



**Report on**  
**JAVA Fundamental Project**

**Identity Management System**  
**[IAM Project]**

**Submitted by: Sanjog BHATTARAI**

**Submitted to: Professor. Thomas BROUSSARD**

## Acknowledgements:

I extend my sincere thanks to **Professor. Thomas BROUSSARD** who continuously helped throughout the course, and without his guidance, this project would have been an uphill task.

I am also grateful to my friends in Specialization semester EPITA International master's program who helped with me regarding some of the issues that I encountered during the development of this project. Also, I would like to thank Mr. Rohit BASNET who continuously helped me to solve issues right from the start to end of the project.

# Contents

## [1. Subject Description](#)

### [1.1 Introduction](#)

#### [1.1.1 Overview:](#)

#### [1.1.2 Background:](#)

#### [1.1.3 Objective:](#)

#### [1.1.4 Development Environment](#)

## [2. Subject Analysis](#)

### [2.1 Major Features](#)

### [2.2 Application Feasibility](#)

### [2.3 Data Description](#)

### [2.4 Expected Results](#)

### [2.5 Algorithms Study](#)

### [2.6 Scope and limitations](#)

#### [Scope:](#)

#### [Limitations:](#)

#### [Evolution:](#)

## [3. Conception](#)

### [3.1 Chosen Algorithm](#)

### [3.2 Data Structures](#)

### [3.3 Global Application Flow](#)

### [3.4 Global schema and major features schema](#)

## [4. Console Operation Description](#)

### [4.1 Authenticate](#)

### [4.2 Create Operation](#)

### [4.3 Update Operation](#)

### [4.4 Delete Operation](#)

### [4.5 Listing Operation](#)

## [5. Configuration Instructions](#)

### [5.1 Setup The Database](#)

### [5.2 Run The Application](#)

### [5.3 Developers](#)

## [6. Commented Screenshots](#)

## [Bibliography](#)

## **1. Subject Description**

### **[1.1] Introduction**

#### **[1.1.1] Overview:**

This report contain all the detail from start to end of IAM project development.

#### **[1.1.2] Background:**

Keeping track of personal information disclosed on the Internet, as well as maintaining high standards of internal control and information security within companies has become increasingly complicated. Identity and Access Management applications lay at the core of resolving these types of problems.

Identity Management (IAM) can be defined as following:

A comprehensive set of processes that enable end users to securely access a broad range of internal and external IT systems control the digital representation of users and manage information about identities.

#### **[1.1.3] Objective**

The main goal of this project is to develop an application using Java which can automatically connect to a database and store information in the database and also update the database.

The following are the objectives that the application needs to satisfy:

1. Authenticates a user
2. Create an Identity
3. Update an Identity
4. Delete an Identity

#### **[1.1.4] Development Environment**

PLATFORM USED : DELL OS 10.1

LANGUAGE USED : Core Java

**IDE : Eclipse**

DATABASE : Derby

## **2. Subject analysis**

### **[2.1] Major Features**

- Highly user-friendly
- Platform Independent
- Easy to use
- Robust
- Data entry restricted to avoid errors
- Clean separation of various components
- Easy Modification

### **[2.2] Application Feasibility**

#### 1) Strengths

- Database Driven application and faster to access than manual file system record.

#### 2) Weakness

- Not much user information attributes included for now. - User information storing may be time consuming at first.

#### 3) Threats

- Some Information may be confidential and can be stolen.
- System may hang if run on low hardware configuration system.

#### 4) In Future

- Many user attributes can be added to store more user information.
- Can be implemented with Graphical User Interface (GUI) .
- Search Feature can be implemented.
- multiple admin users can be added and implement in **database**.

### **[2.3] Data Description**

The user will enter:

- Display name
- Email
- Actions to perform

## **[2.4] Expected Results**

- Administrators of system will be able to view ,add, modify and update the user information easily.

## **[2.5] Algorithm Study**

For this system a simple and basic algorithm is used for the insertion, manipulation, deletion and listing of the user's information.

## **[2.6] Scope and limitations**

The present project does not really allow Identity to connect to the database. The next step will be to grant privileges to each Identity and check the right to display the menu. Right now, only the root the schema can connect to the database.

Other step will be to include a Graphical interface to the application.

Managing sql jointures are not very easy especially between attributes and the other classes, another step will be to use a tool such as Hibernate or Spring

The search feature also is not well advanced and search methods need to be implemented The Identity password is not yet secured.

## **3.Conception**

The application contains the following packages:

- Launcher package
- Datamodel package
- Services package

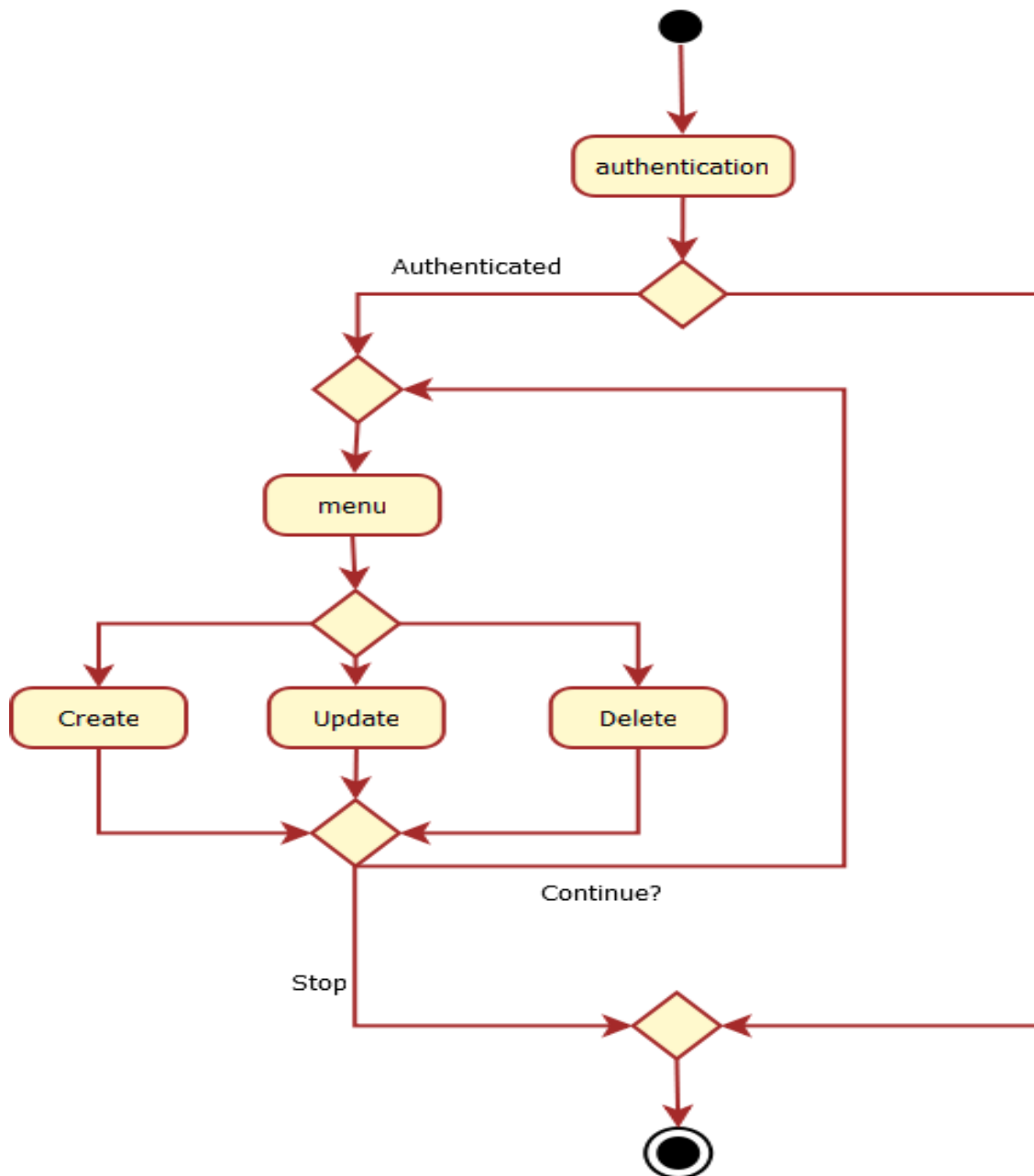
### **[3.1] Chosen Algorithm**

The user with administrator's right have to login first with credentials (username = adm and password = pwd). If authentication is approved the menu for insert, update, delete and list users will appear. Then they will be able to perform the actions specified. The application is halted if admin choses quit option.

### **[3.2] Data Structures**

- class
- lists

### **[3.3] Global Application Flow**



### [3.4] Global Schema and Major Features Schema

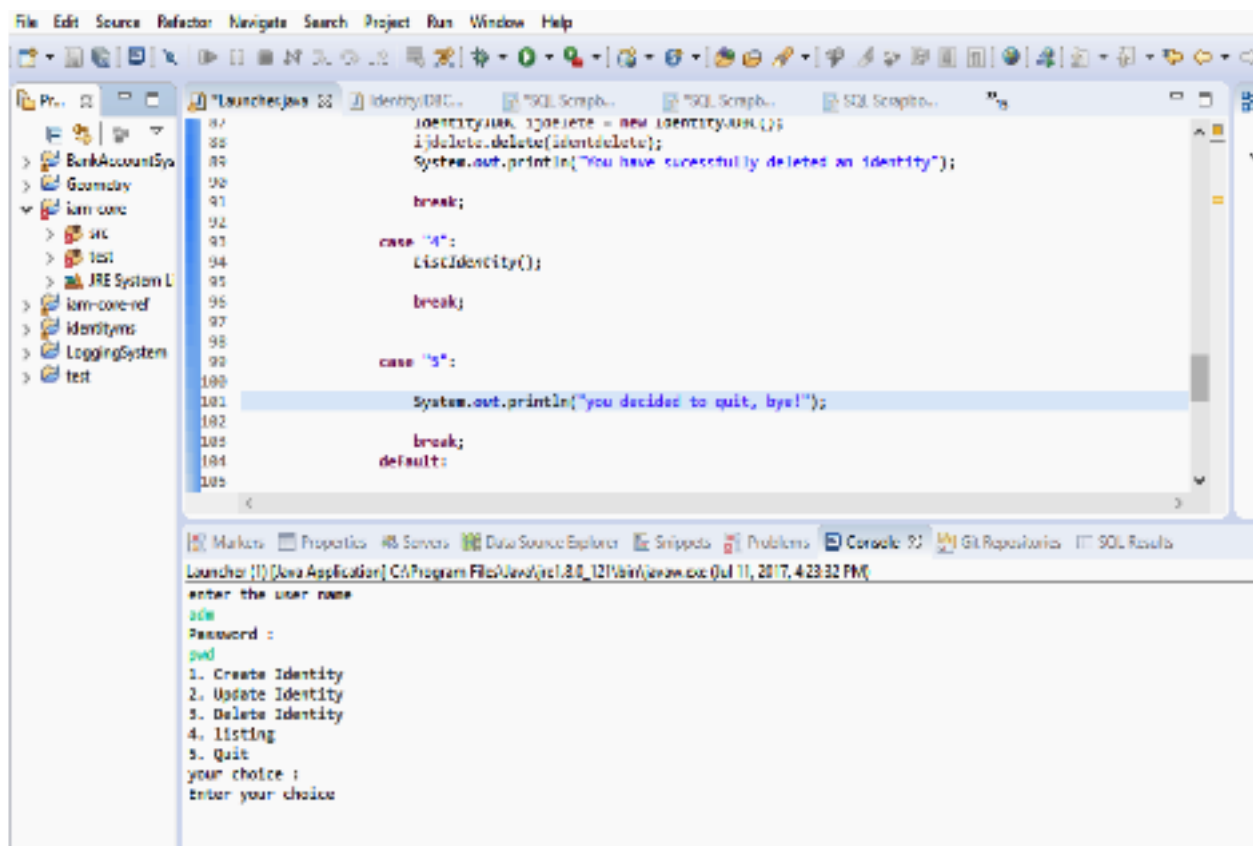
<b>Indentity</b>
<b>uname</b>
<b>uemail</b>
<b>methods</b>



## 4. Console Operations Description

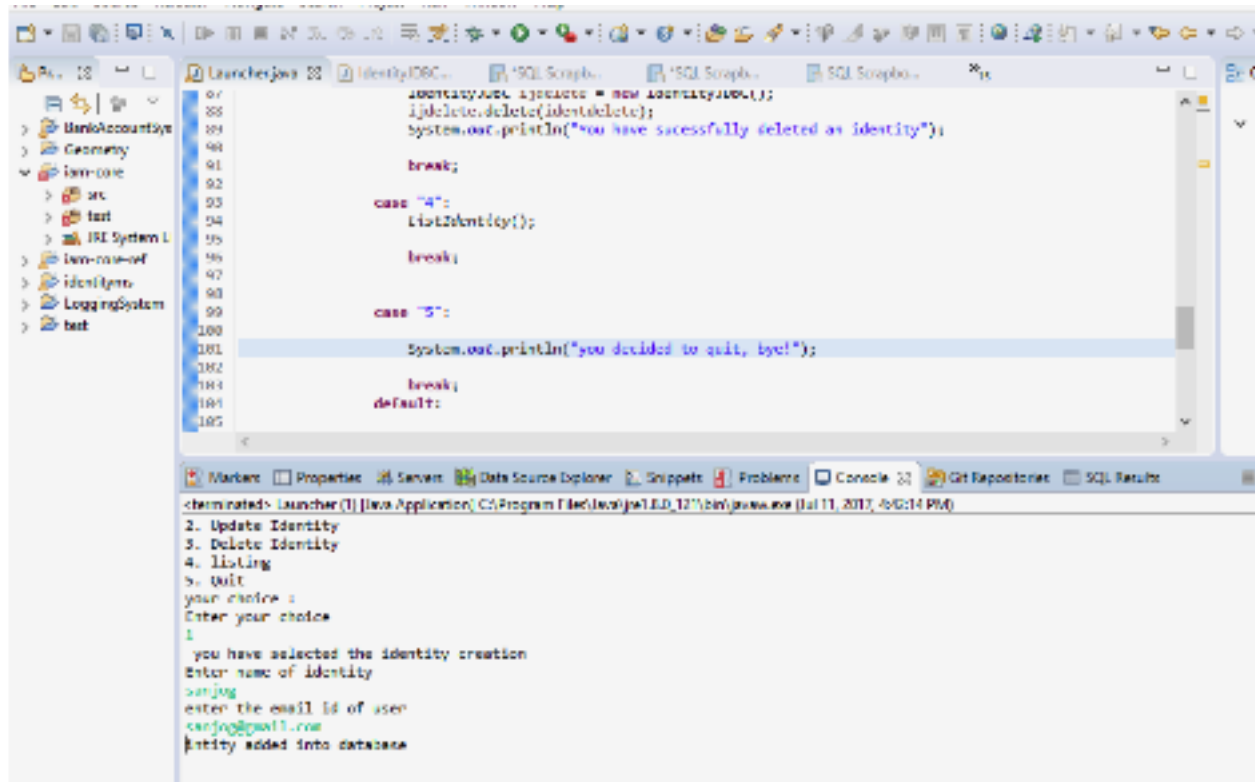
### [4.1] Authentication

- The user launches the application
- The application requests the login
- The user enters a user name
- The application requests the password
- The user enters the password



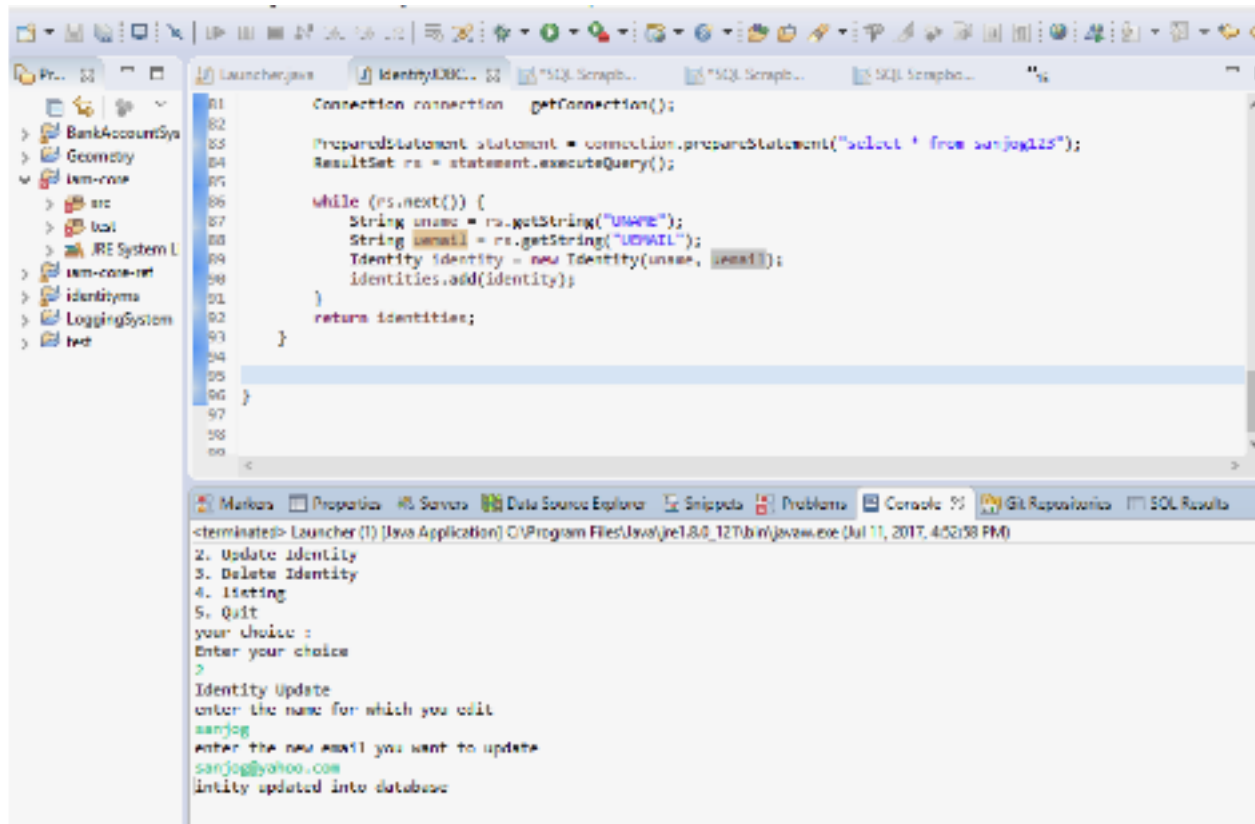
### [4.2] Create Operation

- The application prompt the user to enter the choice
- The user types "1"
- The application prompts the user that "you have selected the identity creation"
- Enter name of Identity and Enter the email id of user
- The application creates an Identity



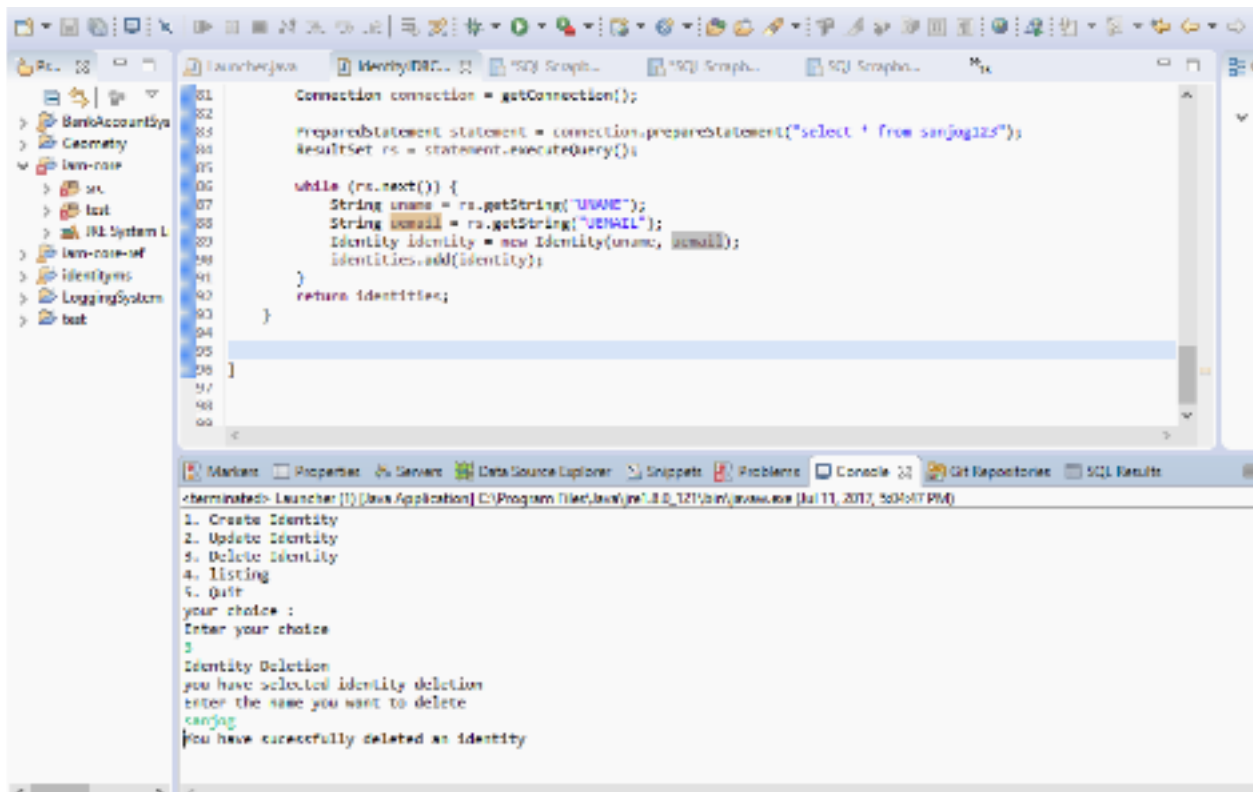
### [4.3] Update Operation

- The application prompt the user to select an action .
- The user types “2”
- The application presents to Identity update
- The application prompts the user to enter a value for that attribute
- The user enters a value for the selected attributes
- The application update the identity
- The application prompt the user to continue or the leave the application



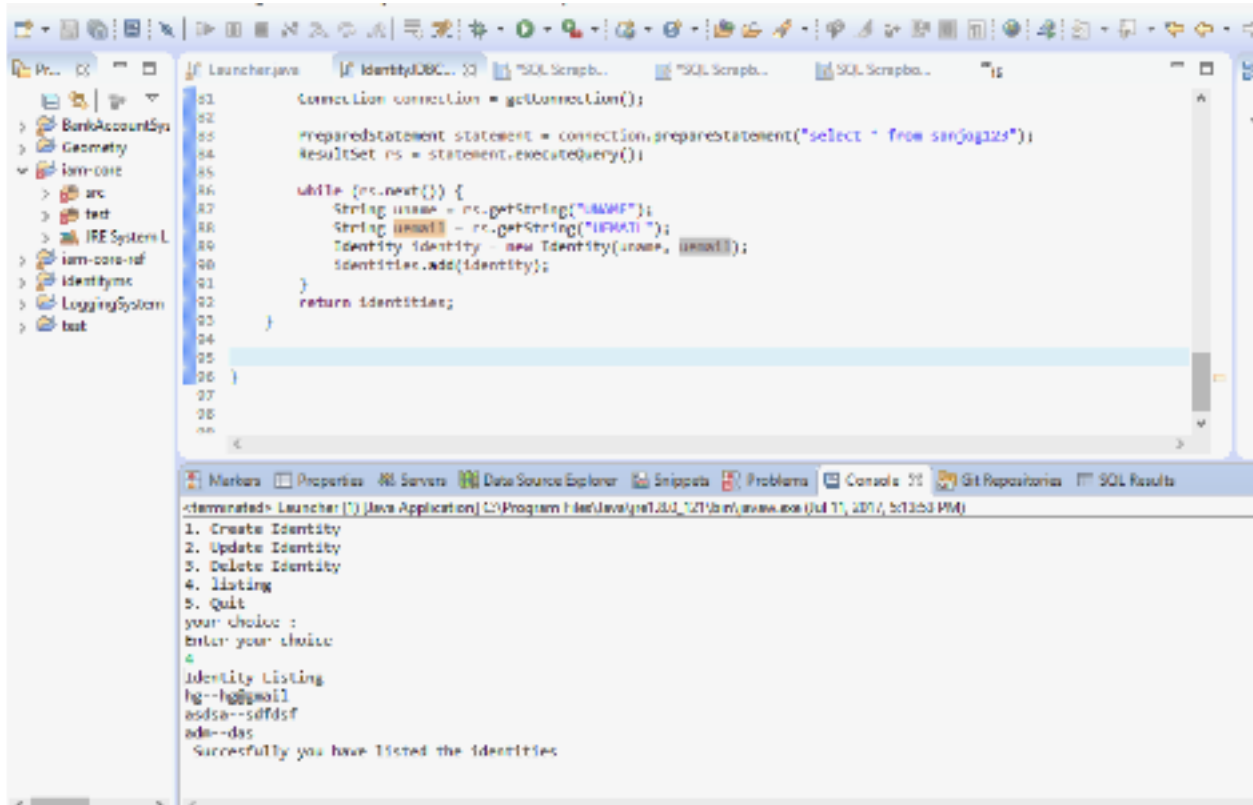
#### [4.4] Delete Operation

- The application prompt the user to select an action.
- The user types “3”
- The application prompts the user that “Identity Deletion”
- Enter the name you want to delete
- The application prompt the user to continue or the leave the application



## [4.5] Listing Operation

- The application prompt the user to select an action.
- The user types “4”
- The application prompts the user that “Identity Listing”
- The application prompt the user to continue or the leave the application



## 5. Configuration Instructions

### [5.1] Setup The Database

- Download apache derby client
- Create an instance of derby name it instance name =mynew with username="sanjog"and password="sanjog".
- run /db-derby-10.13.1.1-bin/bin/startNetworkServer
- Go to iam-sanjog/src/fr/epita/iam/config/config.properties
- edit user value with the user name you used to create your derby schema
- Do the same with password, the name if instance if you used your own values
- Go to iam-sanjogcore/sql run create\_tables.sql on your instance mynew

### [5.2] Run The Application

- Go to iam-sanjogcore/src/fr/epita/iam/launcher execute ConsoleLauncher.java

### [5.3] Developers

- Go to iam\_sanjogcore/javadoc for documentation

## **6. Commented Screenshots**

- No screenshot is available .We have not yet implement the Graphical User Interface.

## **Bibliography**

<http://thomas-broussard.fr/>

<http://www.tutorialspoint.com/>

[www.stackoverflow.com](http://www.stackoverflow.com)