

**A PBL Report**  
**of**  
**“B.tech computer science”**  
**in**  
**“SECURE, a desktop application”**  
**Submitted to**



**SAGE UNIVERSITY**  
**BHOPAL (M.P.)**

**Submitted by**

**ARYAN SAHU**

**AAYUSH KHARE**

**PIYUSH SHASTRWE**

**20BTE3CSE10008**

**20BTE3CSE10001**

**20BTE3CSE10023**

**Under the supervision of**

**Prof. Nikhil chourasiya**

**SCHOOL OF ENGINEERING AND TECHNOLOGY**  
**SAGE UNIVERSITY, BHOPAL**  
**(2020-21)**



**SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, Bhopal  
SCHOOL OF ENGINEERING AND TECHNOLOGY**

**DECLARATION**

**WEARYAN SAHU, AAYUSH KHARE, PIYUSH SHASTRWE** student of “**Bachelor Of Technology in COMPUTER SCIENCE**”, session **2021-22**, hereby informed that the work presented in this project entitled “**SECURE**” is the outcome of my own work, is a bonafide and correct to the best of my knowledge and work has been carried out taking care of ethics. The work presented does not infringe any patented work and has not been submitted to any other University or anywhere else.

**Date:**  
**Place: Bhopal**

**Student Name**  
**ARYAN SAHU**  
**AYUSH KHARE**  
**PIYUSH SHASTRWE**

**Supervisor**  
**Prof. Nikhil chourasiya (Guide)**  
**( Bhopal)**

**Dean/HOS**



**SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, Bhopal**

**SCHOOL OF ENGINEERING AND TECHNOLOGY**

**CERTIFICATE**

This is to certify that the work embodies in this project entitled “**SECURE**” being submitted by **ARYAN SAHU, AAYUSH KHARE, PIYUSH SHASTRI**, [Enrollment No.**20BTE3CSE10008, 20BTE3CSE10001, 20BTE3CSE10023**] in partial fulfillment of the requirement for the award of “**Bachelor of Technology in Computer science**” to Sage University, Bhopal (M.P) during the academic year **2021-22** is a record of bonafide piece of work, carried out by her/his under my supervision and guidance in the **School of Engg. And Technology, Bhopal.**

**Supervisor**  
**Prof. Nikhil Chourasiya (Guide)**  
**SSOET, Bhopal**

**HOS**  
**SSOET, Bhopal**



**SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, BHOPAL  
SCHOOL OF ENGINEERING AND TECHNOLOGY**

**DECLARATION OF PLAGIARISM**

WE hereby declare that the work embodies in this project entitled “**SECURE**” being submitted by, **ARYAN SAHU, AAYUSH KHARE, PIYUSH SHASTRWE**[Enrollment No. **20BTE3CSE10008, 20BTE3CSE10001, 20BTE3CSE10023**] of “**Bachelor Of Technology in Computer science**” in **SCHOOL OF ENGG. AND TECHNOLOGY, Bhopal**, is an authentic record of my own work carried under the supervision and guidance of **Prof. Nikhil Chourasiya** and has not been submitted anywhere.

WE also declare that “A check for Plagiarism has been carried out on this project report and is found within the accepted limit and report of which is enclosed as annexure I”.

**Students Name**  
**ARYAN SAHU,**  
**AAYUSH KHARE**  
**PIYUSH SHASTRWE**

**Supervisor**  
**Prof. Nikhil Chourasiya**

**HOS/Dean Signature**



**SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, Bhopal**

**SCHOOL OF ENGINEERING AND TECHNOLOGY**

**CERTIFICATE OF APPROVAL**

The project entitled “**SECURE**” being submitted by **ARYAN SAHU, AAYUSH KHARE, PIYUSH SHASTRWE**, [Enrollment No. **20BTE3CSE10008, 20BTE3CSE10001, 20BTE3CSE10023**] has been examined by us and is hereby approved for the partial fulfillment for the award of “**Bachelor of Technology in COMPUTER SCIENCE**” for which it has been submitted. It is understood that by this approval the undersigned do not necessary endorse or approve any statement made, opinion expressed or conclusion drawn therein but approved the project only for the purpose for which it has been submitted.

**(Internal Examiner)**  
**Prof. (Guide)**  
**(SSOET, SAGE, Bhopal)**

**(External Examiner)**



**SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, Bhopal  
SCHOOL OF ENGINEERING AND TECHNOLOGY**

**ACKNOWLEDGEMENT**

It is my proud privilege to present a project on “**SECURE**”. WE take this opportunity to express deep sense of gratitude and would like to give thanks to my guide, **Prof. Nikhil Chourasiya** **SCHOOL OF Engg. And TECHNOLOGY**, Bhopal, for their valuable guidance, inspiration and encouragement that has led to successful completion of this work. WE would like to express my heartfelt thanks towards HOS. Sanjeev Shrivastava. WE would like to express my heartfelt thanks to Nivedita Singh , for their valuable suggestions throughout the project work. WE could not have accomplished, what WE actually have, without their guidance.

WE would like to express my heartfelt thanks and sense of gratitude to Prof. Abhigyan Tiwari for being a constant source of inspiration. WE are also thankful to all faculty members and staff of SSOET for their suggestion and support.

WE would like to deeply thank my family and friends for all the support and encouragement they have rendered time to time.

Last but not the least, WE dedicate my work to almighty God without whose wish and helping hands this work would not have taken the shape it has now and also to my family members whose support and encouragement had led me to complete this task.

## TABLE OF CONTENTS

S. No	Title	Page No
1.	INTRODUCTION	1
2.	OBJECTIVE & MOTIVATION	2
3.	TECHNOLOGIES USED	3
4.	PROPOSED WORK	6
6.	IMPLEMENTATION & RESULT ANALYSIS	29
7.	CONCLUSION AND FUTURE WORK	35
8.	REFERENCE	36
9.	PLAGIARISM REPORT	37

## LIST OF FIGURES

S. No.	Title	Page No.
1	Login Page	29
2	Login Failed	29
3	Login details	30
4	Login successful	30
5	Signup page with details	31
6	Signup: must contain all necessary characters	31
7	Signup: Passwords do not match	32
8	Logout Window	32
9	Database Part 1	33
10	Database Part 2	33
11	Flowchart	34

## LIST OF TABLES

S.No.	Title	Page No.
1	Table of classes Created and their description	6

# INTRODUCTION

**“SECURE a desktop application”**. A desktop application is a software program which runs stand alone to perform a specific task by the end user.

**SECURE** is a desktop application which can be used for various tasks and can be installed on various devices, it is specially designed for maintaining the record of the users.

Secure is a application which Is launched as soon as the desktop booted up and the user can only use the system after logging in it , after logging in it the details of user is stored in its data server .

It will maintain the record of the user for:-

- For how much time the user has used the pc
- Attendance of the user
- Whether user has shut down the system properly or not
- For security purposes



## OBJECTIVE & MOTIVATION

It's a time consuming process to take attendance of multiple students in the labs also it is next to impossible to detect the mischievous activated done by the students to the system, so our objective was to create a system which automatically update their attendance and can keep a record of the "mischievous" activities done by them these activities could range anywhere between unplugging the system's hardware or as simple as not shutting the pc down properly.

We aim to create an application software which could be Implemented and deployed in our own Sage University Bhopal. The aim is to create this software in a way that a pc based server could communicate with all the other computer in the university connected to each other over the intranet. We for a software which could keep a record of all these activities and can be handy for the teachers to maintain the proper decorum of the class.

We realized that to create this project we would need to use a variety of technologies, the technologies which we decied to use in this project are mentioned in next few pages.

# TECHNOLOGIES USED

- **JAVA:**

Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let programmers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need to recompile.

Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages. As of 2019, Java was one of the most popular programming languages in use according to GitHub, particularly for client–server web applications, with a reported 9 million developers.

Java was originally developed by James Gosling at Sun Microsystems and released in May 1995 as a core component of Sun Microsystems' Java platform. The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licenses. As of May 2007, in compliance with the specifications of the Java Community Process, Sun had relicensed most of its Java technologies under the GPL-2.0-only license. Oracle offers its own HotSpot Java Virtual Machine, however the official reference implementation is the OpenJDK JVM which is free open-source software and used by most developers and is the default JVM for almost all Linux distributions.

As of October 2021, Java 17 is the latest version. Java 8, 11 and 17 are the current long-term support (LTS) versions. Oracle released the last zero-cost public update for the legacy version Java 8 LTS in January 2019 for commercial use, although it will otherwise still support Java 8 with public updates for personal use indefinitely. Other vendors have begun to offer zero-cost builds of OpenJDK 8 and 11 that are still receiving security and other upgrades.

- Apache Derby:

Apache Derby (previously distributed as IBM Cloudscape) is a relational database management system (RDBMS) developed by the Apache Software Foundation that can be embedded in Java programs and used for online transaction processing. It has a 3.5 MB disk-space footprint.<sup>[1]</sup>

Apache Derby is developed as an open source project under the Apache 2.0 license. For a time, Oracle distributed the same binaries under the name Java DB. In June 2015 they announced that for JDK 9 they would no longer be doing so.

Apache Derby, an [Apache DB subproject](#), is an open source relational database implemented entirely in Java and available under the [Apache License, Version 2.0](#). Some key advantages include:

- Derby has a small footprint -- about 3.5 megabytes for the base engine and embedded JDBC driver.
- Derby is based on the Java, JDBC, and [SQL](#) standards.
- Derby provides an embedded JDBC driver that lets you [embed](#) Derby in any Java-based solution.
- Derby also supports the more familiar client/server mode with the [Derby Network Client JDBC driver and Derby Network Server](#).
- Derby is easy to install, deploy, and use.

# PROPOSED WORK

We created five different classes for three different jFrames, one class for managing the data in the apache derby database and one class as a data type.

Class Name	Discription	Lines of code
LogInForm	<p>This is a JFrame form which will pop up as soon as the windows is booted successfully, this page cannot be closed.</p> <p>It has two fields, namely username and password, it has a two buttons, namely Sign Up and Login.</p> <p>If the username and password matches with our database, the user will be logged in and can use the computer now.</p> <p>If the signup is pressed the user will be re-directed to our signup page.</p> <p>This page communicates with the database using the DataManager class.</p>	183
SignUpForm	<p>This is a JFrame form which appears via a button in the LogInForm class. This page has all the required fields which are necessary for our database.</p> <p>It has a button called the signup button which when pressed checks all the details and then if everything is good to go the user is registered in the database.</p> <p>Once this happens a button pops-up which redirects the user to the login Page.</p>	321
LogOutForm	<p>Once a user is logged in successfully this page appears, this page is minimizable but cannot be closed.</p> <p>A user can use the computer only when this form is running.</p> <p>This page only has one button called the logout button.</p> <p>Once this button is pressed the attendance is marked and if and mischievous activity was done to the</p>	73

	computer this page automatically updates the database using the DataManager class. Once the logout button is pressed the system shutdown automatically.	
DataManager	This class deals directly with the database. All the other classes uses methods of this class to update, retrieve, insert and delete data from database. This class is the core of our Software without which the software is just graphical user interface. The methods in this class uses the user defined data Type StudentInfo which is specifically created for this Software.	87
StudentInfo	This class is a logical userdefined data type, all the classes use this datatype to make the code more readable and it makes Intra-class communication easy since the object of this class once passed contains all the necessary info about the student using the system.	32

- Login Forn:

```
import java.time.LocalTime;
import javax.swing.JFrame;
import javax.swing.JOptionPane;
import java.sql.*;
```

```
public class LogInForm extends javax.swing.JFrame {
```

```
    public LogInForm() {
        initComponents();
        setLocationRelativeTo(null);
        setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);
    }
```

```
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {
```

```
jLabel1 = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();
jLabel3 = new javax.swing.JLabel();
tEmail = new javax.swing.JTextField();
tPassword = new javax.swing.JPasswordField();
jButton1 = new javax.swing.JButton();
jButton2 = new javax.swing.JButton();
jLabel4 = new javax.swing.JLabel();
jLabel5 = new javax.swing.JLabel();
```

```
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
jLabel1.setFont(new java.awt.Font("Times New Roman", 0, 36)); // NOI18N
jLabel1.setText("Secure");
```

```
jLabel2.setText("EnrollmentNumber");
```

```
jLabel3.setText("Password");
```

```
jButton1.setText("LogIn!");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
```

```
jButton2.setText("Register");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});
```

```
jLabel4.setText("A Desktop Application");
```

```

jLabel5.setFont(new java.awt.Font("Tahoma", 0, 10)); // NOI18N
jLabel5.setForeground(new java.awt.Color(255, 0, 51));
jLabel5.setText("*note: The above field is case sensitive, use uppercase
letters");

    javax.swing.GroupLayout layout = new
    javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
    layout.createSequentialGroup()
        .addGap(183, Short.MAX_VALUE)
        .addComponent(jLabel1)
        .addGap(239, 239, 239))
        .addGroup(layout.createSequentialGroup()

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(47, 47, 47)

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
        .addComponent(jButton2,
    javax.swing.GroupLayout.DEFAULT_SIZE,
    javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jButton1,
    javax.swing.GroupLayout.DEFAULT_SIZE,
    javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addGroup(layout.createSequentialGroup()

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel2)
        .addComponent(jLabel3))
        .addGap(18, 18, 18)

```



```

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(tEmail)
            .addComponent(tPassword)
            .addComponent(jLabel5,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))))
        .addGroup(layout.createSequentialGroup()
            .addGap(155, 155, 155)
            .addComponent(jLabel4)))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jLabel4)
            .addGap(51, 51, 51)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel2)
            .addComponent(tEmail,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(8, 8, 8)
            .addComponent(jLabel5)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel3)
            .addComponent(tPassword,
javax.swing.GroupLayout.PREFERRED_SIZE,

```

```

javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(28, 28, 28)
    .addComponent(jButton1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
    .addComponent(jButton2)
    .addContainerGap(47, Short.MAX_VALUE))
);

```

```

    pack();
} // </editor-fold>

```

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    SignUpForm obj=new SignUpForm();
    obj.setVisible(true);
    this.setVisible(false);
}

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    String t=LocalTime.now()+"";
    String email=tEmail.getText();
    String enroll=email;
    String password=tPassword.getText();
    try{
        if(DataManager.isPresent(enroll, email, password)){
            JOptionPane.showMessageDialog(null, "Logged In!");
        }else{
            JOptionPane.showMessageDialog(null,"Wrong details Entered");
            return;
        }
    }catch(SQLException e){JOptionPane.showMessageDialog(null,"Error: 404,
TRY AGAIN");}
    DataManager.addLogInTime(email, t);
    new LogOutForm(email,LocalTime.now()).setVisible(true);
    this.setVisible(false);
    this.dispose();

}

```

```

public static void main(String args[]) {

    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default
look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(LogInForm.class.getName()).log(java.util.loggi
ng.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(LogInForm.class.getName()).log(java.util.loggi
ng.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(LogInForm.class.getName()).log(java.util.loggi
ng.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(LogInForm.class.getName()).log(java.util.loggi
ng.Level.SEVERE, null, ex);
    }
    //</editor-fold>

    java.awt.EventQueue.invokeLater(new Runnable() {

```

```

        public void run() {
            new LogInForm().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JTextField tEmail;
private javax.swing.JPasswordField tPassword;
// End of variables declaration
}

```

- **SignUpForm:**

```

import javax.swing.JOptionPane;
import java.swing.JFrame;

```

```

public class SignUpForm extends javax.swing.JFrame {

```

```

    public SignUpForm() {
        initComponents();
        setLocationRelativeTo(null);
        setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);
        back.setVisible(false);
    }
    static boolean checker(String s){
        boolean caps=false;
        boolean small=false;
        boolean cha=false;
        boolean number=false;
        for(char i = 'a' ; i<= 'z' ;i++ ){

```

```

        for(int j=0;j<s.length();j++){
            if(s.charAt(j)==i){
                small=true;
                break;
            }
        }
        if(small==true){
            break;
        }
    }
    for(char i = 'A' ; i<= 'Z' ;i++){
        for(int j=0;j<s.length();j++){
            if(s.charAt(j)==i){
                caps=true;
                break;
            }
        }
        if(caps==true){
            break;
        }
    }
    for(char i = 33 ; i<= 47 ;i++){
        for(int j=0;j<s.length();j++){
            if(s.charAt(j)==i){
                cha=true;
                break;
            }
        }
        if(cha==true){
            break;
        }
    }
    if(!cha){
        for(char i = 58 ; i<= 64 ;i++){
            for(int j=0;j<s.length();j++){
                if(s.charAt(j)==i){
                    cha=true;
                    break;
                }
            }
            if(cha==true){
                break;
            }
        }
    }

```

```

        }
    }
}
if(!cha){
    for(char i = 92 ; i<= 96 ;i++ ){
        for(int j=0;j<s.length();j++){
            if(s.charAt(j)==i){
                cha=true;
                break;
            }
        }
        if(cha==true){
            break;
        }
    }
}
if(!cha){
    for(char i = 123 ; i<= 126 ;i++ ){
        for(int j=0;j<s.length();j++){
            if(s.charAt(j)==i){
                cha=true;
                break;
            }
        }
        if(cha==true){
            break;
        }
    }
}

for(char i = '0';i<='9';i++){
    for(int j=0;j<s.length();j++){
        if(s.charAt(j)==i){
            number=true;
            break;
        }
    }
    if(number==true){
        break;
    }
}

if(caps && small && cha && number) return true;

```

```

        else return false;
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jLabel5 = new javax.swing.JLabel();
        jLabel6 = new javax.swing.JLabel();
        jLabel7 = new javax.swing.JLabel();
        jLabel8 = new javax.swing.JLabel();
        jLabel9 = new javax.swing.JLabel();
        tEnrollmentNumber = new javax.swing.JTextField();
        tName = new javax.swing.JTextField();
        tEmail = new javax.swing.JTextField();
        tPassword = new javax.swing.JPasswordField();
        tConfirmPassword = new javax.swing.JPasswordField();
        tSchool = new javax.swing.JTextField();
        tBranch = new javax.swing.JTextField();
        tSpecial = new javax.swing.JTextField();
        Register = new javax.swing.JButton();
        back = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jLabel1.setText("Enter the following details to register");

        jLabel2.setText("Enrollment Number");

        jLabel3.setText("Name");

        jLabel4.setText("Email");

```

```
jLabel5.setText("Password");

jLabel6.setText("Confirm Password");

jLabel7.setText("School");

jLabel8.setText("Branch");

jLabel9.setText("Special");

tEnrollmentNumber.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tEnrollmentNumberActionPerformed(evt);
    }
});

Register.setText("Register");
Register.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        RegisterActionPerformed(evt);
    }
});

back.setText("Go To Login Page");
back.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        backActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
```



```

getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addGap(126, 126, 126)
        .addComponent(jLabel1))
    .addGroup(layout.createSequentialGroup()
        .addGap(59, 59, 59)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
    .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jLabel2)
    .addComponent(jLabel3)
    .addComponent(jLabel4)
    .addComponent(jLabel5)
    .addComponent(jLabel6)
    .addComponent(jLabel7)
    .addComponent(jLabel8)
    .addComponent(jLabel9))
        .addGap(79, 79, 79)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
    .addComponent(tEnrollmentNumber)
    .addComponent(tName)
    .addComponent(tEmail)
    .addComponent(tPassword)
    .addComponent(tConfirmPassword)
    .addComponent(tSchool)
    .addComponent(tBranch)
    .addComponent(tSpecial,
javax.swing.GroupLayout.DEFAULT_SIZE, 189, Short.MAX_VALUE)))
        .addComponent(Register, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(back, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))))

```

```

        .addContainerGap(55, Short.MAX_VALUE))
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup())
        .addGap(52, 52, 52)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING
        )
            .addGroup(layout.createSequentialGroup())

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING
        )
            .addGroup(layout.createSequentialGroup())

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING
        )
            .addGroup(layout.createSequentialGroup())

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING
        )
            .addGroup(layout.createSequentialGroup())

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING
        )
            .addGroup(layout.createSequentialGroup())

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING
        )
            .addGroup(layout.createSequentialGroup()
                .addComponent(jLabel1)
                .addGap(41, 41, 41)
                .addComponent(jLabel2))
                .addComponent(tEnrollmentNumber,
                    javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE,
                    javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(18, 18, 18)
                .addComponent(jLabel3))
    );

```

```

        .addComponent(tName,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addComponent(jLabel4))
        .addComponent(tEmail,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addComponent(jLabel5))
        .addComponent(tPassword,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addComponent(jLabel6))
        .addComponent(tConfirmPassword,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addComponent(jLabel7))
        .addComponent(tSchool,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addComponent(jLabel8))
        .addComponent(tBranch, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING
)
        .addComponent(jLabel9)
        .addComponent(tSpecial, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(31, 31, 31)
        .addComponent(Register)

```

```

        .addGap(18, 18, 18)
        .addComponent(back)
        .addContainerGap(22, Short.MAX_VALUE))
    );

    pack();
} // </editor-fold>
private StudentInfo dataInsertor(){
    StudentInfo si=new StudentInfo();
    si.enrollmentNumber=tEnrollmentNumber.getText();
    si.name=tName.getText();
    si.email=tEmail.getText();
    si.password=tPassword.getText();
    si.school=tSchool.getText();
    si.branch=tBranch.getText();
    si.special=tSpecial.getText();
    si.lastLogIn="";
    si.lastLogOut="";
    si.lastLogOutTime="";
    si.attendence="";
    //    System.out.println("data read");
    return si;
}
private void tEnrollmentNumberActionPerformed(java.awt.event.ActionEvent evt) {

}

private void RegisterActionPerformed(java.awt.event.ActionEvent evt) {
    if(!checker(tPassword.getText())){
        JOptionPane.showMessageDialog(null,"The password must contain a capital
letter, a small letter, a number and a special character");
    }else if(!(tPassword.getText().equals(tConfirmPassword.getText()))){
        JOptionPane.showMessageDialog(null,"The passwords do not match");
    }else{
        StudentInfo si=new StudentInfo();
        si=dataInsertor();
        DataManager.insertor(si);
        back.setVisible(true);
    }
}
}

```

```

private void backActionPerformed(java.awt.event.ActionEvent evt) {
    LogInForm obj=new LogInForm();
    obj.setVisible(true);
    this.setVisible(false);
    this.dispose();
}

```

```

private javax.swing.JButton Register;
private javax.swing.JButton back;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JTextField tBranch;
private javax.swing.JPasswordField tConfirmPassword;
private javax.swing.JTextField tEmail;
private javax.swing.JTextField tEnrollmentNumber;
private javax.swing.JTextField tName;
private javax.swing.JPasswordField tPassword;
private javax.swing.JTextField tSchool;
private javax.swing.JTextField tSpecial;

}

```

- **LogOutForm:**

```

import javax.swing.JFrame;
import javax.swing.JOptionPane;

public class LogOutForm extends javax.swing.JFrame {
    String enroll;
    LocalTime lit;
    long start=System.currentTimeMillis();

    public LogOutForm(String s, LocalTime t) {
        initComponents();
        setLocationRelativeTo(null);
        setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);
        enroll=s;
        lit=t;
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jButton1 = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jButton1.setText("LOGOUT");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout layout = new
        javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

```

```

        .addGroup(layout.createSequentialGroup()
            .addGap(34, 34, 34)
            .addComponent(jButton1, javax.swing.GroupLayout.PREFERRED_SIZE,
334, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(32, Short.MAX_VALUE))
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(124, 124, 124)
            .addComponent(jButton1)
            .addContainerGap(146, Short.MAX_VALUE))
    );

    pack();
} // </editor-fold>

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    LocalDateTime lot=LocalTime.now();
    Long end=System.currentTimeMillis();
    Long timeElapse=end-start;
    DataManager.lastLogOut(enroll, lot.toString());
    DataManager.lastLogInTime(enroll, timeElapse.toString());
    if(timeElapse>1800000){
        int attendance=DataManager.getAttendance(enroll);
        attendance++;
        DataManager.attendance(enroll,attendance+"");
    }
    JOptionPane.showMessageDialog(null,"LogOut Successfull!");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

private javax.swing.JButton jButton1;

}

```

- **DataManager:**

```

import java.sql.*;
import javax.swing.JOptionPane;

```

```

public class DataManager {
    private static Connection connector()throws SQLException{
        Connection
con=DriverManager.getConnection("jdbc:derby://localhost:1527/Secure","aryansahu400
","Aryansahu@400");
        return con;
    }
    public static void inserter(StudentInfo st){
        try{
            Connection con=DataManager.connector();
            PreparedStatement pa = con.prepareStatement("insert into details
values(?,?,?,?,?,?,?,?,?,?,?)");
            pa.setString(1,st.enrollmentNumber);
            pa.setString(2,st.name);
            pa.setString(3,st.email);
            pa.setString(4,st.password);
            pa.setString(5,st.school);
            pa.setString(6,st.branch);
            pa.setString(7,st.special);
            pa.setString(8,st.lastLogIn+"");
            pa.setString(9,st.lastLogOut+"");
            pa.setString(10,st.lastLogOutTime+"");
            pa.setString(11,"0");
            pa.executeUpdate();
        }catch(SQLException e){JOptionPane.showMessageDialog(null,"Error: 404, TRY
AGAIN");}
    }

    public static boolean isPresent(String e1, String e2 , String pass)throws SQLException{
        Connection con=DataManager.connector();
        PreparedStatement pa=con.prepareStatement("select * from details");
        ResultSet rs=pa.executeQuery();
        while(rs.next()){
            String en=rs.getString(1);
            String em=rs.getString(3);
            String p=rs.getString(4);
            if(en.equalsIgnoreCase(e1) && pass.equals(p))return true;
            else if(pass.equals(p) && em.equalsIgnoreCase(e2)) return true;
        }
        return false;
    }
}

```



```

public static void del(String e){
    try{
        Connection con=DataManager.connector();

        PreparedStatement st= con.prepareStatement("DELETE FROM details WHERE
enrollmentnumber = ?");
        st.setString(1,e);
        st.executeUpdate();

    }catch(SQLException ex){JOptionPane.showMessageDialog(null,"Error: 404, TRY
AGAIN");}
}

public static void insertAt(String enroll, String columb, String value ){
    try{
        Connection con=DataManager.connector();
        String s="UPDATE details SET "+columb+" = ? WHERE enrollmentnumber = ?";
        PreparedStatement ps=con.prepareStatement(s);
        ps.setString(1,value);
        ps.setString(2,enroll);
        ps.executeUpdate();
    }catch(SQLException ex){JOptionPane.showMessageDialog(null,"Error: 404,
TRY AGAIN");}
}

public static int getAttendance(String enroll){
    try{
        Connection con=DataManager.connector();
        String s="SELECT * details WHERE enrollmentnumber = ?";
        PreparedStatement ps=con.prepareStatement(s);
        ps.setString(1, enroll);
        ResultSet rs=ps.executeQuery();
        int a=Integer.parseInt(rs.getString(11));
    }catch(SQLException ex){JOptionPane.showMessageDialog(null,"Error: 404, TRY
AGAIN");}
    return 0;
}

public static void addLogInTime(String e, String t){
    insertAt(e,"lastlogin",t);
}

public static void lastLogOut(String enroll, String value){
    insertAt(enroll,"lastlogout", value);
}

public static void lastLogInTime(String enroll, String value){
    insertAt(enroll,"lastlogintime", value);
}

```

```

    }
    public static void attendance(String enroll, String value){
        insertAt(enroll,"attendance", value);
    }

}

```

- **StudentInfo:**

```

public class StudentInfo {
    public String enrollmentNumber;
    public String name;
    public String email;
    public String password;
    public String school;
    public String branch;
    public String special;
    public String lastLogIn;
    public String lastLogOut;
    public String lastLogOutTime;
    public String attendance;

    StudentInfo(){
    }

    StudentInfo(String e,String n,String em, String pa, String s,String b,String
sp,String lli, String llo, String llot, String at ){
        name=n;
        enrollmentNumber=e;
        school=s;
        branch=b;
        special=sp;
        email=em;
        password=pa;
        lastLogIn=lli;
        lastLogOut=llo;
        lastLogOutTime=llot;
    }
}

```

```
attendance=at;
```

```
}
```

```
}
```

# IMPLEMENTATION & RESULT ANALYSIS

The following are the pictures of the GUI forms of our project:

Fig 1

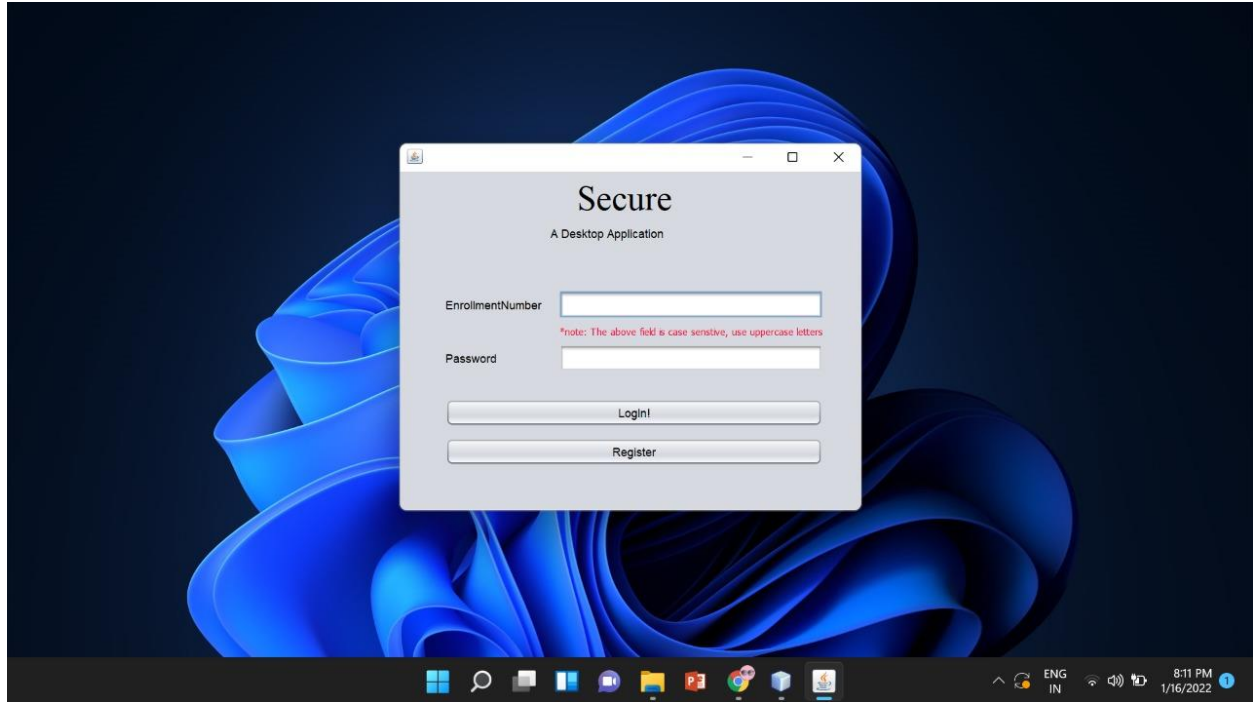


Fig 2

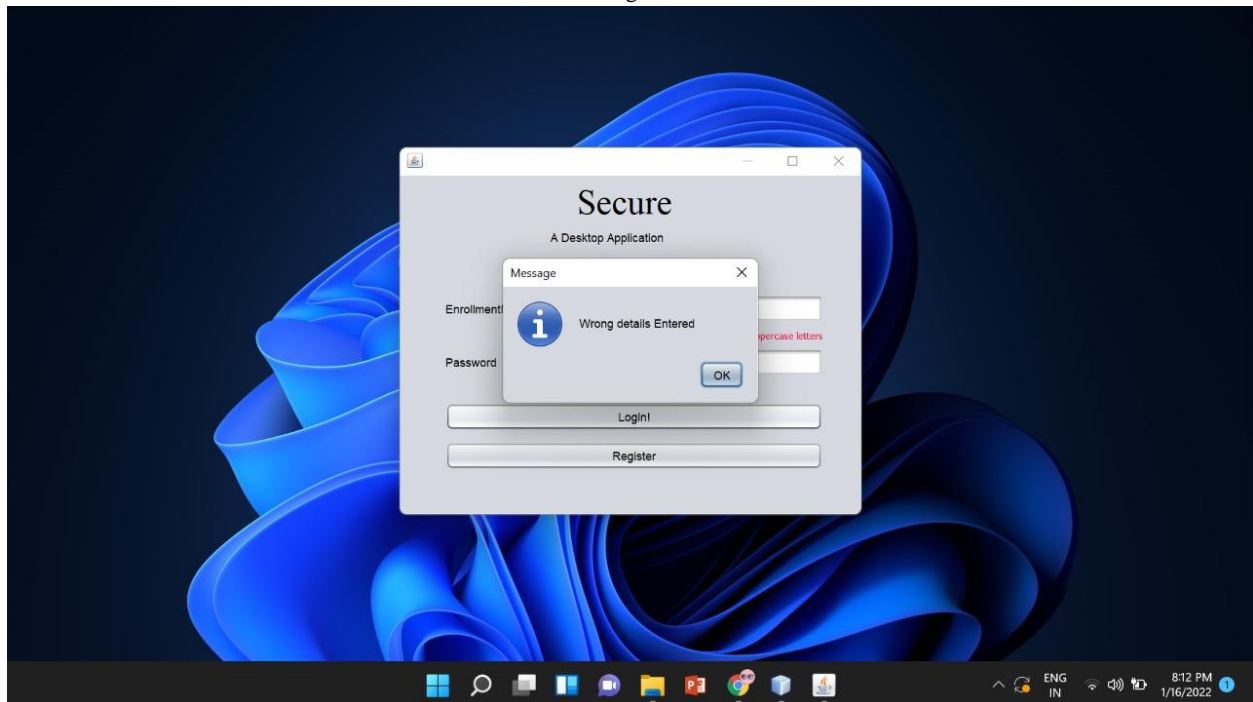


Fig 3

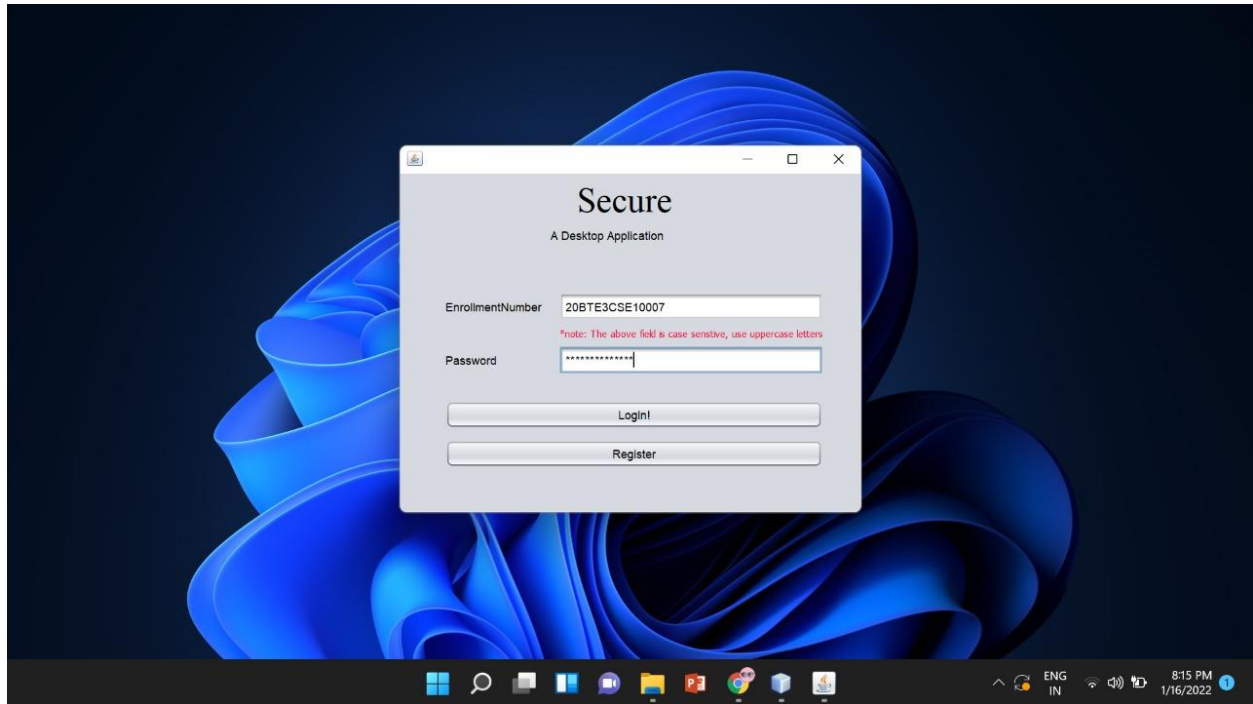


Fig 4

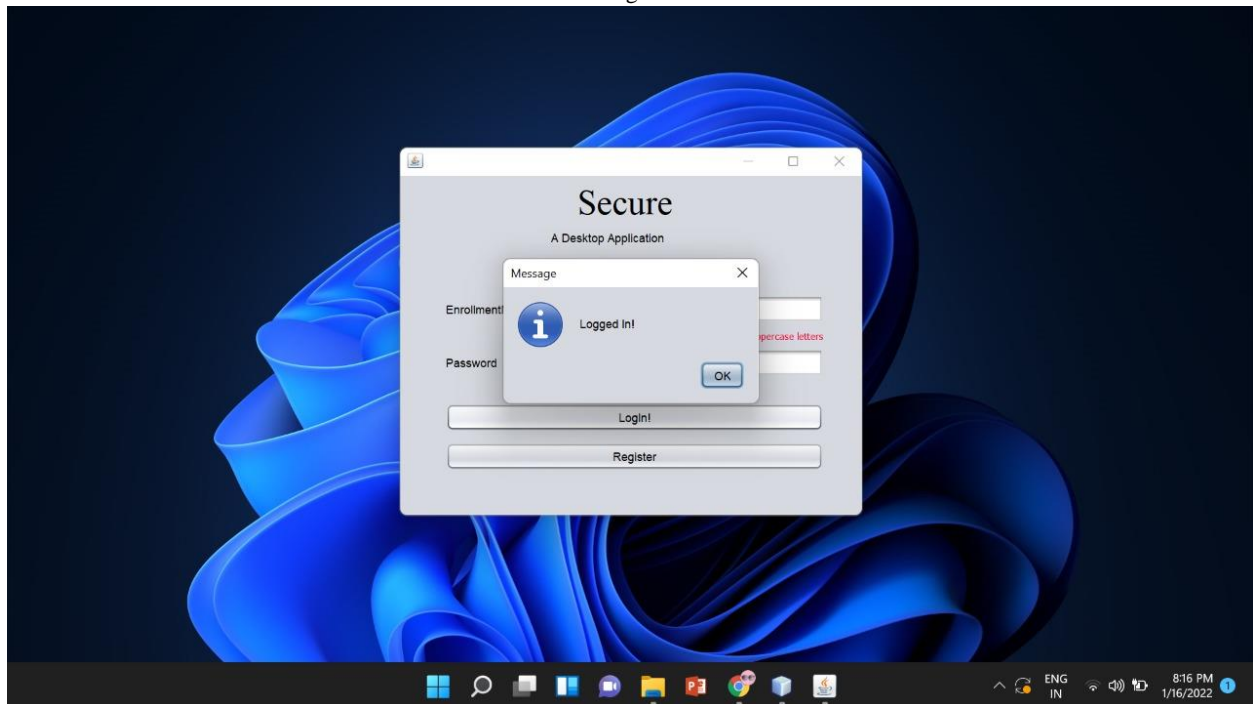
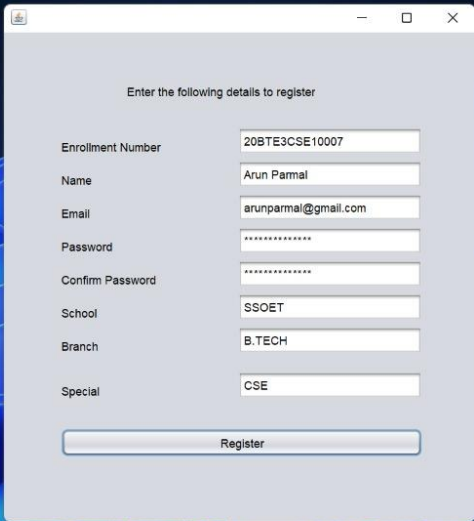


Fig 5

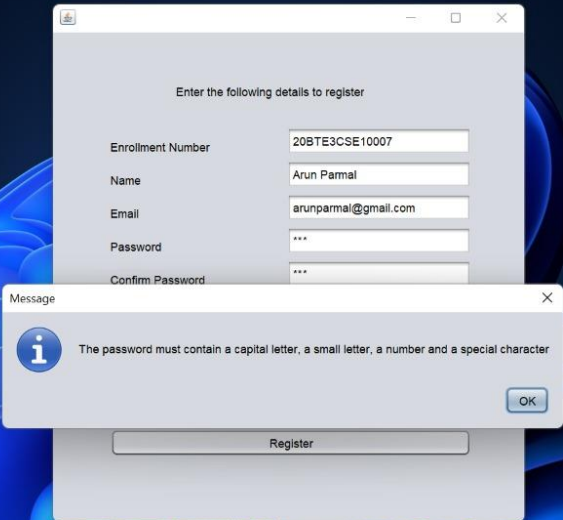


Enter the following details to register

Enrollment Number	20BTE3CSE10007
Name	Arun Parmal
Email	arunpamal@gmail.com
Password	*****
Confirm Password	*****
School	SSOET
Branch	B.TECH
Special	CSE

Register

Fig 6



Enter the following details to register

Enrollment Number	20BTE3CSE10007
Name	Arun Parmal
Email	arunpamal@gmail.com
Password	***
Confirm Password	***

Register

Message

The password must contain a capital letter, a small letter, a number and a special character

OK

Fig 7

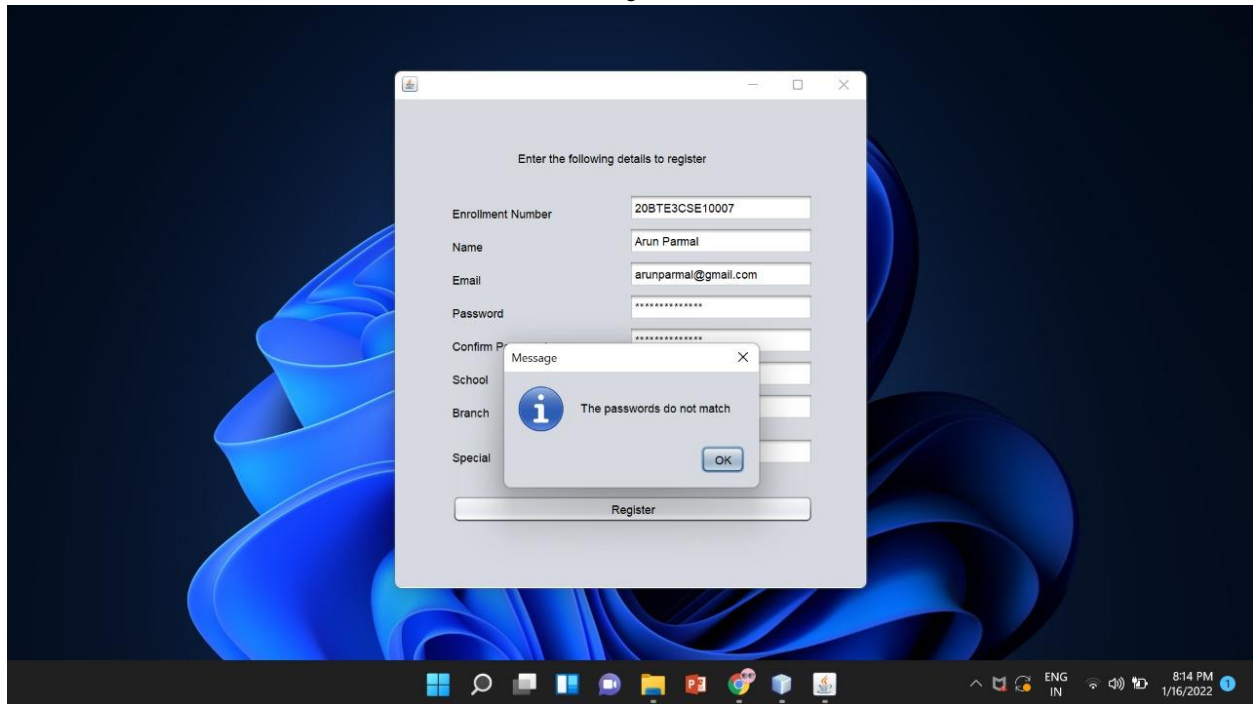


Fig 8

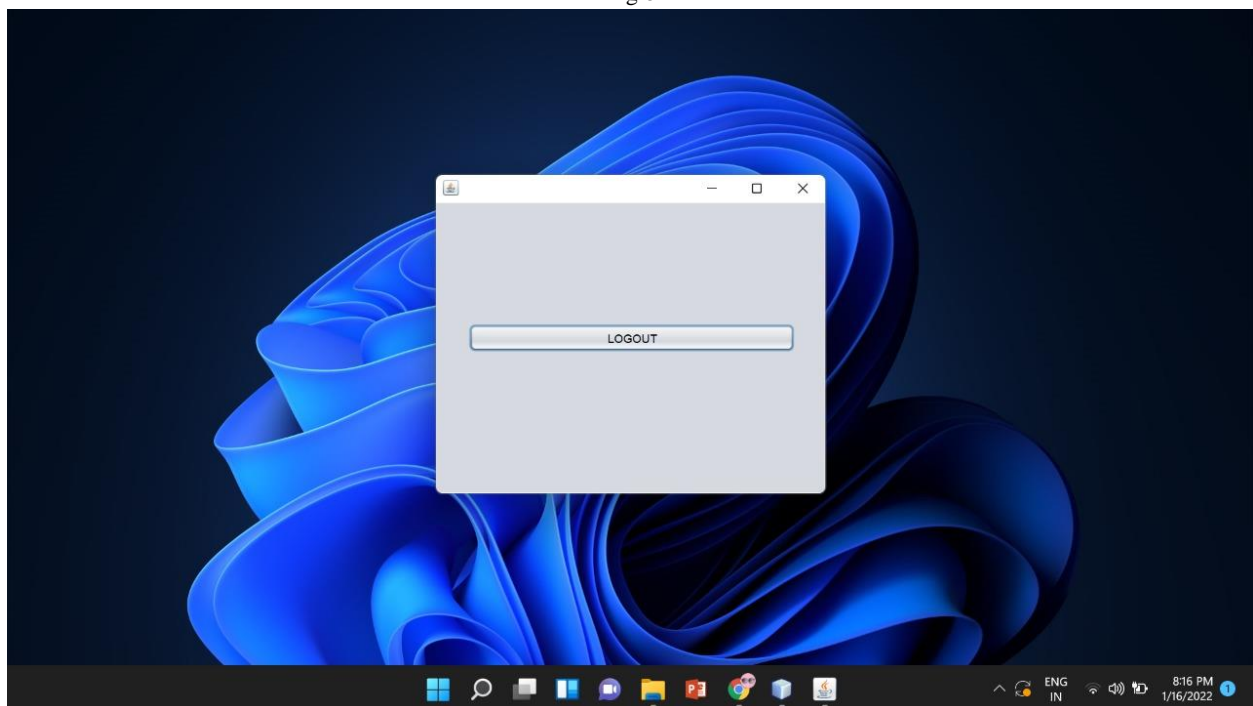




Fig 9

NetBeans IDE 8.0.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+F)

Projects Services Files

Databases

- MySQL Server at localhost:3306 [root] [discon]
- Java DB
- Drivers
- jdbc:derby://localhost:1527/sample [app on]
- jdbc:derby://localhost:1527/Secure [aryansat]
- ARYANSAHU400
  - Tables
    - DETAILS
      - ENROLLMENTNUMBER
      - NAME
      - EMAIL
      - PASSWORD
      - SCHOOL
      - BRANCH
      - SPECIAL
      - LASTLOGIN
      - LASTLOGOUT
      - LASTLOGINTIME
      - ATTENDANCE
    - Indexes
    - Foreign Keys
  - Views
  - Procedures
  - Other schemas
- Servers
- Maven Repositories
- Cloud
- Hudson Builders
- Task Repositories

Output SQL 1 [jdbc:derby://localhost:1527/Secure [aryansat]]

Connection: jdbc:derby://localhost:1527/Secure [aryansat] on ARYANSAHU400

```
1 select * from ARYANSAHU400.DETAILS;
```

select \* from ARYANSAHU400.DETAILS

Page Size: 20 Total Rows: 4 Page: 1 of 1 Matching Rows:

#	SPECIAL	LASTLOGIN	LASTLOGOUT	LASTLOGINTIME	ATTENDANCE
1	CSE	14:25:32.310287700	14:25:42.046452100	3274	0
2	CSE	13:34:13.795	13:34:20.952	5236	0
3	CSE	18:20:58.028	18:21:40.070	33067	0
4	CSE	20:15:58.681133	20:17:01.344273400	46736	0

1:1 INS

8:21 PM 1/16/2022

Fig 10

NetBeans IDE 8.0.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+F)

Projects Services Files

Databases

- MySQL Server at localhost:3306 [root] [discon]
- Java DB
- Drivers
- jdbc:derby://localhost:1527/sample [app on]
- jdbc:derby://localhost:1527/Secure [aryansat]
- ARYANSAHU400
  - Tables
    - DETAILS
      - ENROLLMENTNUMBER
      - NAME
      - EMAIL
      - PASSWORD
      - SCHOOL
      - BRANCH
      - SPECIAL
      - LASTLOGIN
      - LASTLOGOUT
      - LASTLOGINTIME
      - ATTENDANCE
    - Indexes
    - Foreign Keys
  - Views
  - Procedures
  - Other schemas
- Servers
- Maven Repositories
- Cloud
- Hudson Builders
- Task Repositories

Output SQL 1 [jdbc:derby://localhost:1527/Secure [aryansat]]

Connection: jdbc:derby://localhost:1527/Secure [aryansat] on ARYANSAHU400

```
1 select * from ARYANSAHU400.DETAILS;
```

select \* from ARYANSAHU400.DETAILS

Page Size: 20 Total Rows: 4 Page: 1 of 1 Matching Rows:

#	ENROLLMENTNUMBER	NAME	EMAIL	PASSWORD	SCHOOL	BRANCH
1	20BTE3CSE10008	Aryan Sahu	aryansahu400@gmail.com	Aryansahu@400	SSOET	B.Tech
2	20BTE3CSE10001	Aayush Khare	Aayushkhare@gmail.com	Aayushkhare@400	SSOET	B.Tech
3	20BTE3CSE10023	Piyush Bhai	piyushshastri65@gmail.com	Piyush@400	SSOET	B.Tech
4	20BTE3CSE10007	Arun Parmal	arunparmal@gmail.com	Arunparmal@400	SSOET	B.Tech

1:1 INS

8:20 PM 1/16/2022



# METHODOLOGY

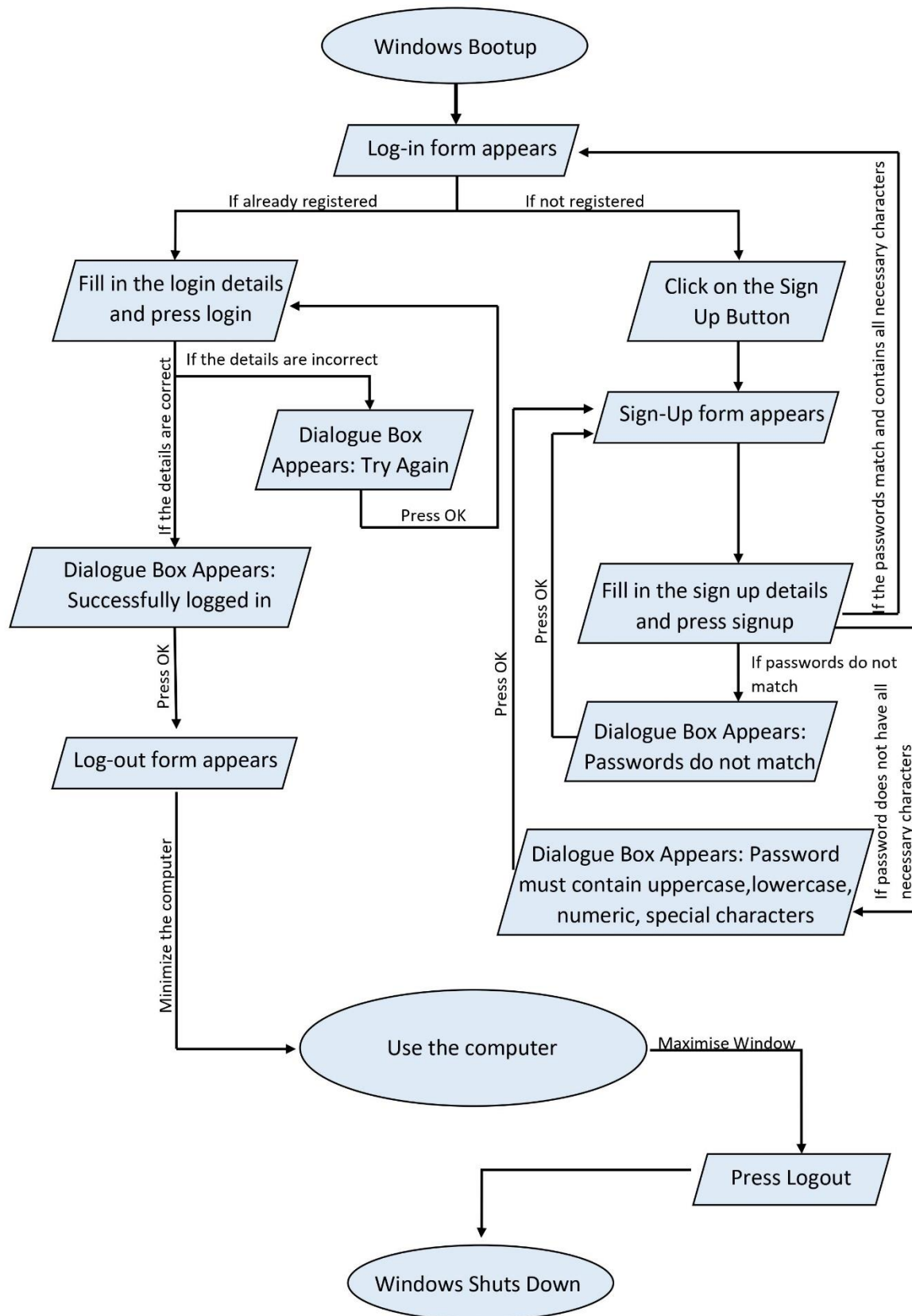


Fig 11

## CONCLUSION & FUTURE WORK

This software when implemented on all the computers of Sage university Bhopal over the intranet, will give a very secure environment for the students of the university.

It will increase the life time of any particular system and will eliminate the extra time which is usually spent in taking attendance manually.

### Future Work:

- In the future we will try to implement some more features for our software make it more efficient and try to make it for our collage labs and set up a proper server and use mysql .
- Also try to get the record of the devices which are plugged or unplugged in the system.
- Also try to add a feature so that if the student has not shut down the system properly, it will stop the user to login in the system after 3 times and it can only be allowed by the teacher from the server room.

# REFERENCE

- <https://www.geeksforgeeks.org/>
- [https://en.wikipedia.org/wiki/Main\\_Page](https://en.wikipedia.org/wiki/Main_Page)
- <https://www.w3schools.com/>
- <https://www.codecademy.com/catalog>
- <https://www.paperpass.net/>

# Plagiarism Report

1/4

PaperPass .net

## Title

Secure Plagiarism Report

20%

SIMILARITY INDEX

14%

ACADEMIC

18%

INTERNET

Date:

2022-01-31 06:24:47(+00:00 UTC)

Report ID:

61f780cb589e33892

Word count:

3928

Character count:

27178

---

**Similar sources**


---

<b>1</b>	<ul style="list-style-type: none"> <li>● How should I add a pre-loader in my java stand alone ...</li> <li>● <a href="https://stackoverflow.com/questions/60875724/how-should-i-add-a-pre-loader-in-my-java-stand-alone-application">https://stackoverflow.com/questions/60875724/how-should-i-add-a-pre-loader-in-my-java-stand-alone-application</a></li> </ul> Internet	<b>11.6%</b>
<b>2</b>	<ul style="list-style-type: none"> <li>● Netbeans GUI and SQL issue - Stack Overflow</li> <li>● <a href="https://stackoverflow.com/questions/14087314/netbeans-gui-and-sql-issue">https://stackoverflow.com/questions/14087314/netbeans-gui-and-sql-issue</a></li> </ul> Internet	<b>11.3%</b>
<b>3</b>	<ul style="list-style-type: none"> <li>● What's wrong with my GUI code? [closed] - Stack Overflow</li> <li>● <a href="https://stackoverflow.com/questions/19772309/whats-wrong-with-my-gui-code">https://stackoverflow.com/questions/19772309/whats-wrong-with-my-gui-code</a></li> </ul> Internet	<b>11.3%</b>
<b>4</b>	<ul style="list-style-type: none"> <li>● Having problems with lists in java - Stack Overflow</li> <li>● <a href="https://stackoverflow.com/questions/61577887/having-problems-with-lists-in-java">https://stackoverflow.com/questions/61577887/having-problems-with-lists-in-java</a></li> </ul> Internet	<b>11.3%</b>
<b>5</b>	<ul style="list-style-type: none"> <li>● GroupLayout z-order - Stack Overflow</li> <li>● <a href="https://stackoverflow.com/questions/39691711/grouplayout-z-order">https://stackoverflow.com/questions/39691711/grouplayout-z-order</a></li> </ul> Internet	<b>10.8%</b>
<b>6</b>	<ul style="list-style-type: none"> <li>● COCOMO-II/cocomo_sim2.java at master · Bhanups123/COCOMO-II ...</li> <li>● <a href="https://github.com/Bhanups123/COCOMO-II/blob/master/cocomo_sim2.java">https://github.com/Bhanups123/COCOMO-II/blob/master/cocomo_sim2.java</a></li> </ul> Internet	<b>9.4%</b>
<b>7</b>	<ul style="list-style-type: none"> <li>● GroupLayout\$SequentialGroup.addContainerGap - Java</li> <li>● <a href="https://www.tabnine.com/code/java/methods/javafx.swing.GroupLayout\$SequentialGroup/addContainerGap">https://www.tabnine.com/code/java/methods/javafx.swing.GroupLayout\$SequentialGroup/addContainerGap</a></li> </ul> Internet	<b>8.4%</b>
<b>8</b>	<ul style="list-style-type: none"> <li>● API library-based identification and documentation of usage patterns</li> <li>● Hamzeh Eyal Salman</li> <li>● International Journal of Computer Applications in Technology,2018</li> </ul> Academic	<b>7.9%</b>
<b>9</b>	<ul style="list-style-type: none"> <li>● Java APIs, Extensions and Libraries</li> <li>● Kishori Sharan</li> </ul> Others	<b>4.3%</b>
<b>10</b>	<ul style="list-style-type: none"> <li>● Java (programming language) - Wikipedia</li> <li>● <a href="https://en.wikipedia.org/wiki/Java_(programming_language)">https://en.wikipedia.org/wiki/Java_(programming_language)</a></li> </ul> Internet	<b>2.5%</b>
<b>11</b>	<ul style="list-style-type: none"> <li>● Pro NetBeans™ IDE 6 Rich Client Platform Edition</li> </ul> Others	<b>1.5%</b>
<b>12</b>	<ul style="list-style-type: none"> <li>● Why this java program not working properly? - Stack Overflow</li> <li>● <a href="https://stackoverflow.com/questions/62119786/why-this-java-program-not-working-properly">https://stackoverflow.com/questions/62119786/why-this-java-program-not-working-properly</a></li> </ul> Internet	<b>1.5%</b>
<b>13</b>	<ul style="list-style-type: none"> <li>● What is A Derby Database ? - DBA</li> <li>● <a href="http://dba.fyicenter.com/Interview-Questions/DERBY/What_is_A_Derby_Database_.html">http://dba.fyicenter.com/Interview-Questions/DERBY/What_is_A_Derby_Database_.html</a></li> </ul> Internet	<b>1.4%</b>
<b>14</b>	<ul style="list-style-type: none"> <li>● Practical JSF in Java EE 8</li> <li>● Michael Müller</li> </ul> Others	<b>1.2%</b>
<b>15</b>	<ul style="list-style-type: none"> <li>● JavaServer Faces</li> <li>● Josh Juneau</li> </ul>	<b>1.2%</b>

Others		
16	<ul style="list-style-type: none"> <li>● java netbeans - gists · GitHub</li> <li>● <a href="https://gist.github.com/3854312">https://gist.github.com/3854312</a></li> </ul> Internet	1.1%
17	<ul style="list-style-type: none"> <li>● Apache Derby</li> <li>● <a href="https://db.apache.org/derby/">https://db.apache.org/derby/</a></li> </ul> Internet	1.1%
18	<ul style="list-style-type: none"> <li>● Adaptation of the humanoid robot to speech disfluency therapy</li> <li>● Łukasz Kwaśniewicz, Wiesława Kuniszyk-Jóźkowiak, Grzegorz M. Wójcik, Jolanta Masiak</li> <li>● Bio-Algorithms and Med-Systems, 2016</li> </ul> Academic	1.0%
19	<ul style="list-style-type: none"> <li>● Java EE 8 Recipes</li> <li>● Josh Juneau</li> </ul> Others	1.0%
20	<ul style="list-style-type: none"> <li>● Java 8 Recipes</li> <li>● Josh Juneau</li> </ul> Others	1.0%
21	<ul style="list-style-type: none"> <li>● Java EE 7 Recipes</li> <li>● Josh Juneau</li> </ul> Others	1.0%
22	<ul style="list-style-type: none"> <li>● Beginning EJB in Java EE 8</li> <li>● Jonathan Wetherbee, Massimo Nardone, Chirag Rathod, Raghu Kodali</li> </ul> Others	1.0%
23	<ul style="list-style-type: none"> <li>● Elementor #647 - SUSHIL LEKHI</li> <li>● <a href="https://sushillekhi.com/index.php/elementor-647/">https://sushillekhi.com/index.php/elementor-647/</a></li> </ul> Internet	0.9%
24	<ul style="list-style-type: none"> <li>● Tuberculosis–Diagnostic Expert System: An architecture for translating patients information from the web for use in tuberculosis diagnosis</li> <li>● Victor C Osamor, Ambrose AAzeta, Oluseyi O Ajulo</li> <li>● Health Informatics Journal, 2014</li> </ul> Academic	0.8%
25	<ul style="list-style-type: none"> <li>● GroupLayout\$SequentialGroup.addComponent - Java - Tabnine</li> <li>● <a href="https://www.tabnine.com/code/java/methods/javafx.swing.GroupLayout\$SequentialGroup/addComponent">https://www.tabnine.com/code/java/methods/javafx.swing.GroupLayout\$SequentialGroup/addComponent</a></li> </ul> Internet	0.8%
26	<ul style="list-style-type: none"> <li>● Beginning Java™ SE 6 Platform</li> </ul> Others	0.7%
27	<ul style="list-style-type: none"> <li>● Practical Java Machine Learning</li> <li>● Mark Wickham</li> </ul> Others	0.7%
28	<ul style="list-style-type: none"> <li>● Programming platform for NAO robot in cognitive interaction applications</li> <li>● Mohd Azfar Miskam, Syamimi Shamsuddin, Hanafiah Yussof, Abdul Rahman Omar, Muhammad Zaiyad Mud...</li> <li>● 2014 IEEE International Symposium on Robotics and Manufacturing Automation (ROMA), 2014</li> </ul> Others	0.7%
29	<ul style="list-style-type: none"> <li>● OGC-to-W3C Services: A Wrapper-Based Solution for Geospatial Metadata Exchange</li> <li>● Pasquale Giovanni, Michela Bertolotto, Monica Sebillio, Giuliana Vitiello</li> <li>● Geosciences, 2018</li> </ul> Academic	0.7%

<b>30</b>	<ul style="list-style-type: none"> <li>● Practical Microservices Architectural Patterns</li> <li>● Binildas Christudas</li> </ul> Others	<b>0.6%</b>
<b>31</b>	<ul style="list-style-type: none"> <li>● About: Apache Derby - DBpedia</li> <li>● <a href="https://dbpedia.org/page/Apache_Derby">https://dbpedia.org/page/Apache_Derby</a></li> </ul> Internet	<b>0.6%</b>
<b>32</b>	<ul style="list-style-type: none"> <li>● Java Code Examples for javax.swing.GroupLayout.Alignment</li> <li>● <a href="https://www.programcreek.com/java-api-examples/?api=javax.swing.GroupLayout.Alignment">https://www.programcreek.com/java-api-examples/?api=javax.swing.GroupLayout.Alignment</a></li> </ul> Internet	<b>0.6%</b>
<b>33</b>	<ul style="list-style-type: none"> <li>● Web based distant monitoring and control for greenhouse systems using the Sun SPOT modules</li> <li>● Simon Janos, Goran Martinovic</li> <li>● 2009 7th International Symposium on Intelligent Systems and Informatics, 2009</li> </ul> Others	<b>0.5%</b>
<b>34</b>	<ul style="list-style-type: none"> <li>● Java (programming language) - LinkedIn</li> <li>● <a href="https://www.linkedin.com/pulse/java-programming-language-soumik-dmgsolution-1e?trk=public_profile_article_view">https://www.linkedin.com/pulse/java-programming-language-soumik-dmgsolution-1e?trk=public_profile_article_view</a></li> </ul> Internet	<b>0.5%</b>