Global Journal of Engineering Science and Research Management

ANDROID BASED SECURED PHOTO IDENTIFICATION SYSTEM USING DIGITAL WATERMARKING

Prof.Abhijeet A.Chincholkar [*1], Ms.Najuka B.Todekar [2], Ms.Sunita V.Ghai[3]
[*1]M.E. Digital Electronics, JCOET Yavatmal, India.
[2] UG Student, B.E.Computer Engg. Dept .JDIET, Yavatmal, India.
[3] UG Student, B.E.Computer Engg. Dept .JDIET, Yavatmal, India.
*Correspondence Author: chincholkarabhijeet@gmail.com

Abstract

With the development of information technique, the power of image processing software become stronger, and it can easily edit the digital images more easily. So the copyright protection, the integrity authentication of digital image becomes an urgent issue. Digital watermarking is an adaptation of the commonly used and well-known paper watermarks to the digital world. Digital watermarking describes methods and technologies that allow hiding of information, for example a number or text, in digital media, such as images, video and audio. With the use of this digital watermarking technology our aim will be to generate a Digital Camera Tool for Android System. This tool can digitally watermark any photo captured by using Camera with a Hidden signature as Mobile number. This signature will be encrypted and then this encrypted signature will be added as watermark. Whenever verification is required the tool will extract Mobile Number from the Photo.

Introduction

Creating database applications has become an essential part of any Developer's work. Although creating a form to display table and database information isn't difficult, it can take time. In existing application we are creating databases in the different types of database system. In each particular database system it has special enterprise managers, in which it creates different type of tools to manage the database. In such types of database enterprise managers we have need to learn each and every tools to manage the databases.

In our project is using different databases on one platform. The user does not need to use the different tools for database. By using this software that is enterprise manager we can operate different databases as per user choice. This tool will internally use JDBC to perform low level DBMS operations on database. It will use metadata classes of JDBC to retrieve information about database. This tool will initially ask for connect information such as Driver Name, DSN, USERID, PASSWORD etc. and will connect will that database. Once it is connected we can interactively perform DBMS operations such as creation of tables, deletion, updation, searching.

It can generate different types of Database forms by using following modules.
  a)   Object Browser: It will project list of Database Objects such as tables, views, users etc. In Hierarchical order. If an database object is selected it will display information about it as well as its contents.
  b)   Query manager: It will be sql+ type of tool which will send our queries to database for execution & will display result.
  c)   Data Form: This module will provide Data Form. These database forms will automatically generated for different tables. It will perform all operations such as Search, move next, move previous, move first, move last , add, edit, delete etc.

Global Journal of Engineering Science and Research Management

d) Table wizard: It will provide a wizard to generate table. It will interactively ask information about table such as column name, data types, their sizes, primary key etc. It will internally generate a create table statement & will send it to database for execution.

Creating databases is a very tedious job when we connect with any application. We use the Open Database Connectivity (JDBC-ODBC).In our project we use the java methods i.e. Runtime ( ) to call the JDBC-ODBC Bridge driver in our application. So we not need to call this bridge externally.

The input to the project will be wizard answering to questions which   Contain detail about the forms like database, DSN, UserID, password, table name, etc. In this project user maintain connectivity of different databases. User must to install the databases which he wants to use on his system.

Literature Review

Oracle Enterprise Manager (hereafter referred to as Enterprise Manager) is the Oracle integrated management solution for managing the complete Oracle environment. All services within your enterprise, including hosts, databases, listeners, application servers, HTTP Servers, and Web applications, are easily managed as one cohesive unit. This chapter acknowledges the challenges you face as a database administrator and states how Enterprise Manager can help you meet these challenges.

Oracle Enterprise Manager Architecture consists of a three-tier framework. Unlike a two-tier client-server structure, where a client tier responsible for the presentation of information accesses data from a server, Oracle Enterprise Manager utilizes a three-tier architecture which includes:
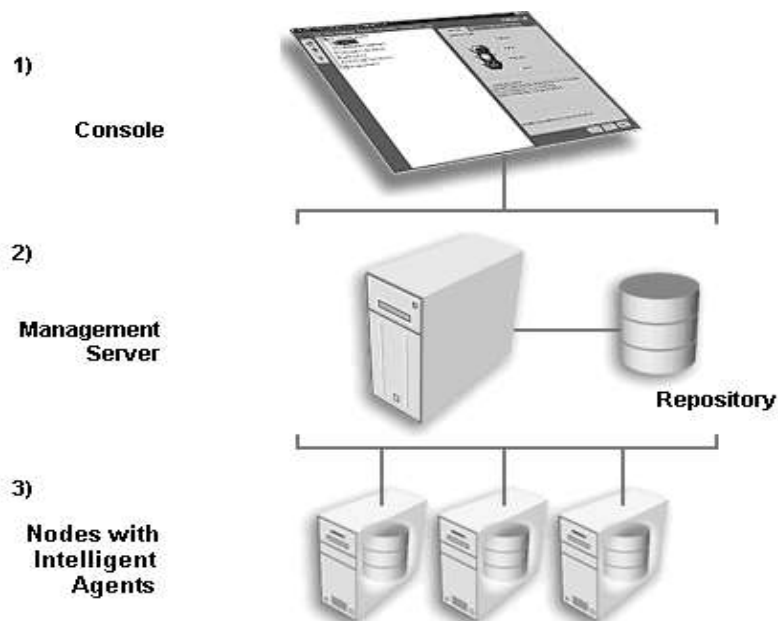

Figure 1 Three-Tier Architecture

Global Journal of Engineering Science and Research Management

The first tier of Oracle Enterprise Manager is comprised of clients such as consoles and management applications, which present a graphical user interface to administrators for all management tasks. These client components can be installed locally or brought up with a web browser.

The second tier of the Oracle Enterprise Manager framework is comprised of one or more Oracle Management Servers. The Management Server is the core of the Enterprise Manager framework. It provides administrative user accounts, processes management functions such as jobs and events, and manages the flow of information between the Console (first tier) and the nodes with Intelligent Agents (third tier).

The Oracle Management Server uses a repository to store all system data, application data, information about the state of managed nodes, and information about any system management packs. A repository is a set of database tables that must be located in a supported Oracle database accessible to the Oracle Management Server.
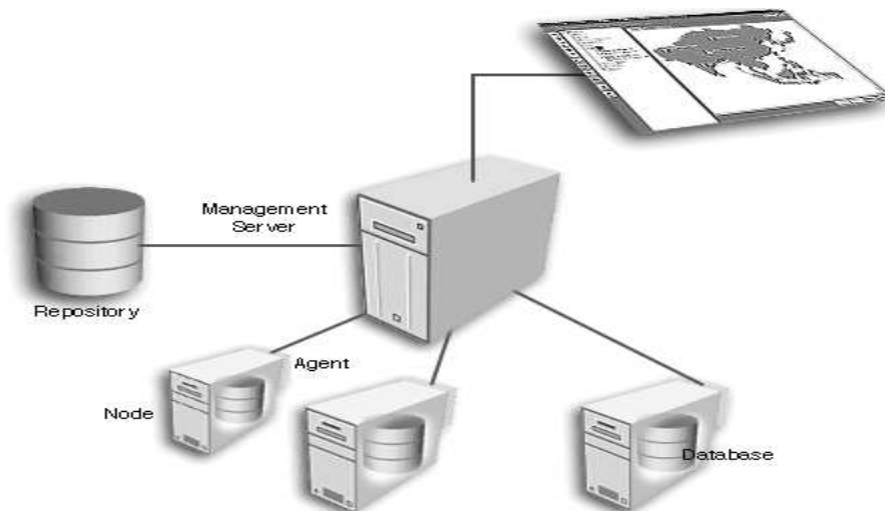
Figure 2 Management Server

Java is loosely based on C++ syntax, and is meant to be Object-Oriented Structure of java is midway between an interpreted and a compiled language. The java compiler into Byte Codes, which are secure and portable across different platforms, compiles Java programs. These byte codes are essentially instruction encapsulated in single type, to what is known as a java virtual machine (JVM), which resides in standard browser. JVM verifies these byte codes when download by the browser for integrity. JVM is available for all OS. JVM converts these byte codes into machine specific instructions at runtime. Java Features are Simple, Object oriented, Distributed, Compiled and Interpreted, Robust, Secure, High performance, Multithreaded, Dynamic, Platform independent and Portable, Exception and Exception Handling.

Global Journal of Engineering Science and Research Management

Oracle is one of the most widely used database system in the world. It runs virtually on all platforms ranging from Windows-based PCs to UNIX servers. It also comes with an array of programming tool and environments and provides access to database from a variety of programming languages.

In recent years, more and more universities in the United States and elsewhere are using Oracle in their database courses as the primary vehicle to illustrate database concepts and principles. This has resulted in the need for a concise book on Oracle programming to supplement the traditional text in the database courses.

System Overview

JDBC is a Java API for executing SQL statements. It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. In other words, with the JDBC API, it isn't necessary to write one program to access a Sybase database. The combination of Java and JDBC lets a programmer write it once and run it anywhere.

Java, being robust, secures, easy to use, easy to understand, and automatically downloadable on a network, is an excellent language basis for database applications. What is needed is a way for Java applications to talk to a variety of different databases. JDBC is the mechanism for doing this.

The JDBC API supports both two-tier and three-tier models for database access. In the two-tier model, a Java applet or application talks directly to the database. This requires a JDBC driver that can communicate with the particular database management system being accessed. A user's SQL statements are delivered to the database, and the results of those statements are sent back to the user. This is referred to as a client/server configuration.

In the three-tier model, commands are sent to a "middle tier" of services, which then send SQL statements to the database. The database processes the SQL statements and sends the results back to the middle tier, which then sends them to the user. In many cases the three-tier architecture can provide performance advantages.

Global Journal of Engineering Science and Research Management
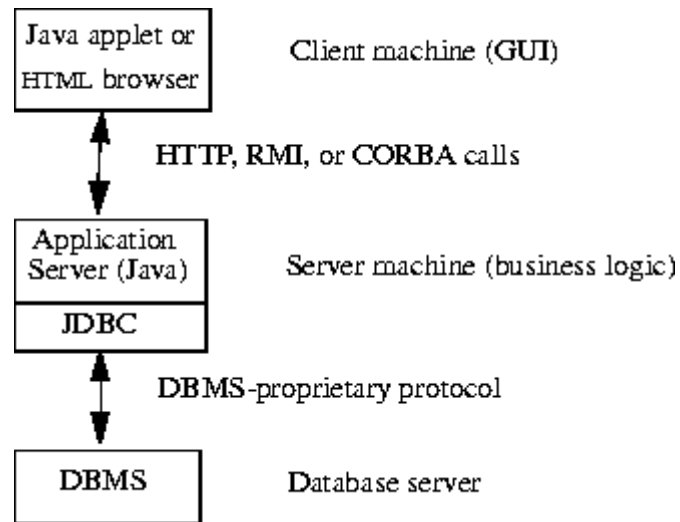
Figure 3 Three Tier Model

Java Soft provides three JDBC product components as part of the Java Development Kit (JDK).

1. The JDBC driver manager: - The JDBC driver manager is the backbone of the JDBC architecture. It actually is quite small and simple; its primary function is to connect Java applications to the correct JDBC driver and then get out of the way.
2. The JDBC driver test suite: - The JDBC driver test suite provides some confidence that JDBC drivers will run your program. Only drivers that pass the JDBC driver test suite can be designated JDBC COMPLIANTTM.
3. The JDBC-ODBC Bridge: - The JDBC-ODBC Bridge allows ODBC drivers to be used as JDBC drivers. It was implemented as a way to get JDBC off the ground quickly, and long term will provide a way to access some of the less popular DBMS if JDBC drivers are not implemented for them.JDBC drivers are not implemented for them.

Global Journal of Engineering Science and Research Management
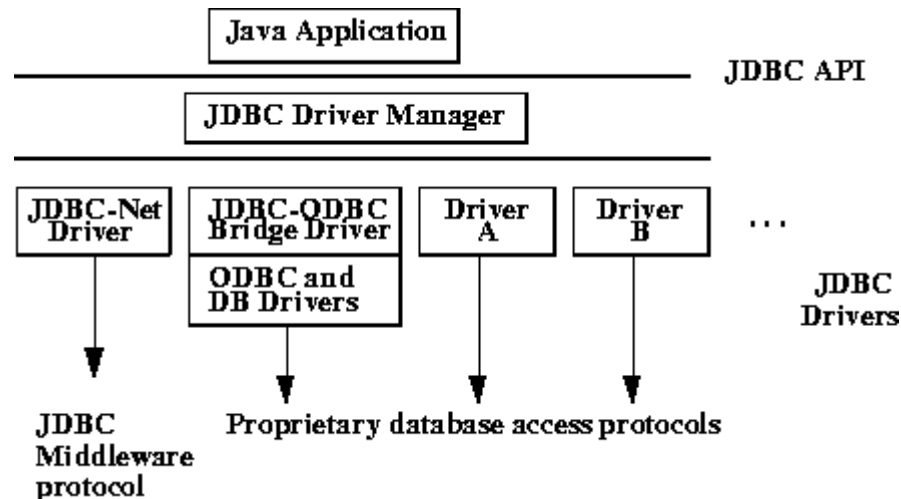
Figure 4 Java Soft Frameworks

Implementation

For creating different forms user have to select some inputs. Those inputs are converted into XML file and this XML file is replaced in templates, which is written in java. It is generic structure of java class.

a) DSN (Data Source Name): -This is identification of dataset. It allows to the driver for the connection. Various database have different steps for creating DSN. It is an interface between databases and application.

b) 7.2 User ID and Password: -For the security purpose some database have user id and password like in Oracle is Scott, tiger respectively.

c) 7.3 Table name: -Table name identification of the data, which is present in tabular format. User has to select the specified table name. Columns are field presents in table. According to requirement user have to select it

d) 7.4 Primary Key: -It uniquely identifies the entities within an entity set. A key is property of entity set rather than of individual entities.

Global Journal of Engineering Science and Research Management

Figure 5 create DSN

Global Journal of Engineering Science and Research Management
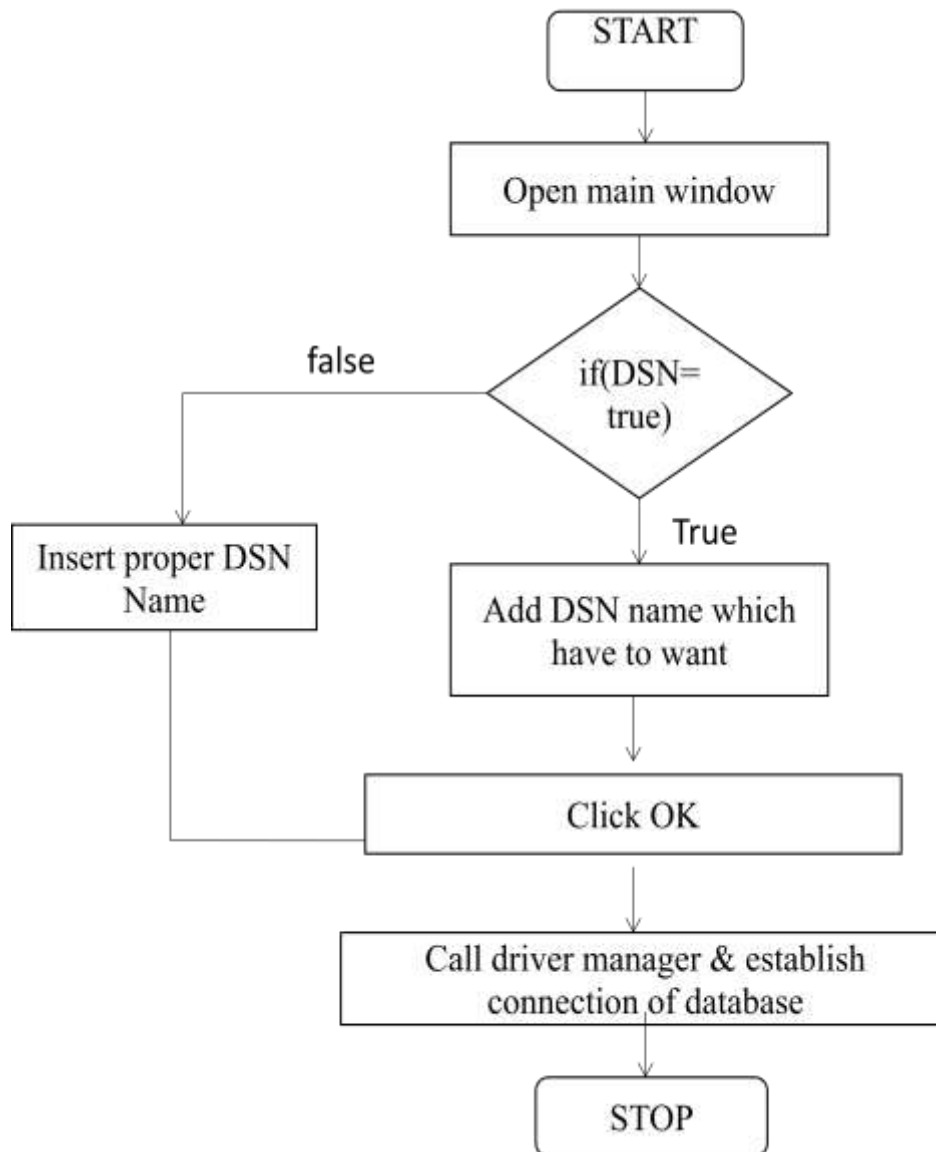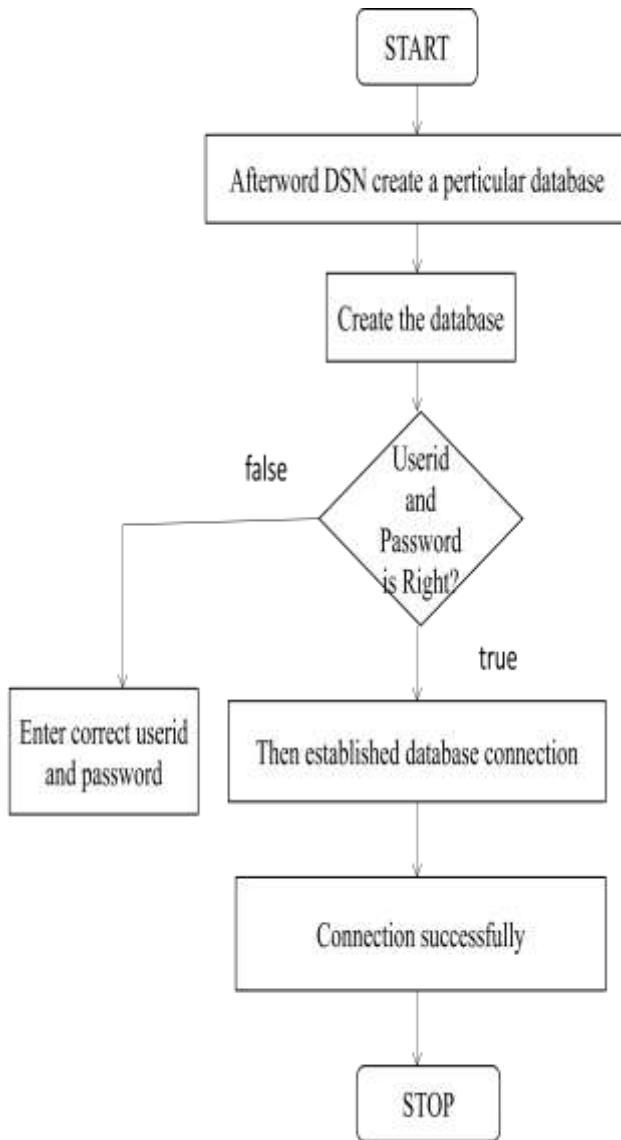
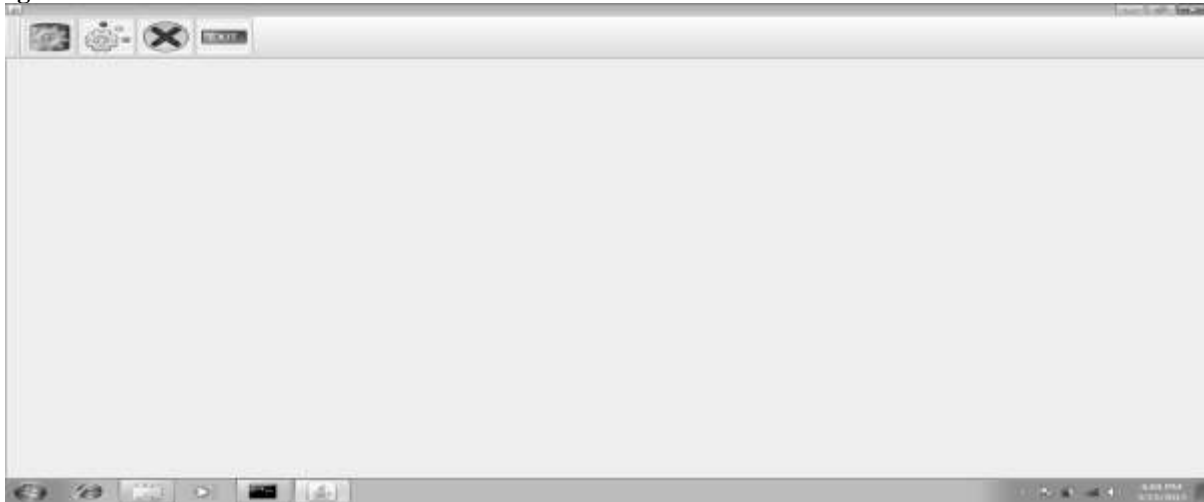*Figure 6 Login to Created DSN*   *Figure 7 Connecting with Database*
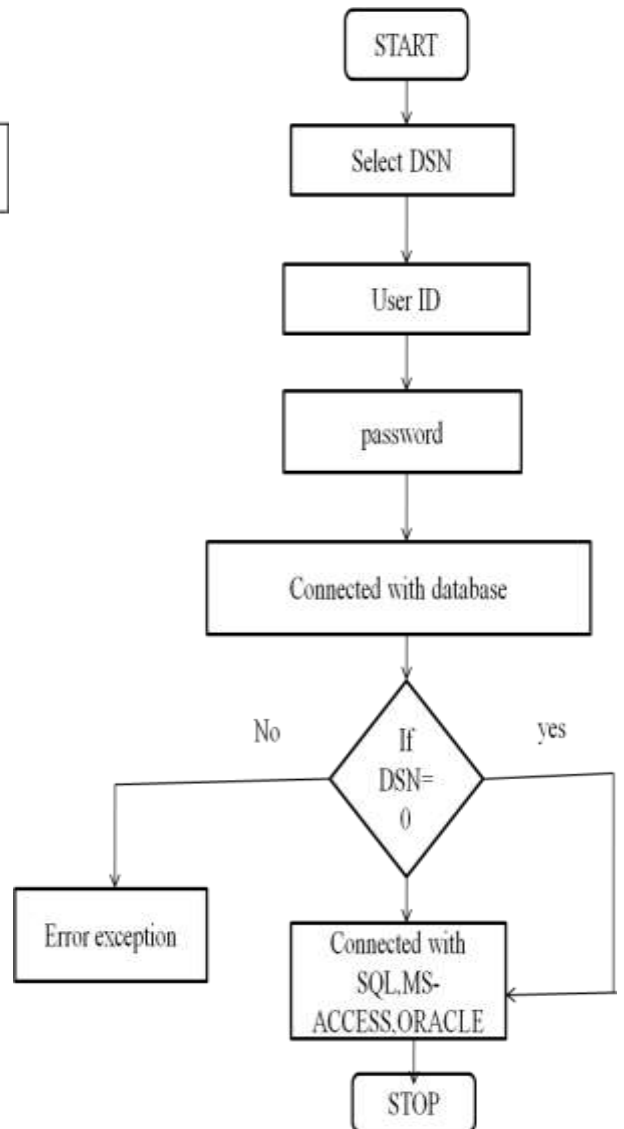
Global Journal of Engineering Science and Research Management
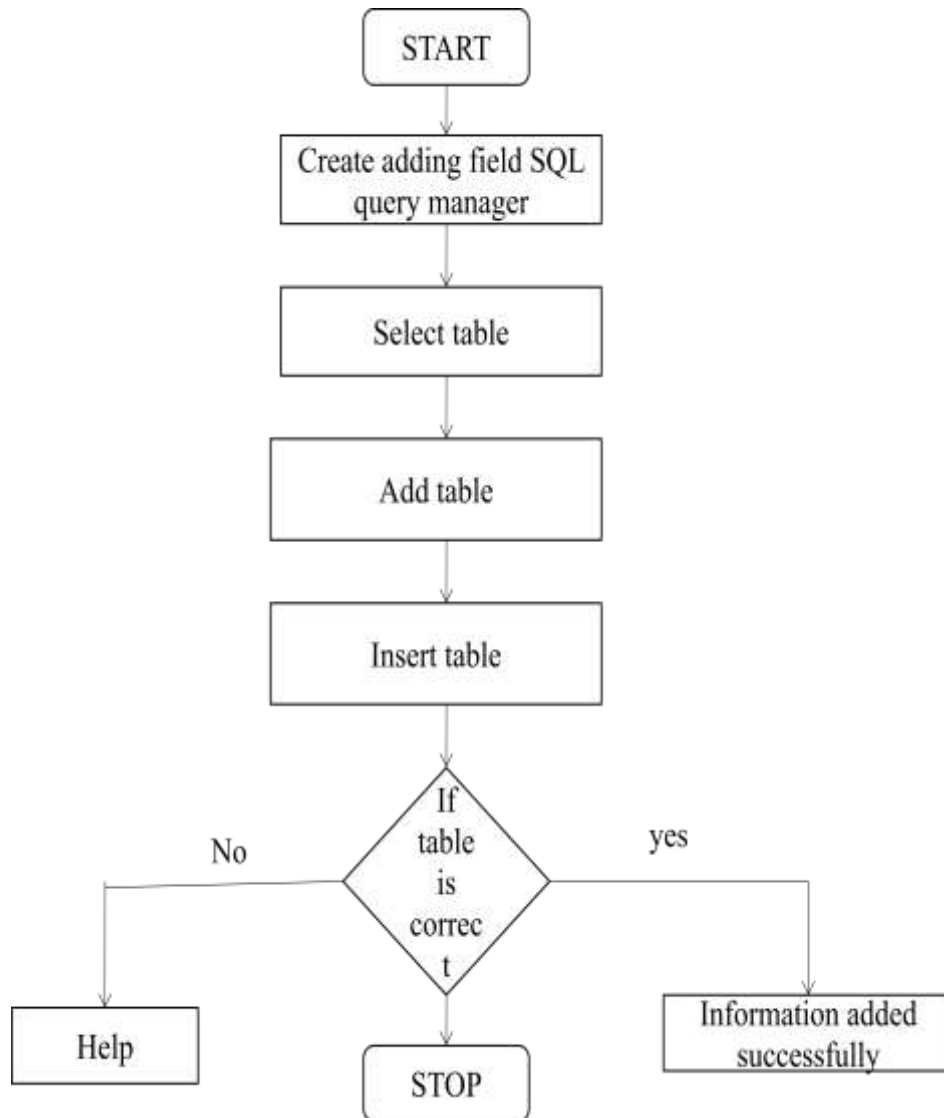
Fig.9.4 Inserting Information

Global Journal of Engineering Science and Research Management

Runtime Snapshots

1. **Opening Window**

2. **Create DSN for Database**

Global Journal of Engineering Science and Research Management
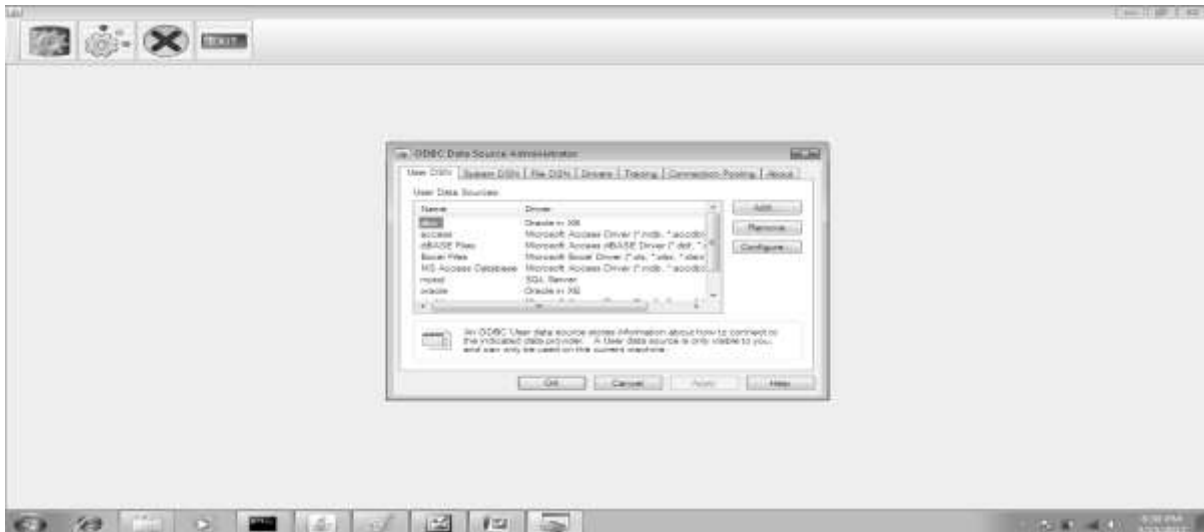
3. **Query Manager Window**

4. **Create oracle DSN**

Global Journal of Engineering Science and Research Management
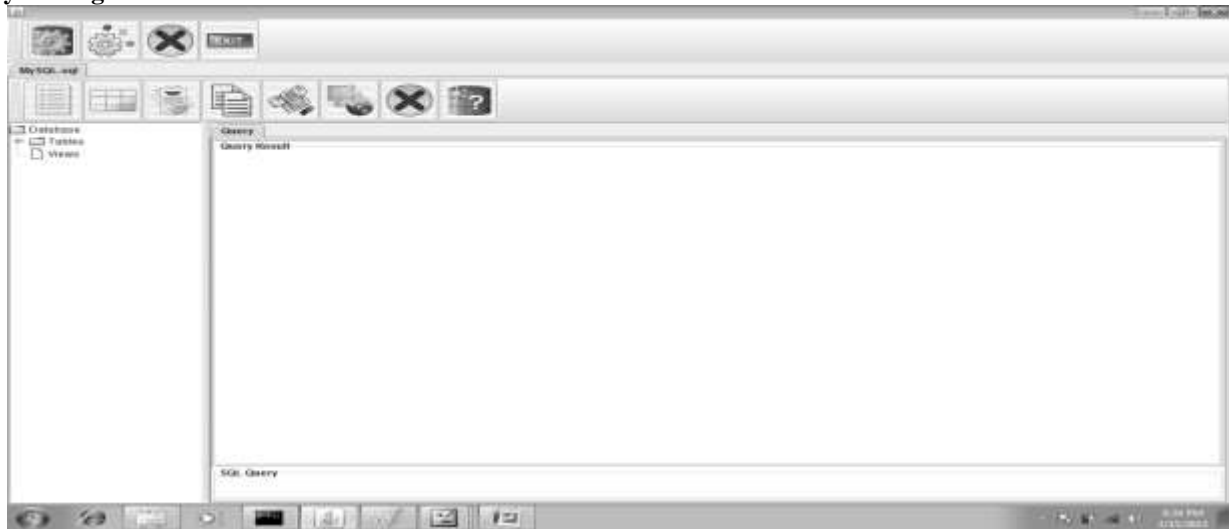
5. **Create table**
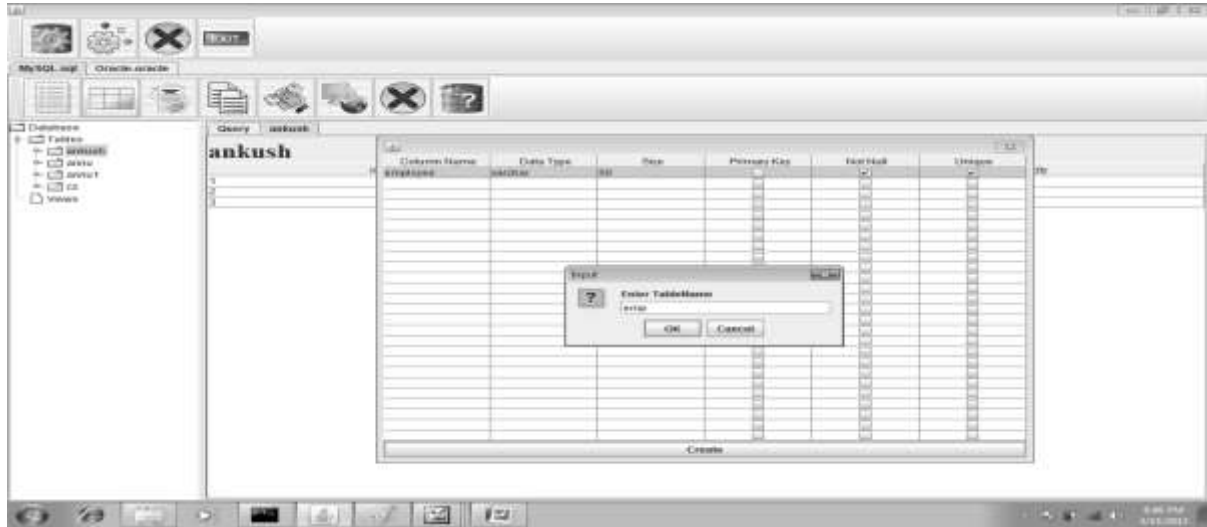
6. **Entering data from Query manager**

Global Journal of Engineering Science and Research Management
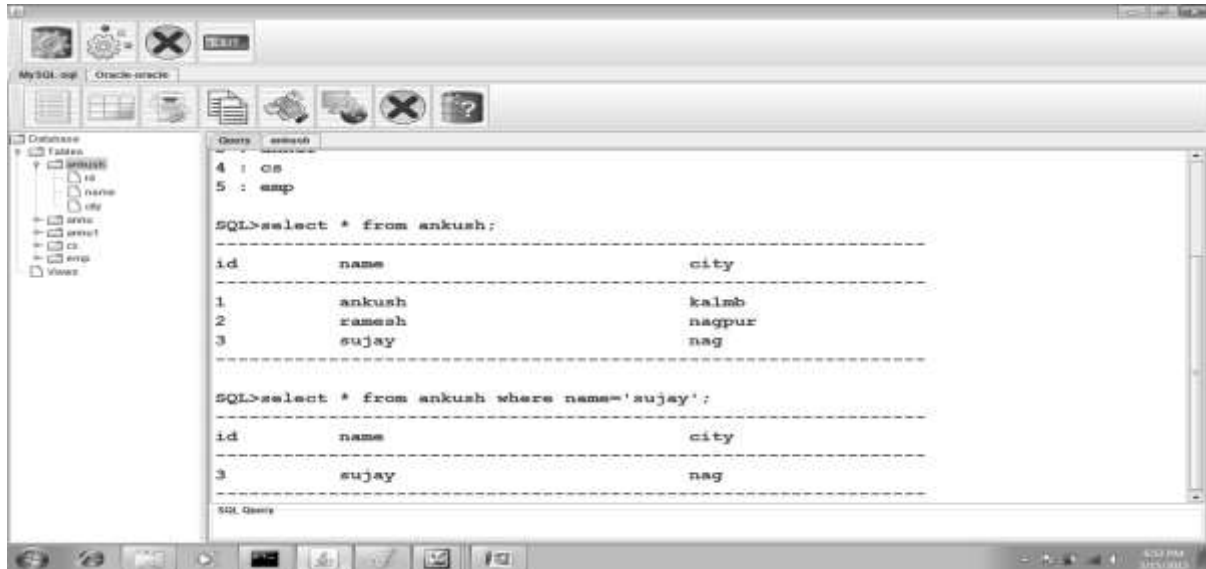
7.   Adding information Through ADD command window
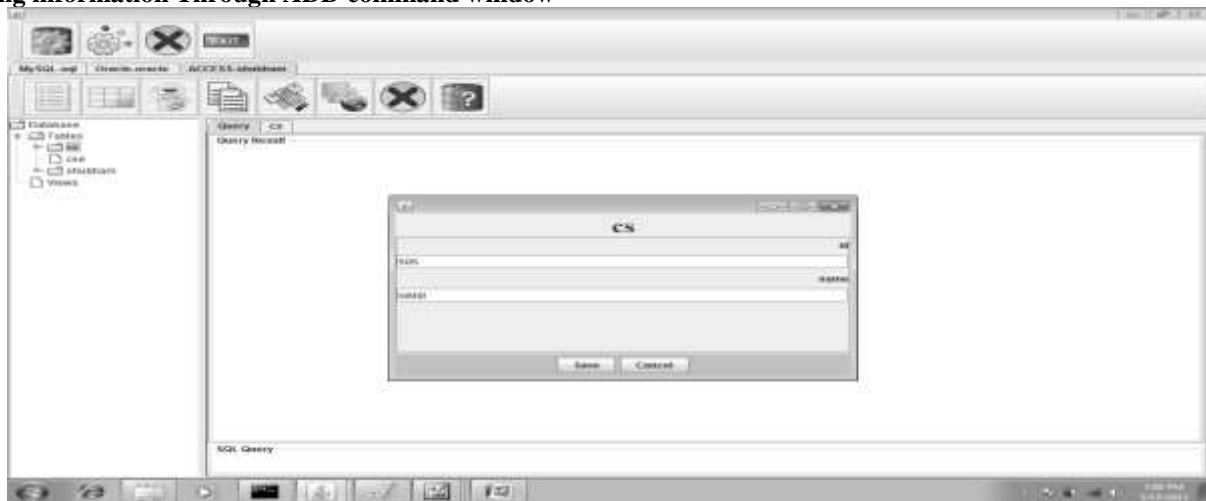
8.   Showing result for table

Global Journal of Engineering Science and Research Management

Conclusion and Future Scope

This Project will reduce the efforts of Programmer to create the databases with making a form using different databases. Project can supports any database like, Microsoft SQL Server, Oracle, DB2, My-SQL, and Microsoft Access. It can create database within one application window, so user does not have to use different tools for different databases. This tool will internally use JDBC to perform low level DBMS operations on database. It will use metadata classes of JDBC to retrieve information about database.

In this project module we create basic connectivity to the different databases using JDBC driver. This project is basically based on desktop application using Java technology. In our project there is no connection with network. So our future development plan is to modify this project to working with network layers such as using the concept of Cloud Computing.

References

1. *Complete Reference.*
2. *wordnet.princeton.edu/Perl/Waban*
3. *en.wikipedia.org/wiki/Wizard*
4. *en.wikipedia.org/wiki/Wizard_(software)*
5. *www.ontrack.com/glossary/*
6. *teladesign.com/ma-thesis/glossary.htm*
7. *faculty.washington.edu/~Larsson/conf/aiha98/primer/glossary.htm*
8. *www.spiritonline.com/dictionary/w.html*

Global Journal of Engineering Science and Research Management

9. *Java 2 fifth edition –Herbert Scheldt.*