Contents

ABSTRACT

Technology is an important part of every day's life. Our project is aims to deliver daily notices, and result of students to respective student and all the teachers at tip of theair fingers.

Every participant in this program need to register himself first by creating a account. Creating a account comprises of filling essential details about self. Both teachers and student will create account which would be forwarded to HOD for verification and activation of account along with activation of account Roll no. and Teacher ID would be assigned to student and teachers respectively automatically. Each username of teacher or student will be unique.

On dashboard of every participant different option will be available along with their provided details.Teachers can view result of all the students by applying different filters such as 'year of study' or PRN of student. Student can view their respective results only. Only teachers can post/ delete notices from the common noticeboard which will be visible to all the participants..

All teachers can view complete data of student. Various filters such as 'year of study', 'passing year', or 'PRN' are available to filter out the students. HOD can view complete details about students and also teachers.Data of 'graduated student' also remains in the same system which in future can be retrieved by year of passing(batch) or PRN number of student. Analysis of result is available for every personal result of student.

1. INTRODUCTION

1.1 Problem Definition

Need for system:

This project can be deployed in college for every day use. Result of student will be available with teachers and students . Data will be accessible to all the people all the time at a single place. Important notices can be posted on the common noticeboards through which the message can be delivered to all the students at once. To keep track of every student, teachers can use this system to view semester wise report of student at one place. Ease in managing the the data of passed out student . Data of the passed is stored in system with the unique PRN student.

Basically this system can be deployed in every college or similar type of program can be used in other organizations which have admin and employee hierarcy.

2. SPECIFIC REQUIRMENTS

The system analysis contains a planning and design phases where a logical design of system is developed and to work accordingly a plan is established. Also the requirements of system are identified and the operating environment is identified.

2.1 Hardware Requirements

- Windows 10 & Windows 7 Operating System.
- 1 GB RAM
- Intel®core2duo processor@3.4GHz
- 200MB memory Space

2.2 SoftwareRequirements

- MySql
- Netbeans IDE oJava
- MySql connector Java
- jCalander library
- JDK (1.8)

3. SOFTWARE USED

3.1 Java

Java is a programming language originally developed by James Gosling at Sun Microsystems (which is now a subsidiary of Oracle Corporation) and released in 1995 as a core component of Sun Microsystems Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code (class file) that can run on any Java Virtual Machine (JVM) regardless of computer architecture. Java is a general-purpose, concurrent, class-based, object-oriented language that is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run any- where". The java is independent to platform so it's important. Java is currently one of the most popular programming languages in use, and is widely used from application software to web
applications.

James Gosling, Mike Sheridan, and Patrick Naught on initiated the Java language project in June 1991. Java was originally designed for interactive television, but it was too advanced for the digital cable television industry at the time. The language was initially called Oak after an oak tree that stood outside Gosling's office; it went by the name Green later, and was later renamed Java, from a list of random words. Gosling aimed to implement a virtual machine and a language that had a familiar C/C++
of notation.

Java is an object-oriented programming language developed by Sun Microsystems in 1990s. Since then, Java has gained enormous popularity as a computer language. Java was chosen as the programming language for network computers. It is a universal front end for enterprise database. Sun Microsystems states that, "Java is a simple, object-oriented,

distributed, secure, architecture, robust, multi-threaded and dynamic language. The program can be written once, and run anywhere". One of the most significant advantages of Java is that, it has the ability to move easily from one computer to another. It also has the ability to run the same program on many different operating systems. With such exemplary benefits, Java is a hot favourite among techies and software professional sit allows you to create modular programs and reusable codes.

Java Features

1] Simple, Small and familiar:

Java is a simple and small language. The Syntax of java is just like C++, so it is very easy to learn. It is simple because it i) does not use header files ii) eliminated the use of pointer iii) operator overloading and virtual base classes are eliminated.

2] Object oriented:

Java is a pure Object oriented. Everything in java is object. All programs and data reside inside objects and classes

3] Distributed:

Java has networking facilities. so java can create application on network.

4] Robust:

java gives importance to memory management by using the technique called Garbage Collection and Exception handling.

5] Secure:

since java is used on internet, security is an important issue. A security code is asked before a java code is interpreted on internet.

6] Platform independent:

Java compiler generates a platform independent code called byte code.

7] Portable:

The Byte code generated by java can be used on any machine. So it can be portable.

8] Compiled and Interpreted:

Generally, computer languages are either complied or interpreted. But java combines both compiler and interpreted.

9] High performance:

The use of byte code makes the performance high. the speed is also high with comparing c, c++.

10] Multithreading and interactive:

Multithreading means handling more than one job at a time. Java supports Multithreading.

11] Dynamic and extensible:

Java is a dynamic language. So it is capable of linking dynamic new classes, methods and objects. Java supports functions written in C and C++ also. These functions are called native methods. During Run-Time Native methods can be linked dynamically.

3.2 MySql

MySQl is an open-source relational database management system (RDBMS).Its name is a combination of

"My", the name of co-founder Michael Widenius's daughter,and "SQL", the abbreviation for Structured Query Language.

MySQL is free and open-source software under the terms of the GNU General Public License, and is also
available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company
MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation).In 2010, when Oracle
acquired Sun, Widenius forked the open-source MySQL project to create MariaDB.
MySQL is a component of the LAMP web application software stack (and others), which is an acronym for
Linux, Apache, MySQL, Perl/PHP/Python. MySQL is used by many database-driven web applications,
including Drupal, Joomla, phpBB, and WordPress. MySQL is also used by many popular websites, including
Facebook,Flickr,MediaWiki, Twitter,and YouTube.

MySQL is written in C and C++. Its SQL parser is written in yacc, but it uses a home-brewed lexical
analyzer.MySQL works on many system platforms, including AIX, BSDi, FreeBSD, HP-UX, eComStation,
i5/OS, IRIX, Linux, macOS, Microsoft Windows, NetBSD, Novell NetWare, OpenBSD, OpenSolaris, OS/2
Warp, QNX, Oracle Solaris, Symbian, SunOS, SCO OpenServer, SCO UnixWare, Sanos and Tru64. A port of
MySQL to OpenVMS also exists.

The MySQL server software itself and the client libraries use dual-licensing distribution. They are offered under GPL version 2,or a proprietary license.

Support can be obtained from the official manual.Free support additionally is available in different IRC channels and forums. Oracle offers paid support via its MySQL Enterprise

products. They differ in the scope of services and in price. Additionally, a number of third party organisations exist to provide support and services, including MariaDB and Percona.

MySQL has received positive reviews, and reviewers noticed it "performs extremely well in the average case" and that the "developer interfaces are there, and the documentation (not to mention feedback in the real world via Web sites and the like) is very, very good".It has also been tested to be a "fast, stable and true multi-user, multi-threaded sql database server".

MySql Features

MySQL is offered under two different editions: the open source MySQL Community Server and the proprietary Enterprise Server.MySQL Enterprise Server is differentiated by a series of proprietary extensions which install as server plugins, but otherwise shares the version numbering system and is built from the same code base.

Major features as available in MySQL 8.0:

- A broad subset of ANSI SQL 99, as well as extensions
- Cross-platform support
- Stored procedures, using a procedural language that closely adheres to SQL/PSM
- Triggers
- Cursors
- Updatable views
- Online Data Definition Language (DDL) when using the InnoDB Storage Engine.
- Information schema
- Performance Schema that collects and aggregates statistics about server execution and query performance for monitoring purposes.
- A set of SQL Mode options to control runtime behavior, including a strict mode to better adhere to SQL standards.
- X/Open XA distributed transaction processing (DTP) support; two phase commit as part of

this, using the default InnoDB storage engine

- Transactions with savepoints when using the default InnoDB Storage Engine. The NDB Cluster Storage Engine also supports transactions.
- ACID compliance when using InnoDB and NDB Cluster Storage Engines.
- SSL support
- Query caching
- Sub-SELECTs (i.e. nested SELECTs)
- Built-in replication support
- Asynchronous replication: master-slave from one master to many slaves or many masters to one slave
- Semi synchronous replication: Master to slave replication where the master waits on replication.
- Synchronous replication: Multi-master replication is provided in MySQL Cluster.
- Virtual Synchronous: Self managed groups of MySQL servers with multi master support can be done using: Galera Cluster or the built in Group Replication plugin.
- Full-text indexing and searching
- Embedded database library
- Unicode support
- Partitioned tables with pruning of partitions in optimizer
- Shared-nothing clustering through MySQL Cluster
- Multiple storage engines, allowing one to choose the one that is most effective for each table in the application.
- Native storage engines InnoDB, MyISAM, Merge, Memory (heap), Federated, Archive, CSV, Blackhole, NDB Cluster.
- Commit grouping, gathering multiple transactions from multiple connections together to increase the number of commits per second.
- The developers release minor updates of the MySQL Server approximately every two months. The sources can be obtained from MySQL's website or from MySQL's GitHub repository, both under the GPL license.

3.3 NetBeans

NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called *modules*. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, it has extensions for other languages like PHP, C, C++, HTML5,[4] and JavaScript. Applications based on NetBeans, including the NetBeans IDE, can be extended by third party developers.

The NetBeans Platform is a framework for simplifying the development of Java Swing desktop applications.

The NetBeans IDE bundle for Java SE contains what is needed to start developing NetBeans plugins and

NetBeans Platform based applications; no additional SDK is required.

Applications can install modules dynamically. Any application can include the Update Center module to allow users of the application to download digitally signed upgrades and new features directly into the running application. Reinstalling an upgrade or a new release does not force users to download the entire application again.

The platform offers reusable services common to desktop applications, allowing developers to focus on the logic specific to their application. Among the features of the platform are:

- User interface management (e.g. menus and toolbars)
- User settings management
- Storage management (carries out efficient storage)
- Window management
- Wizard framework (supports step-by-step dialogs)
- NetBeans Visual Library
- Integrated development tools

4. TABLE FORMAT

Login student table:

Field	Type	Null	Key	Default	Extra
UserID	varchar(15)	NO	PRI	NULL	
Password	blob	NO		NULL	
Status	tinyint(4)	NO		0	
Secure1	blob	YES		NULL	
Secure2	blob	YES		NULL	
Secure3	blob	YES		NULL	

Login teacher table:

Field	Type	Null	Key	Default	Extra
UserID	varchar(15)	NO	PRI	NULL	
Password	varchar(15)	NO		NULL	
Status	tinyint(4)	NO		0	
Secure1	varchar(15)	YES		NULL	
Secure2	varchar(15)	YES		NULL	
Secure3	varchar(15)	YES		NULL	

Student table:

Field	Type	Null	Key	Default	Extra
UserID	varchar(15)	NO	UNI	NULL	
PRN	varchar(20)	NO	PRI	NULL	
Rollno	int(11)	YES	UNI	NULL	
FirstName	varchar(15)	NO		NULL	
LastName	varchar(15)	NO		NULL	
MiddleName	varchar(15)	YES		Unknown	
Year	varchar(20)	NO		NULL	
batch	varchar(45)	YES		NULL	
Division	tinyint(4)	NO		NULL	
Contact	bigint(20)	YES		NULL	
Gender	varchar(7)	YES		Unknown	
DOB	date	YES		NULL	

Teacher table:

Field	Type	Null	Key	Default	Extra
UserID	varchar(25)	NO	PRI	NULL	
Teacherid	int(11)	YES	UNI	NULL	
FirstName	varchar(10)	NO		NULL	
LastName	varchar(10)	NO		NULL	
MiddleName	varchar(10)	YES		Unknown	
Contact	bigint(15)	YES		NULL	
Designation	varchar(25)	YES		Unknown	
Experience	int(4)	YES		NULL	
Qualification	varchar(25)	YES		Unknown	
DOB	date	YES		NULL	
Gender	char(7)	YES		Unknown	
MaritalStatus	varchar(25)	YES		Unknown	

FE Result table:

Field	Type	Null	Key	Default	Extra
PRN	varchar(15)	NO	PRI	NULL	
maths1	int(11)	YES		NULL	
physics	int(11)	YES		NULL	
graphics	int(11)	YES		NULL	
electronics	int(11)	YES		NULL	
bce	int(11)	YES		NULL	
maths2	int(11)	YES		NULL	
chemistry	int(11)	YES		NULL	
bme	int(11)	YES		NULL	
electrical	int(11)	YES		NULL	
mechanics	int(11)	YES		NULL	

SE Result table:

Field	Type	Null	Key	Default	Extra
PRN	varchar(15)	NO	PRI	NULL	
DS	int(11)	YES		NULL	
COA	int(11)	YES		NULL	
DELD	int(11)	YES		NULL	
FDS	int(11)	YES		NULL	
PSOOP	int(11)	YES		NULL	
Maths3	int(11)	YES		NULL	
CG	int(11)	YES		NULL	
PAI	int(11)	YES		NULL	
DSF	int(11)	YES		NULL	
FCCN	int(11)	YES		NULL	

TE Result table:

Field	Type	Null	Key	Default	Extra
PRN	varchar(15)	NO	PRI	NULL	
TOC	int(11)	YES		NULL	
SEPM	int(11)	YES		NULL	
HCI	int(11)	YES		NULL	
DBMS	int(11)	YES		NULL	
OS	int(11)	YES		NULL	
CNT	int(11)	YES		NULL	
SP	int(11)	YES		NULL	
DAA	int(11)	YES		NULL	
CC	int(11)	YES		NULL	
DSBDA	int(11)	YES		NULL	

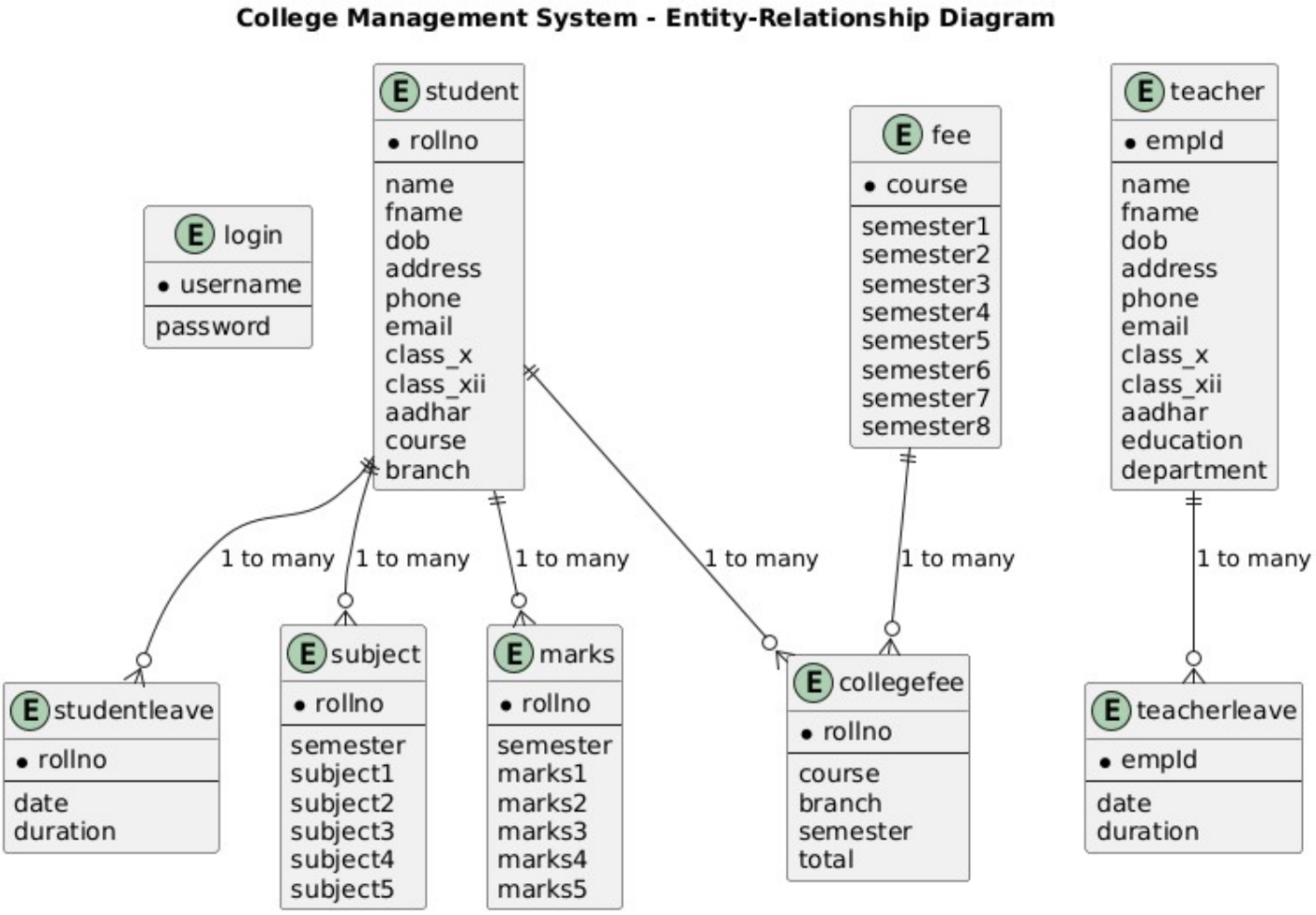
BE Result table:

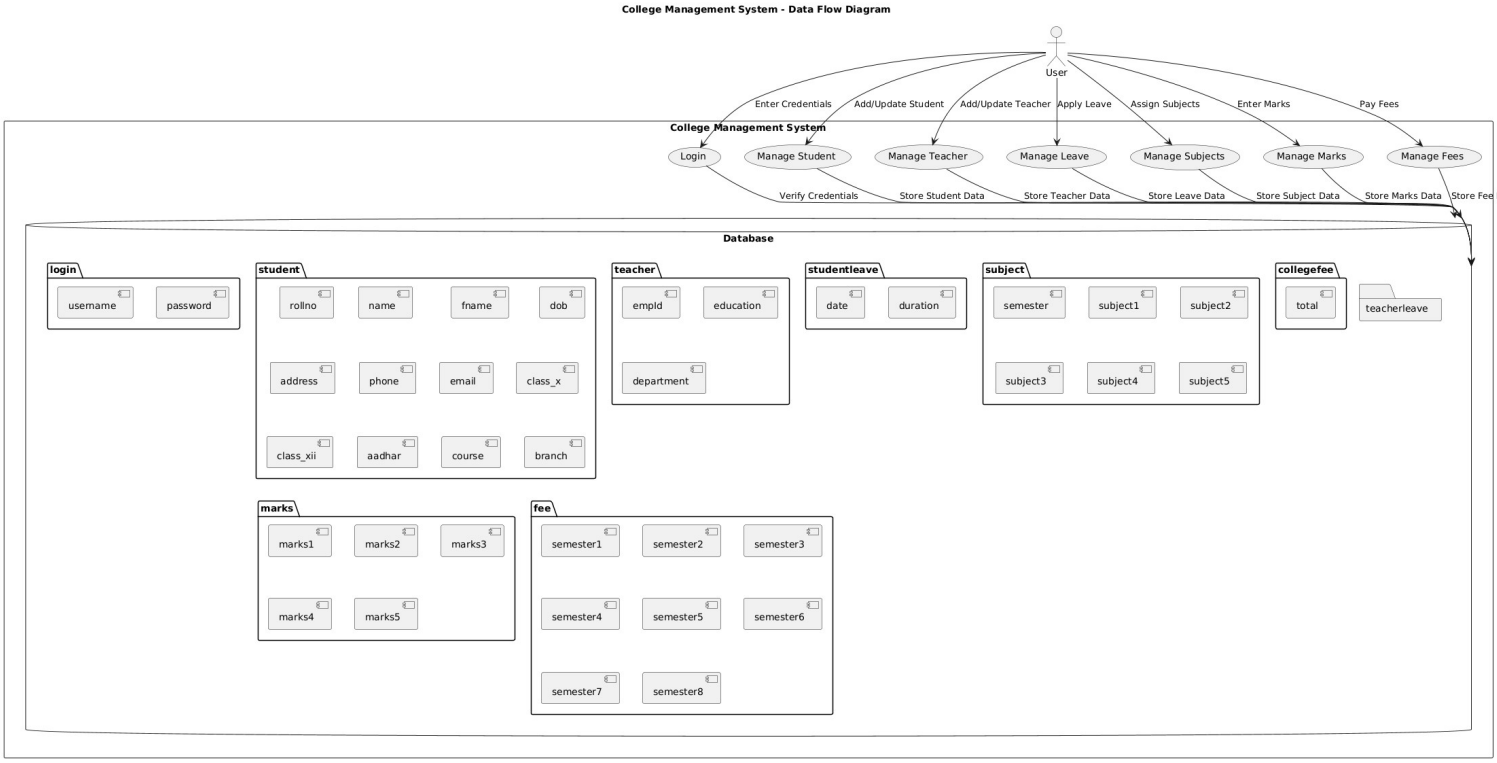
Field	Type	Null	Key	Default	Extra
PRN	varchar(15)	NO	PRI	NULL	
InfoSecurity	int(11)	YES		NULL	
ML	int(11)	YES		NULL	
Softdesign	int(11)	YES		NULL	
Elective1	int(11)	YES		NULL	
Elective2	int(11)	YES		NULL	
DCS	int(11)	YES		NULL	
UC	int(11)	YES		NULL	
Elective3	int(11)	YES		NULL	
Elective4	int(11)	YES		NULL	

Noticeboard table:

Field	Type	Null	Key	Default	Extra
MessageID	int(11)	NO	PRI	NULL	auto_increment
name	varchar(15)	NO		NULL	
datestamp	timestamp(6)	NO		NULL	
message	varchar(255)	YES		NULL	
TeacherID	int(11)	YES	MUL	NULL	

- ER Diagram


College Management System - Entity-Relationship Diagram
student
• rollno
name
fname
dob
address
phone
email
class_x
class_xii
aadhar
course
branch
login
• username
password
fee
• course
semester1
semester2
semester3
semester4
semester5
semester6
semester7
semester8
teacher
• empId
name
fname
dob
address
phone
email
class_x
class_xii
aadhar
education
department
1 to many
1 to many
1 to many
1 to many
1 to many
1 to many
studentleave
• rollno
date
duration
subject
• rollno
semester
subject1
subject2
subject3
subject4
subject5
marks
• rollno
semester
marks1
marks2
marks3
marks4
marks5
collegefee
• rollno
course
branch
semester
total
teacherleave
• empId
date
duration


College Management System - Data Flow Diagram
User
Enter Credentials
Add/Update Student
Add/Update Teacher
Apply Leave
Assign Subjects
Enter Marks
Pay Fees
College Management System
Login
Manage Student
Manage Teacher
Manage Leave
Manage Subjects
Manage Marks
Manage Fees
Verify Credentials
Store Student Data
Store Teacher Data
Store Leave Data
Store Subject Data
Store Marks Data
Store Fee Data
Database
login
username
password
student
rollno
name
fname
dob
address
phone
email
class_x
class_xii
aadhar
course
branch
teacher
empId
education
department
studentleave
date
duration
subject
semester
subject1
subject2
subject3
subject4
subject5
collegefee
total
teacherleave
marks
marks1
marks2
marks3
marks4
marks5
fee
semester1
semester2
semester3
semester4
semester5
semester6
semester7
semester8

6. SAMPLE CODE

(Login page sample code):

```
package college;
import com.sun.org.apache.xalan.internal.xsltc.runtime.BasisLibrary;
import javax.swing.ButtonGroup;
import javax.swing.JOptionPane;
import java.sql.*;
import java.util.logging.Level;
import java.util.logging.Logger;
/**
* @author Tanay
*/
public class Login extends javax.swing.JFrame {
/**
* Creates new form Login
*/
private final ButtonGroup G;
public String  loginAs;
public Login() {
initComponents();
G = new ButtonGroup();
G.add(jRadioButton1);
G.add(jRadioButton2);
}
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
private void initComponents() {
jPanel1 = new javax.swing.JPanel();
jLabel2 = new javax.swing.JLabel();
jLabel9 = new javax.swing.JLabel();
jRadioButton1 = new javax.swing.JRadioButton();
jRadioButton2 = new javax.swing.JRadioButton();
jLabel3 = new javax.swing.JLabel();
jLabel4 = new javax.swing.JLabel();
jPasswordField1 = new javax.swing.JPasswordField();
jLabel6 = new javax.swing.JLabel();
jLabel7 = new javax.swing.JLabel();
jLabel8 = new javax.swing.JLabel();
jTextField1 = new javax.swing.JTextField();
jLabel5 = new javax.swing.JLabel();
jLabel10 = new javax.swing.JLabel();
jLabel1 = new javax.swing.JLabel();
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setTitle("Login Page");
setMaximumSize(new java.awt.Dimension(1920, 1080));
setMinimumSize(new java.awt.Dimension(1920, 1080));
getContentPane().setLayout(null);
jPanel1.setMaximumSize(new java.awt.Dimension(1920, 1080));
jPanel1.setMinimumSize(new java.awt.Dimension(1920, 1080));
jPanel1.setRequestFocusEnabled(false);
jPanel1.setLayout(null);
jLabel2.setFont(new java.awt.Font("Book Antiqua", 0, 80)); // NOI18N
```

```
jLabel2.setText("Login");
jPanel1.add(jLabel2);
jLabel2.setBounds(580, 90, 210, 170);
jLabel9.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Images/Backward
    Arrow.png"))); // NOI18N
jLabel9.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
jLabel9.addMouseListener(new java.awt.event.MouseAdapter() {
public void mousePressed(java.awt.event.MouseEvent evt) {
jLabel9MousePressed(evt);
}
});
jPanel1.add(jLabel9);
jLabel9.setBounds(50, 10, 60, 80);
jRadioButton1.setFont(new java.awt.Font("HP Simplified Light", 0, 36)); // NOI18N
jRadioButton1.setText("Teacher");
jRadioButton1.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
jRadioButton1.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEvent evt) {
jRadioButton1ActionPerformed(evt);}
});
jPanel1.add(jRadioButton1);
jRadioButton1.setBounds(690, 260, 150, 50);
jRadioButton2.setFont(new java.awt.Font("HP Simplified Light", 0, 36)); // NOI18N
jRadioButton2.setText("Student");
jRadioButton2.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
jRadioButton2.addActionListener(new java.awt.event.ActionListener() {
```

```
public void actionPerformed(java.awt.event.ActionEvent evt) {
jRadioButton2ActionPerformed(evt);}
});
jPanel1.add(jRadioButton2);
jRadioButton2.setBounds(510, 260, 150, 50);
jLabel3.setFont(new java.awt.Font("Malgun Gothic", 0, 44)); // NOI18N
jLabel3.setText("Username");
jPanel1.add(jLabel3);
jLabel3.setBounds(470, 350, 280, 50);
jLabel4.setFont(new java.awt.Font("Malgun Gothic", 0, 44)); // NOI18N
jLabel4.setText("Password");
jPanel1.add(jLabel4);
jLabel4.setBounds(470, 530, 290, 50);
jPasswordField1.setBackground(new java.awt.Color(221, 221, 221));
jPasswordField1.setFont(new java.awt.Font("Monospaced", 0, 36)); // NOI18N
jPasswordField1.setBorder(null);
jPanel1.add(jPasswordField1);
jPasswordField1.setBounds(470, 600, 430, 60);
jLabel6.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Images/Forward
    Arrow.png"))); // NOI18N
jLabel6.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
jLabel6.addMouseListener(new java.awt.event.MouseAdapter() {
public void mousePressed(java.awt.event.MouseEvent evt) {
jLabel6MousePressed(evt);}
});
jPanel1.add(jLabel6);
```

```
jLabel6.setBounds(660, 670, 50, 70);
jLabel7.setFont(new java.awt.Font("Tahoma", 0, 15)); // NOI18N
jLabel7.setText("Create Account.");
jLabel7.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
jLabel7.addMouseListener(new java.awt.event.MouseAdapter() {
public void mousePressed(java.awt.event.MouseEvent evt) {
jLabel7MousePressed(evt);}
});
jPanel1.add(jLabel7);
jLabel7.setBounds(650, 750, 110, 40);
jLabel8.setFont(new java.awt.Font("Tahoma", 0, 15)); // NOI18N
jLabel8.setText("Forgot Password ?");
jLabel8.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
jLabel8.addMouseListener(new java.awt.event.MouseAdapter() {
public void mousePressed(java.awt.event.MouseEvent evt) {
jLabel8MousePressed(evt);
}
});
jPanel1.add(jLabel8);
jLabel8.setBounds(780, 750, 120, 40);
jTextField1.setBackground(new java.awt.Color(221, 221, 221));
jTextField1.setFont(new java.awt.Font("Monospaced", 0, 30)); // NOI18N
jTextField1.setBorder(null);
jPanel1.add(jTextField1);
jTextField1.setBounds(470, 420, 430, 60);
jLabel5.setFont(new java.awt.Font("Verdana", 0, 20)); // NOI18N
```

```
jLabel5.setVerticalAlignment(javax.swing.SwingConstants.TOP);
jLabel5.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0),
    2));
jPanel1.add(jLabel5);
jLabel5.setBounds(450, 240, 470, 550);
jLabel10.setText("Home");
jPanel1.add(jLabel10);
jLabel10.setBounds(60, 70, 50, 30);
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Images/Login.jpg"))); //
    NOI18N
jLabel1.setToolTipText("");
jLabel1.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
jLabel1.setMaximumSize(new java.awt.Dimension(1920, 1080));
jLabel1.setMinimumSize(new java.awt.Dimension(1920, 1080));
jLabel1.setPreferredSize(new java.awt.Dimension(1920, 1080));
jLabel1.setVerticalTextPosition(javax.swing.SwingConstants.TOP);
jPanel1.add(jLabel1);
jLabel1.setBounds(0, 0, 1920, 1080);
getContentPane().add(jPanel1);
jPanel1.setBounds(0, 0, 1920, 1080);
pack();
}// </editor-fold>
private void jRadioButton1ActionPerformed(java.awt.event.ActionEvent evt) {
if(jRadioButton1.isSelected()==true)
{loginAs= " loginteacher ";
```

```
}// TODO add your handling code here:
}
private void jLabel7MousePressed(java.awt.event.MouseEvent evt) {
this.dispose();
CreateAccount c = new CreateAccount();
c.setVisible(true);// TODO add your handling code here:
}
private void jLabel8MousePressed(java.awt.event.MouseEvent evt) {
ForgotPassword1 FP= new ForgotPassword1();
FP.setVisible(true);
this.dispose(); // TODO add your handling code here:
}
private void jLabel9MousePressed(java.awt.event.MouseEvent evt) {
Welcome l= new Welcome();
l.setVisible(true);
this.dispose();// TODO add your handling code here:
}
private void jRadioButton2ActionPerformed(java.awt.event.ActionEvent evt) {
if(jRadioButton2.isSelected()==true)
{loginAs= " loginstudent ";
}// TODO add your handling code here:
}
private void jLabel6MousePressed(java.awt.event.MouseEvent evt) {
String UserID= jTextField1.getText();
UserID=UserID.toLowerCase();
String Password = String.valueOf(jPasswordField1.getPassword());
```

```
try{
//setup connection with database
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con=DriverManager.getConnection(
"jdbc:mysql://localhost:3306/college","Student","Student@99");
boolean alreadyExists= false;
boolean status=false;
boolean PassW=false;
PreparedStatement stmt=con.prepareStatement
("select * from "+loginAs+" where UserID = ? and Status=1");
stmt.setString(1, UserID);
ResultSet rs=stmt.executeQuery();
while(rs.next())
{
alreadyExists=true;//Username is present and active
stmt=con.prepareStatement
("select userID from "+ loginAs +" where UserID = ? and Password= ? and status=1");
stmt.setString(1,UserID);
stmt.setString(2,Password);
rs=stmt.executeQuery();
//to check account status and password
while(rs.next())
{PassW=true;status=true;break;}
if (PassW==false)
{
status=true;
```

```
}
}
if (alreadyExists==false)
{
stmt=con.prepareStatement
("select UserID,Password from "+ loginAs +" where UserID = ? and status=0");
stmt.setString(1,UserID);
rs=stmt.executeQuery();
//to check account status and password
while(rs.next())
{status=false;alreadyExists=true;break;}
}
if(alreadyExists==false)
{
JOptionPane.showMessageDialog(null,"Account Does not exists ");
}
else if(alreadyExists==true && status==true && PassW==true)
{
if (loginAs.equals(" loginstudent "))
{
DashboardStudent DS= new DashboardStudent(UserID,loginAs);
DS.setVisible(true);
this.dispose();
}
else if (loginAs.equals(" loginteacher ")&&
    (UserID.equals("hod")||(UserID.equals("jakade"))) )
```

```
{
DashboardHod DH= new DashboardHod(UserID,loginAs);
DH.setVisible(true);
this.dispose(); }
else
{
DashboardTeacher DT= new DashboardTeacher(UserID,loginAs);
DT.setVisible(true);
this.dispose(); }
}
else if(alreadyExists==true && status==true && PassW==false)
{JOptionPane.showMessageDialog(null,"Incorrect Password !");}
else if(alreadyExists==true && status==false)
{JOptionPane.showMessageDialog(null,"Account is not ACTIVE ! ");}
con.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,
"Erorr 060.\nPlease enter complete details.");
}
}
// Variables declaration - do not modify
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
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.JPanel jPanel1;
    private javax.swing.JPasswordField jPasswordField1;
    private javax.swing.JRadioButton jRadioButton1;
    private javax.swing.JRadioButton jRadioButton2;
    private javax.swing.JTextField jTextField1;
    // End of variables declaration
}
```

7. CONCLUSION

Technology is an important part of every day's life. Our project is aims to deliver daily notices, and result of students to respective student and all the teachers at tip of their fingers.

Every participant in this program need to register himself first by creating a account. Creating a account comprises of filling essential details about self. Both teachers and student will create account which would be forwarded to HOD for verification and activation of account along with activation of account Roll no. and Teacher ID would be assigned to student and teachers respectively automatically. Each username of teacher or student will be unique. On dashboard of every participant different option will be available along with their provided details.

Teachers can view result of all the students by applying different filters such as 'year of study' or PRN of student. Student can view their respective results only. Only teachers can post/ delete notices from the common noticeboard which will be visible to all the participants.All teachers can view complete data of student. Various filters such as 'year of study', 'passing year', or 'PRN' are available to filter out the students. HOD can view complete details about students and also teachers. Data of 'graduated student' also remains in the same system which in future can be retrieved by year of passing(batch) or PRN number of student. Data analysis of each student can be obtained for each semester.

Hence project helps in administration and management of college by aiding in functionalities like result display and noticeboard.

8. REFERENCES

- **Online References**

1. www.oracle.com
2. www.stackoverflow.com
3. Java SE Technical documentation

- **Books Reference**

1. Core JAVA by- Cay S. Hortsmann
2. Using SQLite by- Jay Kreibich,
3. SQL Server – Black Book by- Dalton Patrik
4. Managing and Using MySQL by- Reese G., Yarger R.
5. Java Swing by- By Brian Cole, Robert Eckstein..