**MINI CASE REPORT**

**Problem statement:**

**------------------------------ Operation for Training Institute Application---------------------------------**

Create the three roles:-

* Admin
* Institute
* Student

Functionalities are performed by the Admin in Training Institute Application are

* Admin Login.
* Training Institute Registration.
* Admin has to approve the institute.
* Admin has to check the details of institute and student.
* Admin has to access the institute and delete and view the complaint on it.

Functionalities are performed by the Institute in Training Institute Application after the registration done the admin operations are

* Institute Login.
* Handle the student details and view it.
* Upload any article in the Institute page.
* Update institute profile.
* After student has to send a request to institute and send response to student request.
* Institute only access of about student details and view feedback.

Functionalities are performed by the Student in Training Institute Application after the registration done the Institute operations are

* Register the student details and send request to colleges.
* After approval Students can login credentials with user id & password.
* Student can update they details.
* Checking the availability of the number of seats and no of courses.
* After sending request view response.
* Verify the availability status has to the either approved or not.
* Student can send a feedback to teaching performance.

Entities are:

* Admin
* Institute
* Student
* Faculty
* Complaint
* Upload Article
* Feedback

**Requirements:** Eclipse Software , Tomcat server software, Mysql workbench or Wampserver, jar files.

**Implementation:**

* By using Dynamic web project created the Using MVC method
* Creating a Databases and Fields by using Mysql for Training Institute Application.
* Then using JDBC(Java DataBase Connectivity) we connection to it in eclipse.
* With all needs of login to for mysql URL, Username, Password is required.
* Using Try Catch ClassNotFoundexception isuse because when jdbc driver is using its occur an exceptions so we implement.
* Then coming to our Training institute case study when implementation all the operations, driver connection etc… For this we can separate it like, Domain Package DBConnection Package, Main Class Package, DAO(DataAccessObject) Package, Db framework package and add referenced libraries.
* These all separate we can able to clear understand without any error.

**POJO CLASS RULES:**

* Pure Old Java Object
* Class must be public.
* Variable must be private.
* Must having public default constructor.
* Can have arguments Constructor.
* Every field have public getter and setter method.
* Increase the readability and reusability of a program.
* Getter is accessor where update a value of variable.
* Setter is mutator where read value of variable.

**Implementation:**

* By using Model View Controller Method we create a dynamic web design project.
* And Install and work on Tomcat server because using webservlet.
* Then create packages according.
* After the functionalities performing the all the roles in db connection we can creating the pojo class in java.
* First in main class we have give roles and Objects creation using switch case with userinput , With all operations connection will be in main class.
* Login\_details are create classes and fields by using private then use getter and setter to access and read the value.
* Then Data AccessObject classes can give the role, username, password with DBconnection the Check by database if correct values execute query with successful the display shown Database found and matched.
* After completion DAO Operations for that particular classes and by using db framework methods (SQLMapper, Parammapper, MapRow, ResultSet).
* What ever the operations is display to execute query by using statement and Mysql insert, select, delete, update, alter all these will be control and executed by operations in it.
* Then all coding implemenatation is done and in Main class imports all the package and classes.
* Then execute the code successfully.