# Assessment Parameters

* Complete flow of the application  with validation and exception handling -70%
* Comments/best practice, coding standards – 10%
* Execution of the application (Output) – 20%
* ScreenShot should be submitted along with the solution
* The solution(Project) created by the trainee should have the name like AppName\_Empid Ex:ABCCorp\_675467
* Code with compilation errors will not be considered for evaluation

# *Global Recruitments Organization*

**Problem Statement**: Global Recruitments is a recruitment organization. It needs a software to store all the contact details of the enquiries visiting the office. Allowing to store the contact number, domain interested,etc.. At any moment the organization should be able to fetch the contact details of enquiry on the unique Id allocated.

**Software Details:**

**Section: 1**

* User is allowed to add Enquiry details as follows:
  + First Name
  + Last Name
  + Contact Number
  + Preferred Domain
  + Preferred Location
* Organization is allowed to search the Enquiry based on the enquiry Id generated by the system.
* Sample Screens are as follows.

**Sample Screen**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Global Recruitments\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Choose an operation**

**1. Enter Enquiry Details**

**2. View Enquiry Details on Id**

**3. Exit**

**Please enter a choice:**1

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Enter First Name :**Ranbir

**Enter Last Name :**Gupta

**Enter Contact Number :**9898990004

**Enter Preferred Domain :**Java

**Enter Preferred Location :**Pune

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Thank you Ranbir Gupta your Unique Id is 1001 we will contact you shortly.**

**Choose an operation**

**1. Enter Enquiry Details**

**2. View Enquiry Details on Id**

**3. Exit**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Please enter a choice :** 2

**Enter the Enquiry No . :** 1001

**Id First Name Last Name Contact No. Preferred Domain Preferred Location**

1001 Ranbir Gutpa 9898990004 Java Pune

**Choose an operation**

**1. Enter Enquiry Details**

**2. View Enquiry Details on Id**

**3. Exit**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Please enter a choice :**3

**Thank you selecting us!!**

**Section 2:**

**Following validations should be taken care during entering Enquiry details:**

1. **Contact number should be a 10 digit valid mobile number.**
2. **First name, Preferred Location and Domain cannot be empty and should be alphabets only.**

**Marks Distribution:**

|  |  |
| --- | --- |
| Parameters | Marks |
| **DML** | **30** |
| Adding Enquiry details in Enquiry HashMap | 10 |
| Display operation menu | 3 |
| Prompt asking Enquiry details | 3 |
| Display the success message with Enquiry detail and unique enquiry ID | 7 |
| Service + Bean | 7 |
| **Retrieval** | **13** |
| Display Enquiry details on the Enquiry ID entered | 7 |
| Display error message on incorrect Enquiry ID | 3 |
| Service + Bean | 3 |
| **Common Parameters** | **17** |
| Contact number should be a 10 digit valid mobile number. | 4 |
| First name, Last Name, Preferred Location and Domain cannot be empty and should accept only alphabets. | 4 |
| Exception Handling | 4 |
| Common best practices and coding standards | 5 |
| **Junit** | **10** |
| **Total** | **70** |

**Section 3. Write JUNIT test cases for the methods in the dao class [10 Marks]**

**Note : Use Math.random() method to generate Id.**

**Classes to be created:**

**com.capgemini.contactbook.ui**

class client

{ public static void main ()

{

// User Interface which display the operation menu to add an enquiry, search an enquiry.

// Create object for service and execute the respective methods

}

}

**com.capgemini.contactbook.bean**

public class EnquiryBean

{

Private long enqryId;

private String fName;

private String lName;

private long contactNo;

private String pLocation;

private String pDomain;

}

**Service utility classes and Interfaces :**

**com.capgemini.contactbook.service**

**public interface ContactBookService**

**{**

public long addEnquiry(EnquiryBean bean) throws ContactBookException;

public EnquiryBean getEnquiryDetails(long EnquiryID) throws ContactBookException;

public boolean isValidEnquiry(EnquiryBean bean) throws ContactBookexception;

**}**

**public class ContactBookServiceImpl implements ContactBookService**

**{**

public long addEnquiry(EnquiryBean bean) throws ContactBookException{

}

Public EnquiryBean getEnquiryDetails(long EnquiryID) throws ContactBookException{

}

Public boolean isValidEnquiry(EnquiryBean bean) throws ContactBookexception{

//validateContactNo

// validateFirstName

// validateLastName

// validatePLocation

// validatePDomain

}

**}**

**com.capgemini.contactbook.dao**

**public interface ContactBookDao**

**{**

public long addEnquiry(EnquiryBean bean) throws ContactBookException;

public EnquiryBean getEnquiryDetails(long EnquiryID) throws ContactBookException;

**}**

**public class ContactBookDaoImpl implements ContactBookDao**

**{**

**private Map<Long,EnquiryBean> enquiryEntry; // use HashMap**

**// DAO implementations**

**}**

Add appropriate user defined exception classes and any other supporting classes required.