

References

- [1] 'Recruitment Process' Recruitment process of an IT company, 2012 [online]
Available: <http://recruitment.naukrihub.com/recruitment-process.html>
- [2] Human Resource Management, 2012 [online]
Available: http://en.wikipedia.org/wiki/Human_resource_management
- [3] Basic Object Oriented Concepts 2011 [online]
Available: <http://www.ipipan.gda.pl/~marek/objects/TOA/oobasics/oobasics.html>
- [4] Itris intelligent recruitment software, ITRIS support 2013 [online]
Available: <http://www.itris.co.uk/itris>
- [5] Structured Query Language 2013 [online]
Available: <http://en.wikipedia.org/wiki/SQL>
- [6] Recruiterbox software 2012 [online]
Available: <http://recruiterbox.com/recruitment-software-features/>
- [7] 'Recruiterbox' features, 2013 [online]
Available: <http://doeswhat.com/2011/05/28/interview-with-girish-redekar-ecruiterbox/>
- [8] Orange HRM, HR management software 2012 [online]
Available: <http://www.orangehrm.com/open-source-product-featuresrecruitment.shtml#recruit>
- [9] Php, 2013 [online]
Available: <http://php.net/>
- [10] Oracle Corporation, My SQL Technical support, 2011 [online]
Available: <http://www.mysql.com/support/>

[11] Jboss community, 2011 [online]

Available: <http://www.jboss.org/overview/>

[12] Oracle, 2011 [online]

Available: <http://www.oracle.com/index.html>

[13] Wendy and Michael, Boggs and Boggs, Mastering UML with Rational Rose – BPB Publications, 1999.

[14] Apache.org, Apache tomcat reference, 2011 [online]

Available: <http://tomcat.apache.org/>

[15] Java 2 Platform Enterprise Edition, 2012 [online]

Available: <http://www.webopedia.com/TERM/J/J2EE.html>

[16] Model-View-Controller, 2012 [online]

Available: <http://www.oracle.com/technetwork/articles/javase/index-142890.html>

[17] Roseindia.net, How Struts Works, 2008 [online]

Available: <http://www.roseindia.net/struts/>

[18] JQuery.com, JQuery API, 2012 [online]

Available: <http://api.jquery.com/category/core/>

[19] Relational persistent for Java, 2012 [online]

Available: <http://www.hibernate.org/>

[20] Ajax Tutorials, 2012 [online]

Available: <http://www.w3schools.com/ajax/>

[21] JDBC, 2013 [online]

Available: <http://www.webopedia.com/TERM/J/JDBC.html>

[22] Eclipse documentation, 2012 [online]

Available: <http://www.eclipse.org/documentation/>



Appendix A

Database Design

These are the scripts that will use to create tables in MySQL database schema. It's identified that minimum changes will be needed when using these in Oracle or SQL server databases.

```
CREATE TABLE RMSROLE (
```

```
    ROLEDID    NUMERIC    NOT NULL,  
    DESCRIPTION VARCHAR(80) NOT NULL,  
    PRIMARY KEY (ROLEDID));
```

```
CREATE TABLE RMSSTATUS (
```

```
    STATUSDID  NUMERIC    NOT NULL,  
    DESCRIPTION VARCHAR(80) NOT NULL,  
    PRIMARY KEY (STATUSDID));
```

```
CREATE TABLE RMSRANK (
```

```
    RANKID    NUMERIC    NOT NULL,  
    RANKCODE  VARCHAR(2) NOT NULL,  
    DESCRIPTION VARCHAR(80),  
    PRIMARY KEY (RANKID));
```

```
CREATE TABLE RMSINTERVIEWTYPE (
```

```
    INTERVIEWTYPEDID NUMERIC    NOT NULL,  
    TYPECODE          VARCHAR(20) NOT NULL,  
    DESCRIPTION        VARCHAR(80) NULL,  
    PRIMARY KEY        (INTERVIEWTYPEDID));
```

```
CREATE TABLE RMSVENUE (
```

```
    VENUEDID  NUMERIC    NOT NULL,  
    VENUECODE VARCHAR(20) NOT NULL,  
    DESCRIPTION VARCHAR(80) NULL,  
    PRIMARY KEY (VENUEDID));
```

```

CREATE TABLE RMSUSER (
  USERDID MEDIUMINT NOT NULL AUTO_INCREMENT,
  USERID VARCHAR(20) NOT NULL,
  NAME VARCHAR(80) NOT NULL,
  CONTACTNO VARCHAR(20) NOT NULL,
  EMAIL VARCHAR(50) NOT NULL,
  DESIGNATION VARCHAR(80) NOT NULL,
  ROLE NUMERIC NOT NULL,
  ISDISABLED VARCHAR(1) NOT NULL,
  PASSWORD VARCHAR(50),
  CREATEDAT DATETIME,
  PRIMARY KEY (USERDID),
  FOREIGN KEY (ROLE) REFERENCES RMSROLE(ROLEDID));

```

```

CREATE TABLE RMSFILE (
  FILEDID MEDIUMINT NOT NULL AUTO_INCREMENT PRIMARYKEY,
  MIMETYPE VARCHAR(80) NULL,
  FILENAME VARCHAR(80) NULL,
  ATTACHMENTOBJECT BLOB NULL,
  FLAG NUMERIC(1) DEFAULT 0 NOT NULL);

```

```

CREATE TABLE RMSAPPLICATION (
  APPDID MEDIUMINT NOT NULL AUTO_INCREMENT,
  FIRSTNAME VARCHAR(50) NOT NULL,
  LASTNAMES VARCHAR(80) NULL,
  NIC VARCHAR(20) NOT NULL,
  DOB DATE NULL,
  ISEMPLOYED NUMERIC(2) NULL,
  RANKID NUMERIC NOT NULL,
  EMAIL VARCHAR(50) NOT NULL,
  CONTACT_NO VARCHAR(20) NOT NULL,
  FILEDID MEDIUMINT NOT NULL,
  PRIMARY KEY (APPDID),
  FOREIGN KEY (RANKID) REFERENCES RMSRANK(RANKID),
  FOREIGN KEY (FILEDID) REFERENCES RMSFILE(FILEDID));

```

```

CREATE TABLE RMSVACANCY (
  VACANCYDID MEDIUMINT NOT NULL AUTO_INCREMENT,
  DESIGNATION VARCHAR(80) NOT NULL,
  POSITIONS NUMERIC NOT NULL,
  CRETAEDAT DATE NULL,
  EXPIREDAT DATE NULL,
  PRIMARY KEY (VACANCYDID));

```

```

CREATE TABLE RMSINTERVIEW (
  INTERVIEWDID MEDIUMINT NOT NULL AUTO_INCREMENT,
  INTERVIEWTYPEDID NUMERIC NOT NULL,
  VACANCYDID MEDIUMINT NOT NULL,
  APPDID MEDIUMINT NOT NULL,
  PRIMARY KEY (INTERVIEWDID),
  FOREIGN KEY (INTERVIEWTYPEDID) REFERENCES RMSINTERVIEWTYPE (INTERVIEWTYPEDID),
  FOREIGN KEY (VACANCYDID) REFERENCES RMSVACANCY(VACANCYDID),
  FOREIGN KEY (APPDID) REFERENCES RMSAPPLICATION(APPDID));

```

```

CREATE TABLE RMSSCHEDULE (
  SCHEDULEDID MEDIUMINT NOT NULL AUTO_INCREMENT,
  INTERVIEWDID MEDIUMINT NOT NULL,
  USERDID MEDIUMINT NOT NULL,
  DATE_TIME DATE NOT NULL,
  VENUEDID NUMERIC NOT NULL,
  PRIMARY KEY (SCHEDULEDID),
  FOREIGN KEY (INTERVIEWDID) REFERENCES RMSINTERVIEW(INTERVIEWDID),
  FOREIGN KEY (USERDID) REFERENCES RMSUSER(USERDID),
  FOREIGN KEY (VENUEDID) REFERENCES RMSVENUE(VENUEDID) );

```

```

CREATE TABLE RMSNOTE (
  NOTEDID MEDIUMINT NOT NULL AUTO_INCREMENT,
  APPDID MEDIUMINT NOT NULL,
  INTERVIEWDID MEDIUMINT NULL,
  DESCRIPTION VARCHAR(80),
  PRIMARY KEY (NOTEDID),
  FOREIGN KEY (APPDID) REFERENCES RMSAPPLICATION(APPDID);

```


Appendix B

System Implementation

Segment Struts-config.xml file

```
<struts-config>
  <form-beans>
    <form-bean name="loginForm" type="com.rms.action.LoginForm"/>
  </form-beans>
  <global-forwards/>
  <global-exceptions/>
  <action-mappings>
    <action path="/Login"
      type="com.rms.action.LoginAction"
      validate="false"
      parameter="login"
      scope="session"
      name="loginForm">
      <forward name="success" path="login.success"/>
      <forward name="failure" path="/failed.jsp"/>
    </action>
  </action-mappings>
  .....
</struts-config>
```

Segment of Tiles-def.xml file

```
<! DOCTYPE tiles-definitions PUBLIC
"-//Apache Software Foundation//DTD Tiles Configuration 1.3//EN"
"http://struts.apache.org/dtds/tiles-config_1_3.dtd">
<tiles-definitions>
  <definition name="common.layout" path="/common_layout.jsp">
    <put name="header" value="/header.jsp" />
```

```
<put name="body" value="/body.jsp" />
<put name="footer" value="/footer.jsp" />
</definition>
<definition name="common. layout" path="/common_layout.jsp">
  <put name="header" value="/anotherHeader.jsp" />
  <put name="body" value="/anotherBody.jsp" />
  <put name="footer" value="/anotherFooter.jsp" />
</definition>
<tiles-definitions>
```

Login page for RMS



User name:

Password:

Login

Rank.hbm.xml mapping segment for rmsRank table.

```
<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping PUBLIC
"-//Hibernate/Hibernate Mapping DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd" >
<hibernate-mapping>
<class name="com.rms.model.Rank" table="RMSRANK" >
    <id name="rankID" type="java.lang.Long" column="RANKID"
        unsaved-value="null" >
        </id>
<property name="rankCode" type="java.lang.String"
        column="RANKCODE" />
<property name="description" type="java.lang.String"
        column="DESCRIPTION" />
</class>
</hibernate-mapping>
```

Segment of POJO class for the rank.hbm.xml mapping file

```
public class Rank {
    private Long rankID;
    private String rankCode;
    private String description;

    public Long getRankID() {
        return rankID;
    }
}
```

```
}  
public void setRankID(Long rankID) {  
    this.rankID = rankID;  
}  
public String getRankCode() {  
    return rankCode;  
}  
public void setRankCode(String rankCode) {  
    this.rankCode = rankCode;  
}  
public String getDescription() {  
    return description;  
}  
public void setDescription(String description) {  
    this.description = description;  
}  
}
```

Appendix C

System Documentation

System Name : - Recruitments Management System (RMS).

Version : - 1.0

System configuration for the application as follows.

Software Requirements

Requirement	Description
Operating System	MS Windows XP, Linux
Web sever	Apache Tomcat, Jboss
Database Management System	MySQL server 5x
Internet Connection	Broad band/ADSL with high(2Mb or above) upload capacity

Hardware Requirements

	Minimum	Recommended
Processor	2.4 GHz	3 GHz Or above
RAM	1 GB	2 GB or Above
Hard Disk	40 GB	80 GB or Above
Graphics Adapter	16 MB	32 MB or Above

Every client PC should support for Internet Explorer FireFox. Best performance from FireFox.

Users PC's consist of following configurations.

Software Requirements

Requirement	Description
Operating System	MS Windows XP, Linux
Applications	Java Script enabled browsers.

Hardware Requirements

	Minimum	Recommended
Processor	1.3 GHz	1.6 GHz Or above
RAM	512 MB	1 GB or Above
Hard Disk	30 GB	40 GB
Graphics Adapter	16 MB	32 MB or Above

Installation

- Configure the database with running all the required scripts and Start the database server (MySQL Server)
- If the server is Jboss 4.0.5.GA Copy the 'rms.war' file into the \jboss-4.0.5.GA\jboss-4.0.5.GA\server\default\deploy folder.
- Start the server
- If the server is Apache tomcat copy the 'rms.war' file into \Apache Software Foundation\Tomcat 6.0\webapps folder.
- Start the server

System URL

<http://localhost:8080/rms/>

** Note: Host name is depending on the deployment server. Port number can be configured by editing *server.xml* in <server>\conf\server.xml.

Appendix D

Test Cases

Test cases for Login page

No	Test Case	Expected Result
4.1	User enters user name and/or password incorrectly and clicks on login button.	User gets the error message of "password Incorrect! Please try again."
4.2	User clicks on login button with black username and password.	User gets the error message of "Please enter valid user name and password!"

Test cases for Add application

No	Test Case	Expected Result
4.3	Enter details and clicks on add application button.	Application will save and success message displays.
4.4	Click on add application button with blank data or blank required fields.	Error message of "Please enter necessary details"

Test cases for search applications

No	Test Case	Expected Result
4.5	Enter details and clicks on search button.	List applications according to the keywords. Ex: name, NIC
4.6	Click on search button with blank data.	List all applications sorted by date in descending order

Test cases for Edit application

No	Test Case	Expected Result
4.7	Click on application in search results.	Application details will be displayed.
4.8	Click on save button in edit applications page	If the required fields are empty, error message displays If required fields filed correctly, application will save and success message displays.

Test cases for Receive applications

No	Test Case	Expected Result
4.9	Clicks on receive applications button	Received applications will list down.
4.10	Click on add application button with blank data or blank required fields.	Error message of " Please enter required details"

Test cases for Delete applications

No	Test Case	Expected Result
4.15	Click on delete icon on particular application in search results.	Popup appear get confirmation on deletion of the particular application.
4.16	Click on delete button in popup in delete application	Application will remove from the list and the system.

Test cases for Add vacancy

No	Test Case	Expected Result
4.11	Enter details and clicks on add vacancy button.	Application will save and success message displays.
4.12	Click on add application button with blank data or blank required fields.	Error message of " Please enter required details"

Test cases for search vacancies

No	Test Case	Expected Result
4.13	Enter details and clicks on search button.	List applications according to the keywords. Ex: name, date
4.14	Click on search button with blank data.	List all applications sorted by date in descending order

Test cases for Edit vacancies

No	Test Case	Expected Result
4.15	Click on particular vacancy in search results.	Vacancy details would display.
4.16	Click on save button in edit vacancy page	If the required fields are empty, error message will display If required fields filled correctly, application will save and success message will display.

Test cases for Delete vacancies

No	Test Case	Expected Result
4.15	Click on delete icon on particular vacancy in search results.	Popup appear get confirmation on deletion of the particular vacancy.
4.16	Click on delete button in popup in delete vacancy	Vacancy will remove from the list and the system.

Appendix E

User Evaluation Criteria

5=Excellent, 4=Good, 3=Average, 2=Fair, 1=Poor

Description	Marks
Easy to use	
Messages / Errors clearly described	
The reports are clear and useful	
The color combination	
Response time	
Functional requirements	
Secure Intellectual Property	
Maintenance of the system	
Comments:	



Appendix F

User Manual

REGISTRATION

User must be get register in the system by contacting the *administrator* of the system. Name, email, contact no, designation and a username should be provided when registering. After registration he or she will assign to a role when according to the designation.

LOIGIN

A faint background illustration shows a hand with the index finger pointing towards a computer monitor. The monitor is on a desk, and a keyboard is partially visible to the left.

User name:

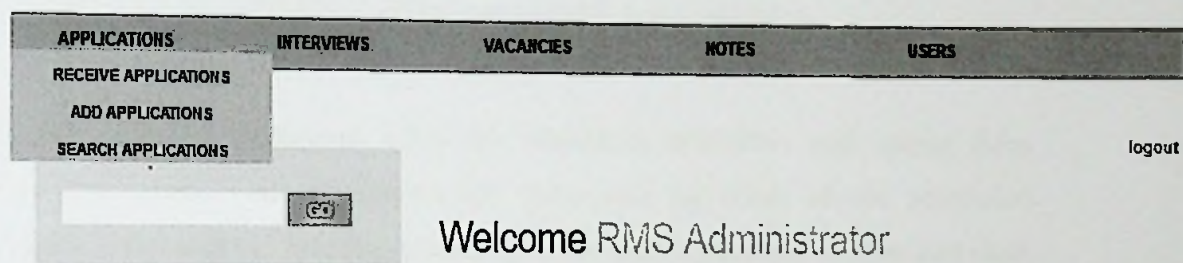
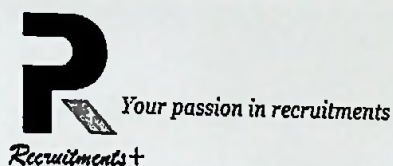
Password:

User can login to the system via login page. If the user forgot his/her username and password both, he/she should contact system administrator for re registration process.

LOGOUT

User can be logout from the system directly by closing the web page or by clicking the *logout* link at the top right corner of every page.

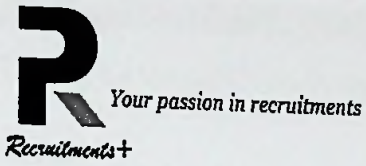
APPLICATIONS



User can check the applications by clicking Receive applications sub menu in applications. Then the received applications can be added to the system by add applications submenu. Further users can add applications directly using this sub menu by giving some required details following the particular curriculum vitae. Clicking on the save button will save the information with the system.

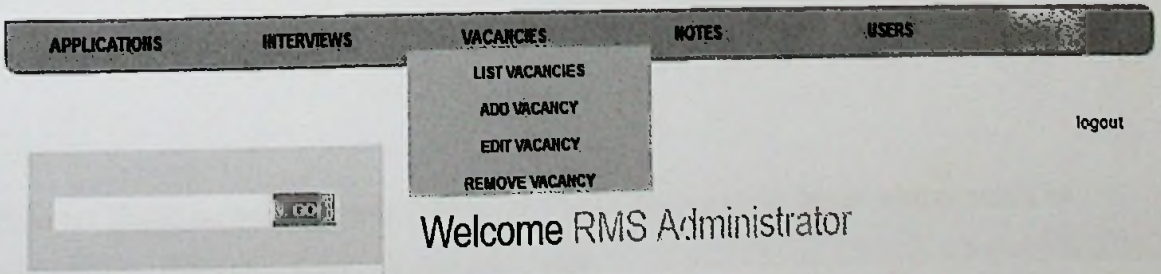
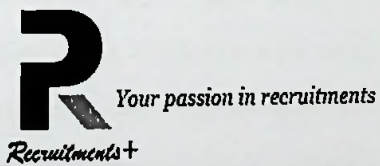
If any particular wants to list currently saved applications, it can be obtained by clicking on list applications sub menu. Click on search button with empty details will list down all the applications without any filtering. Particular application can be edited by clicking on the details in above list. User can save edited application by clicking save button at the bottom of edit application page.

INTERVIEWS



Users can schedule interviews, view the scheduled interviews and update them through this menu. View interviews sub menu will list down all the scheduled interviews with / without filtering according to the search parameters. User can click on any particular interview, if he/she wanted to view it. User can edit schedules by clicking on the edit icon in the right hand corner or or update interviews sub menu.

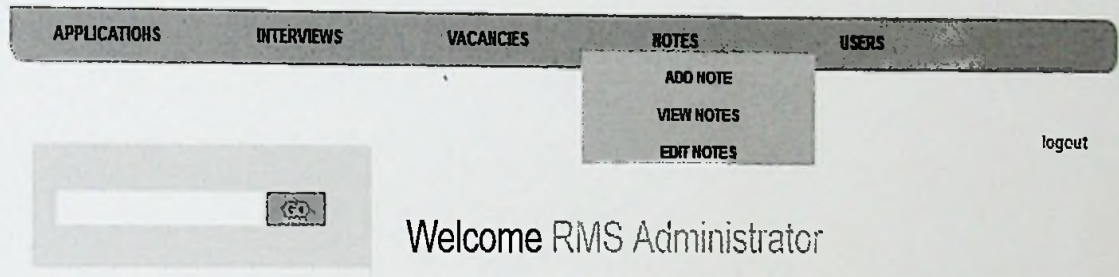
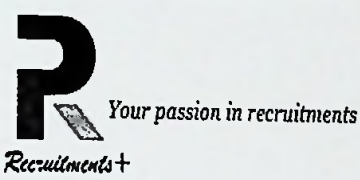
VACANCIES



Users can handle interview details through this menu. List vacancies will list down all the vacancies exist for the current date or user can limit the list by giving additional

parameters. If the user wanted to add new vacancy or list of vacancies, he/she can use add vacancy sub menu. Edit vacancy sub menu help the user to search particular vacancy and edit the details. Clicking on the save button in edit vacancy window will save the updated details. Further edit option is available through the list vacancies too. Clicking on the edit icon in the right hand corner will directed to edit vacancy page.

NOTES



Notes can be added against the applications in interviews via notes menu. Add note used to add a new note. User should select relevant interview and the application following with the note text. User can view the existing notes using view notes sub menu. User should provide the interview id or application id when searching a particular note. User can edit a particular note by clicking on the edit icon in notes list or using edit notes sub menu.

**** Note:** These menus are accessible according to the user role. Some menus won't be accessible to *executive, lead* and *manager* roles.

