectionFactory.getInstance().getConnection();
84         String sql = "select * from project where project_name=?";
85         pst = con.prepareStatement(sql);
86         pst.setString(1, projectName);
87
88         rs = pst.executeQuery();
89         while (rs.next()) {
90             ProjectBean puBean = new ProjectBean();
91             puBean.setProjectName(rs.getString(1));
92             puBean.setMaxEmployee(rs.getInt(2));
93             puBean.setPurpose(rs.getString(3));
94             puBean.setDuration(rs.getInt(4));
95             puBean.setAssignedTo(rs.getString(5));
96             puBean.setMoneyAllocated(rs.getInt(6));
97
98             pList.add(puBean);
99         }
100    }
101 }
102
103 catch (Exception e) {
104     e.printStackTrace();
105 }
106
107 return pList;
108 }
109
110 }
  
```

Writable Smart Insert 1:1 Sign in to Google...

#### 4.SignUpDAOImpl

Java EE - NREGAProject/src/dao/SignUpDAOImpl.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java EE

1 SignUpController.java 2 AttendanceDAOImpl.java 3 *EmployeeDAOImpl.java 4 ProjectDAOImpl.java 5 SignUpDAOImpl.java

1 package dao;
2
3 import java.sql.Connection;
4
5 public class SignUpDAOImpl implements SignUpDAO {
6
7     private Connection con;
8     private PreparedStatement ps;
9     private Statement stat;
10    private ResultSet rs;
11    private String dbstatus;
12
13    public SignUpDAOImpl() {
14        super();
15    }
16
17    @Override
18    public String saveSignUp(SignUpBean subean) {
19        try {
20            con = (Connection) ConnectionFactory.getInstance().getConnec
21            String sql = "INSERT INTO signu
22            ps = con.prepareStatement(sql);
23            ps.setString(1, subean.getUserName());
24            ps.setString(2, subean.getPassword());
25            ps.setString(3, subean.getFirstName());
26            ps.setString(4, subean.getLastName());
27            ps.setString(5, subean.getGender());
28            ps.setString(6, subean.getCity());
29            ps.setString(7, subean.getDistrict());
30            ps.setString(8, subean.getVillage());
31            ps.setString(9, subean.getGender());
32            ps.setDate(10, subean.getDob());
33            ps.setString(11, subean.getHintQuestion());
34            ps.setString(12, subean.getAnswer());
35            ps.setString(13, subean.getRole());
36
37            int row = ps.executeUpdate();
38
39        } catch (Exception e) {
40            e.printStackTrace();
41        }
42    }
43
44
45
  
```

Writable Smart Insert 67:1 Sign in to Google...




Java EE - NREGAProject/src/dao/SignUpDaoImpl.java - Eclipse

```

 41     ps.setString(11, subBean.getInQuestion());
 42     ps.setString(12, subBean.getAnswer());
 43     ps.setString(13, subBean.getRole());
 44
 45     int row = ps.executeUpdate();
 46     if (row > 0) {
 47       dbStatus = "Save Successful!";
 48     } else {
 49       dbStatus = "Save Failed!";
 50     }
 51   } catch (SQLException ex) {
 52     dbStatus = "ERROR: " + ex.getMessage();
 53     ex.printStackTrace();
 54   } catch (Exception ex) {
 55     dbStatus = "ERROR: " + ex.getMessage();
 56     ex.printStackTrace();
 57   }
 58   return dbStatus;
 59 }
 60
 61 @Override
 62 public SignUpBean login(SignUpBean signUpBean) {
 63
 64   try {
 65     con = (Connection) ConnectionFactory.getInstance().getConnection();
 66     stmt = con.createStatement();
 67     String searchQuery = "select * from signup where username= '" +
 68       + signUpBean.getUserName() +
 69       + "' AND password = '" +
 70       + signUpBean.getPassword() + "' ";
 71     rs = stmt.executeQuery(searchQuery);
 72     boolean more = rs.next();
 73
 74     if (!more) {
 75       signUpBean.setValid(false);
 76     } else if (more) {
 77       signUpBean.setFirstName(rs.getString("FirstName"));
 78     }
 79   }
 80 }
```

Java EE - NREGAProject/src/dao/SignUpDaoImpl.java - Eclipse

```

 51 @Override
 52 public SignUpBean login(SignUpBean signUpBean) {
 53
 54   try {
 55     con = (Connection) ConnectionFactory.getInstance().getConnection();
 56     stmt = con.createStatement();
 57     String searchQuery = "select * from signup where username= '" +
 58       + signUpBean.getUserName() +
 59       + "' AND password = '" +
 60       + signUpBean.getPassword() + "' ";
 61     rs = stmt.executeQuery(searchQuery);
 62     boolean more = rs.next();
 63
 64     if (!more) {
 65       signUpBean.setValid(false);
 66     } else if (more) {
 67       signUpBean.setFirstName(rs.getString("FirstName"));
 68       signUpBean.setLastName(rs.getString("LastName"));
 69       signUpBean.setRole(rs.getString("role"));
 70       signUpBean.setPhone(rs.getString("phone"));
 71       signUpBean.setState(rs.getString("state"));
 72       signUpBean.setDistrict(rs.getString("district"));
 73       signUpBean.setVillage(rs.getString("village"));
 74       signUpBean.setGender(rs.getString("gender"));
 75       signUpBean.setDob(Date.valueOf(rs.getString("dob")));
 76       signUpBean.setValid(true);
 77     }
 78   } catch (Exception e) {
 79     e.printStackTrace();
 80   }
 81
 82   return signUpBean;
 83 }
 84 }
 85 }
```

ConnectionFactory



Umesh Kumar Sharma

Java EE - NREGAProject/src/factory/ConnectionFactory.java - Eclipse

```
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java EE
SignUpController.java AttendanceDAOImpl.java *EmployeeDAOImpl.java ProjectDAOImpl.java SignUpDAOImpl.java ConnectionFactory.java

1 package factory;
2 import java.sql.Connection;
3 
4 public class ConnectionFactory {
5     private static ConnectionFactory instance=new ConnectionFactory();
6     String url="jdbc:mysql://localhost:3306/nregaProject";
7     String user="root";
8     String password="123";
9     String driverClass="com.mysql.jdbc.Driver";
10    private ConnectionFactory(){
11        try{
12            Class.forName(driverClass);
13        }catch(ClassNotFoundException e){
14            e.printStackTrace();
15        }
16    }
17    public static ConnectionFactory getInstance(){
18        return instance;
19    }
20    public Connection getConnection(){
21        Connection connection=null;
22        try{
23            connection=DriverManager.getConnection(url,user,password);
24        }catch(SQLException e){
25            e.printStackTrace();
26        }
27        return connection;
28    }
29
30
31
32
33
34 }
```



# Certificate

This is to certify that Ms **BHUMIKA MEHTA** of **HERITAGE INSTITUTE OF TECHNOLOGY KOLKATA**, registration number:**131260110139** of 2013-14 , has successfully completed a project on **MNREGA Employment Management** using **JEE** under the guidance of **Mr Umesh Kumar Sharma**.

Mr Umesh Kumar  
Sharma.

# **Globsyn Finishing School**

*(a division of Globsyn Skills)*

Umesh Kumar Sharma



## Certificate

This is to certify that Mr DIPTA DAS CHOWDHURY of HERITAGE INSTITUTE OF TECHNOLOGY KOLKATA, registration number:**131260110141** of 2013-14 , has successfully completed a project on **MNREGA Employment Management** using **JEE** under the guidance of **Mr Umesh Kumar Sharma.**

---

-----  
Mr Umesh Kumar  
Sharma.

**Globsyn Finishing School**  
(a division of *Globsyn Skills*)

Umesh Kumar Sharma



## Certificate

This is to certify that Ms **SNIGDHA JAIN** of **HERITAGE INSTITUTE OF TECHNOLOGY KOLKATA**, registration number:**131260110178** of **2013-14** , has successfully completed a project on **MNREGA Employment Management** using **JEE** under the guidance of **Mr Umesh Kumar Sharma**.

---

Mr Umesh Kumar  
Sharma.

**Globsyn Finishing School**  
(*a division of Globsyn Skills*)

Umesh Kumar Sharma



## Certificate

This is to certify that Mr **AMIT KUMAR TIWARI** of **GOVT COLLEGE OF ENGINEERING AND CERAMIC TECHNOLOGY**, registration number: **121130110043 OF 2012-13**, has successfully completed a project on **MNREGA Employment Management** using **JEE** under the guidance of **Mr Umesh Kumar Sharma**.

---

-----  
Mr Umesh Kumar  
Sharma.

**Globsyn Finishing School**  
(*a division of Globsyn Skills*)

Umesh Kumar Sharma



# Certificate

This is to certify that Ms **SUBHAM GUPTA** of **GOVT COLLEGE OF ENGINEERING AND CERAMIC TECHNOLOGY**, registration number: **121130110078 OF 2012-13**, has successfully completed a project on **MNREGA Employment Management** using **JEE** under the guidance of **Mr Umesh Kumar Sharma**.

Mr Umesh Kumar  
Sharma.

# **Globsyn Finishing School**

*(a division of Globsyn Skills)*

Students are enthusiastic learner. Have integrated many advance tools and technologies. It shows the love and passion about computer programming.  
Good Luck to all of you my dear students! Have a great future ahead!

Umesh Kumar Sharma

