KNOWLEDGE INSTITUTE OF TECHNOLOGY

(An Autonomous Institution)

Department of Computer Science and Engineering

Mini Project Report

CS3381 – Object Oriented Programming Lab

Academic Year	2023-2024
Year/Sem/Section	II / III / B
Project Title	COMPANY INFORMATION TRACKING SYSTEM (CITS)
Team Members	1. MADHIHA NISHAD Z [611222104083] 2. MADHUMIDHA R [611222104084] 3. MADHUMITHA K [611222104085] 4. MADUSUDANAN V [611222104086] 5. MAHESWARAN K [611222104087]
Staff Name	Mr.R.Sivaguru, AP/CSE
Signature	

Table of Contents

ABSTRACT

CITS is an effective HR tool for maintaining employee and company information. CITS helps you manage your employees more than a data storage program. CITS offers a wide range of reports to give you exactly the information you need. The user can view attendance information of the employee or can find the number of days a particular employee worked. CITS gives you useful information on different report categories. It also provides company name and address under the company profile module to show case the company details to the new user . CITS also provide the list of various department and the corresponding department id's under the department module. CITS gives you powerful information like income of the company and its profits , losses and also expenses in the module called finance which pay way to easy functioning of the company environment . CITS allow access to list of project available and also allows user to assign project to the respective employee and to remove project from the program and also allows you to add new project that need to be completed under a precise module called project. If the user one who accessing the CITS is the new user then the CITS allows the user to register by asking the user details such as email id, password for the id entered, full name of the respective user, address of the user and the user's contact number. In case if the registered user logging in then the system allows him to access all other module without any interference. Before adding employees, it's a good idea to define departments and tasks. You will also need to set up your company's benefits and ratings before adding them to your company profile. When you create a new category, such as an additional department or project, it is immediately selectable on every employee screen. CITS offers an intuitive and user-friendly interface that simplifies data entry and retrieval and reduces the learning curve for employees. Effortlessly search and filter data using powerful search algorithms and generate in-depth reports for strategic analysis. Secure sensitive business data with encryption and access control features. In general, CITS consolidates all company-related information, including financial information, department information, employee information, attendance documentations and project information, into one easily accessible repository. This CITS not only provide company details to the new user but also provide information about the company happenings to the employee to create an effective and coordinated functioning of the company which ultimately contribute to the development of the company in all instance and supports progress.

1.INTRODUCTION

In today's dynamic business landscape, managing and tracking company information efficiently is paramount for success. Introducing the Company Information Tracking System (CITS), a robust and comprehensive solution designed to streamline the organization, retrieval, and analysis of crucial data that drives informed decision-making.

CITS is a state-of-the-art platform meticulously crafted to meet the evolving needs of modern enterprises. With an intuitive user interface, it offers a user-friendly experience, ensuring seamless navigation for both novice and experienced users. The system is adept at handling diverse datasets, including financial records, employee information, client details, and more, providing a centralized repository for all vital company information.

One of CITS's key strengths lies in its ability to enhance data accessibility. By consolidating scattered data sources into a unified platform, CITS eliminates silos and promotes a collaborative work environment. This accessibility fosters cross-functional communication, enabling different departments to share insights and work cohesively toward common organizational goals.

Utilizing encryption protocols and access controls, the system ensures that sensitive information is protected from unauthorized access. Regular security audits and updates further fortify CITS against potential threats, providing peace of mind for businesses entrusting their valuable data to the system.

CITS is not a one-size-fits-all solution; it is customizable to adapt to the unique needs of each business. Whether it's tailoring reports, defining user roles, or integrating with existing systems, the flexibility of CITS ensures that it aligns seamlessly with the specific requirements of the organization. This adaptability makes it an ideal solution for businesses of all sizes and industries.

CITS transforms raw data into meaningful information, turning it into a strategic asset for the organization.

Scalability is another hallmark of CITS, allowing it to grow alongside the business. As companies expand and their data needs evolve, CITS can effortlessly accommodate increased data volume and user load. This scalability future-proofs the investment in the system, ensuring its relevance and effectiveness over the long term.

In conclusion, the Company Information Tracking System is a game-changer in the realm of data management. With its user-friendly interface, robust security measures, efficiency gains, customization options, analytics capabilities, and scalability, CITS stands as a comprehensive solution that empowers businesses to harness the full potential of their data. Embrace CITS and elevate your company's information tracking to new heights, laying the foundation for sustained growth and success in the digital age.

2.SYSTEM ANALYSIS :

It's easy to start in Java, you need to have following hardware and software requirements in order to create and run Java program in your system.

2.1 Hardware Requirements:

Computer:

A modern computer with a decent processor (e.g., Intel Core i5 or equivalent).

Laptop

Ample RAM (at least 8 GB is recommended for smoother development).

Storage:

Adequate free storage space for your development tools, databases, and project files.

Network:

Internet connectivity may be required for downloading software and updates.

2.2 Software Requirements:

Java Development Kit (JDK):

You'll need the Java Development Kit installed. Obtain it from the official Oracle website or consider using OpenJDK, depending on your preference.

Eclipse IDE:

Download and install Eclipse IDE. You can get it directly from the Eclipse website. Ensure you choose the appropriate version based on your operating system.

Database Server:

For a local development environment, consider using XAMPP or MySQL Server. XAMPP includes Apache (web server), MySQL (database server), and PHP. Download and set up XAMPP from the official website or install MySQL Server separately if preferred.

Database Connector/J:

To establish a connection between your Java application and the MySQL database, you'll need the MySQL Connector/J library. Download it from the official MySQL website.

Java Database Connectivity (JDBC):

Verify that your Eclipse setup includes JDBC drivers for MySQL. Usually, JDBC comes bundled with the JDK, but ensure you have the necessary drivers for your chosen database (e.g., MySQL) for seamless connectivity within Eclipse.

Remember, configuring these components ensures a smooth development environment within Eclipse, enabling you to develop Java applications with robust database connectivity and development capabilities.

3.MODULES

3.1 TITLE OF THE MODULES:

- Attendance
- Company profile
- Department
- Finance
- Project
- User module

3.2 DESCRIPTION OF MODULES:

ATTENDANCE:

- This module is very dominant among all modules in the CITS. This module is designed in such a way that employee attendance information is recorded and stored for future reference.
- An emp ID is the only one required for accessing or entering the module. This module in the CITS also provides the working time of a particular employee by detecting and calculating the logging time and log-out time very precisely.
- This particular module of the CITS is much better and more effective than a manual attendance system or record.
- The traditional or widely used manual attendance recording process may cause commotion and confusion, whereas in the CITS, this module would work effectively and provide a genuine and crystal-clear attendance record and time without any errors.

COMPANY PROFILE:

- This module is considered an informative module for the new user, where the user can view information about the company, such as the company name, address of the company, and departments in the company.
- This information might be very useful to the user who is searching for details about the company or seeking employment in the company. This module not

only provides a list of departments but also helps the user have an overview of various departments functioning in the company.

- This module of the CITS also plays a vital role in advertising or showcasing the company among new users and other related organisations.
- The user will also need to set up benefits and ratings before adding them to the user's company profile.
- This module might look simple, but it is a building block of the CITS and also important for expanding the company and contributing to its progress.

DEPARTMENT:

- This module of the CITS contains all the information about the various departments and the list of departments available in the company.
- CITS provides an intuitive, user-friendly interface that simplifies data entry. This module not only provides a list of departments but also provides the corresponding ID of the particular department.
- This module of the CITS is purely informative for both employees and general users. CITS provides useful information about various report categories.
- This list of departments and IDs is always useful for better understanding the organisation by the employee and also the new user.
- This module plays a particularly vital role in the development of the company and a better understanding of the various departments available in CITS. Before adding employees, it's a good idea to define departments and tasks.
- When you create a new category, such as an additional department or project, it is immediately selectable on the admin's screen.
- This department module is the different module among all modules of the CITS, and it is very important and game-changing.

FINANCE:

- The module called finance is the most important and dominant module among all modules of the CITS. This module governs all the important information about the company.

- This module contains all the important and sensitive information, such as income, profit, losses, and expenses of the company.
- This module can be strictly accessed by employees and admin. The new users are strictly prohibited from using this module because only employees of the company can access it. This module contains information that is not safe to share outside the company.
- This is the only reason for not allowing the new user to access this module. This module of the CITS contains the income of the company, which is the turnover of the company.
- This module also contains information about the profit gained by the company from various tasks and projects assigned and completed by the various departments and various teams of the company.
- It also contains details about the losses faced by the company due to the misfunctioning of the project or the flaws at any given instance.
- CITS's finance module contains information about the expenses faced by the company or details about the investment important role in the progress of the company. The restriction of usage by the new user is a very important and notable feature of this module.

PROJECT:

- This module of the CITS contains information about the projects that need to be assigned to the employee, and the employee can also add new projects that need to be added to the list of projects to be completed.
- This module also contains the name of the project and the corresponding ID of the project.
- This module of the CITS is very precise because it contains all the details about the project, such as the expenses that need to be invested in the particular project, the losses faced by the company due to the project, and the turn-over gained due to the particular project alone.
- This module of the CITS is only accessed by the employee, and the new user cannot access it due to security purposes.

- It is very important to prohibit the new user from using this module because there is a greater chance of misusing or stealing information about the project, which may cause a huge loss to the company.
- So, this module is perfectly designed to make the company benefit from the specification of the module and progress in all directions.

USER MODULE:

- This module of the CITS is the basic module where an already registered employee can login or a new user can register and proceed to login
- If the user who is accessing the CITS is a new user, then this module of the CITS allows the user to register by asking for user details such as email ID, password for the id entered, full name of the respective user, address of the user, and the user's contact number.
- If the registered user logs in, then the system allows him to access all other modules without any interference by asking for particulars such as the role of the user in the company and the user's department ID and project ID.
- CITS offers an intuitive and user-friendly interface that simplifies data entry and retrieval and reduces the learning curve for employees.
- Effortlessly search and filter data using powerful search algorithms, and generate in-depth reports for strategic analysis. Secure sensitive business data with encryption and access control features.
- In general, CITS consolidates all company-related information, including financial information, department information, employee information, attendance documentation, and project information, into one easily accessible repository.
- This CITS not only provides company details to the new user but also provides information about company happenings to the employee to create an effective and coordinated functioning of the company, which ultimately contributes to the development of the company in all instances and supports progress.

4.CODING

CODES FOR TESTMETHOD.JAVA

```
package com.cits.Test;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.Scanner;


import com.cits.model.UserModule;

import com.cits.service.Userfunctions;

import com.dbconnection.DBconnection;


public class TestMethod {

	public static void main(String[] args) {

		Userfunctions userFunction = new Userfunctions();

		Scanner scan = new Scanner(System.in);

		do {

			System.out.println("Welcome to Company Information Tracking
Service:");

			System.out.println("1.Login");

			System.out.println("2.Register");

			System.out.println("3.Company profile");

			System.out.println("4.Exit");
```

```
        int value =scan.nextInt();


switch (value) {
case 1:
        System.out.println("Username :");
        String username =scan.next();
        System.out.println("Password :");
        String passward = scan.next();
        String temp ="admin";
        if(username.equals(temp)) {
                try {
                        userFunction.getAdmin(username, passward);
                } catch (SQLException e) {
                        e.printStackTrace();
                }
        }
        try {

                userFunction.getLogin(username, passward);
        } catch (SQLException e) {

                System.out.println(e.getMessage());
        }
```

```
                    break;
                case 2:
                    System.out.println("Emp id");
                    int emp_id = scan.nextInt();
                    System.out.println("Emp name");
                    String emp_name = scan.next();
                    System.out.println("Email");
                    String email = scan.next();
                    System.out.println("Password");
                    String password = scan.next();
                    System.out.println("Address");
                    String address = scan.next();
                    System.out.println("Contact Number");
                    long contact_number = scan.nextLong();
//                  System.out.println("Role");
                    String role = "Null";
//                  System.out.println("Department id");
                    int dept_id = 0;
//                  System.out.println("Project Id");
                    int pro_id = 0;
                    UserModule user = new UserModule(emp_id, emp_name,
email, password, address, contact_number, role, dept_id, pro_id,0);
                    try {
                        userFunction.getRegister(user);
```

```
                } catch (SQLException e) {
                        e.printStackTrace();
                }
                break;
            case 3:
                System.out.println("company name = abc company");
                System.out.println("address      = salem,kakapalayam");

            case 4:
                System.out.println("Thank You!");
                System.exit(1);
                break;

            default:
                break;
            }
            } while (true);

        }
}
```

This Java code defines a console-based application for managing company information. It includes functionalities for user login, registration, and displaying company details. The program utilizes JDBC for database connectivity and employs a **Userfunctions** class for handling user-related operations. Upon execution, it presents a menu with options to login, register, view company profile, or exit. User input via the console determines the flow of the application, invoking corresponding

methods to handle login, registration, or displaying company information. The code operates within a perpetual loop until manually terminated, offering a continuous interactive experience for users managing company-related tasks through the console interface.

CODES FOR USER FUNCTION

```
package com.cits.service;

import java.sql.Connection;

import java.sql.Date;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.time.LocalDateTime;

import java.util.Scanner;


import com.cits.model.Attendance;

import com.cits.model.Project;

import com.cits.model.UserModule;

import com.dbconnection.DBconnection;


public class Userfunctions {

	DBconnection dbc = new DBconnection();

	Scanner scan = new Scanner(System.in);


	public UserModule getLogin(String username, String passward) throws
SQLException {

		Connection con= dbc.getConnection();

		PreparedStatement statement = con.prepareStatement("select * from
user where emp_name=? and password=? and valid=1");

		statement.setString(1, username);

		statement.setString(2, passward);
```

```
                ResultSet rs = statement.executeQuery();
                if(rs.next()) {
                        System.out.println("Login Successful!\nWelcome User");
                        UserModule user = new UserModule(rs.getInt(1),rs.getString(2),
rs.getString(3));
                        //System.out.println(user.toString());
                        getAttendance(user);
                        display();
//                      PreparedStatement statement2 = con.prepareStatement("insert
into attendance values(?,?,?)");
//                      statement2.setInt(1, rs.getInt(1));
//                      statement2.setString(2, "p");
//                      statement2.setObject(3, LocalDateTime.now());
//                      System.out.println("choose the option");
                }
                else {
                        System.out.println("Login Failed");
                }

                return null;

        }
public UserModule  display() throws SQLException {
        do {
        System.out.println("\n\nEnter the choice:");
        System.out.println("1.Profile");
        System.out.println("2.View attendance");
        System.out.println("3.View project details");
        System.out.println("4.Apply leave");
        System.out.println("5.Remarks/feedback");
```

```
        System.out.println("6.Exit");
        Connection con5= dbc.getConnection();
        int val = scan.nextInt();
//      System.out.println("Enter the Emp ID:");
//      int emp_id =scan.nextInt();
        switch(val) {
        case 1:
                System.out.println("Enter the Emp ID:");
                int emp_id =scan.nextInt();
                PreparedStatement statement23 = con5.prepareStatement("select *
from user where emp_id=?");
                statement23.setInt(1,emp_id);
                ResultSet r = statement23.executeQuery();
                System.out.println("YOUR PROFILE : \n");
                if(r.next()) {
                UserModule user = new UserModule(r.getInt(1),r.getString(2),
r.getString(3),r.getString(4),r.getString(5),r.getLong(6),r.getString(7),r.getInt(8),r.getI
nt(9),r.getInt(10));
                System.out.println(user.tostr());
        }
                break;
        case 2:
                System.out.println("Enter the Emp ID:");
                emp_id = scan.nextInt();
                System.out.println("Emp ID:"+emp_id);
                PreparedStatement statement32 = con5.prepareStatement("SELECT *
FROM attendance WHERE emp_id = ?");
                statement32.setInt(1, emp_id);
                ResultSet r1 = statement32.executeQuery();
                System.out.println("ATTENDANCE DETAILS :");
                while (r1.next()) {
```

```
                    Attendance attendance = new Attendance(
                        r1.getInt("emp_id"),
                        r1.getString("check_column"),
                        r1.getDate("Valid")
                        // Add other columns here similarly based on your table structure
                    );
                    System.out.println(attendance.toStr());
                }
                break;
            case 4:
                System.out.println("Apply for leave.\nyes/no:");
                String ch =scan.next();
                switch(ch) {
                case "yes":
                        System.out.println("Enter Emp_id:");
                        String ch1 =scan.next();
                        System.out.println("No.of.days:");
                        String ch2 =scan.next();
                        System.out.println("Date[date/mon/year-date/mon/year]:");
                        String ch3 =scan.next();
                        scan.nextLine();
                        System.out.println("Reason for applying leave");
                        String ch4 =scan.nextLine();
                        System.out.println("\nEmp_id = "+ch1+ "\nNo.of.days =
"+ch2+"\nDate/s = "+ ch3+"\nReason = "+ch4);
                        PreparedStatement statement16 =
con5.prepareStatement("INSERT INTO leave_table (emp_id, days, date, reason)
VALUES (?, ?, ?, ?)");
                        statement16.setInt(1, Integer.parseInt(ch1));
                        statement16.setInt(2, Integer.parseInt(ch2));
```

```
                    statement16.setString(3, ch3);
                    statement16.setString(4, ch4);
                    int Inserted = statement16.executeUpdate();
                    if(Inserted > 0) {
                        System.out.println("\nSuccessfully applied");


                    }
//                  else {
//                      System.out.println("failed to apply");
//                  }
                    break;
                case "no":
                    System.out.println("Thank you");;
            }
                break;
            case 3:
                System.out.println("Enter the Emp ID:");
                emp_id = scan.nextInt();
                PreparedStatement statement22 = con5.prepareStatement("SELECT
pro_id FROM user WHERE emp_id = ?");
                statement22.setInt(1, emp_id);
                ResultSet r3 = statement22.executeQuery();
                int pro_id = 0;
                if (r3.next()) {
                   pro_id = r3.getInt("pro_id");
                }
                System.out.println("PROJECT DETAILS : \n");
                PreparedStatement statementProject =
con5.prepareStatement("SELECT * FROM project WHERE pro_id = ?");
                statementProject.setInt(1, pro_id);
```

```
            ResultSet projectResult = statementProject.executeQuery();
            while (projectResult.next()) {
                    Project project = new Project(
                projectResult.getInt("pro_id"),
                projectResult.getString("pro_name"),
                projectResult.getString("start_date"),
                projectResult.getString("end_date"),
                projectResult.getDouble("income"),
                projectResult.getDouble("expense")
               );
               System.out.println(project.toString());
            }
            break;

case 5:
            scan.nextLine();
            System.out.println("Add your feedback here:");
            String ch5 =scan.nextLine();
            System.out.println("Thank you for your feedback");
            break;
case 6:
            System.out.println("Thank You!");
            System.exit(1);
            break;

default:
            break;
}
}while(true);
```

```
}
public UserModule getAttendance(UserModule user) {

		Connection con1= dbc.getConnection();
		PreparedStatement statement2;
		try {
			statement2 = con1.prepareStatement("insert into attendance
values(?,?,?)");
			statement2.setInt(1, user.getEmp_id());
			statement2.setString(2, "present");
			statement2.setObject(3, LocalDateTime.now());
			statement2.execute();
		} catch (SQLException e) {
			// TODO Auto-generated catch block
			e.printStackTrace();

		}
		return user;
}
```

This Java class, `Userfunctions`, facilitates various functionalities related to user interaction and data retrieval from a database. It encompasses methods to manage user login, display user profiles, attendance tracking, leave applications, project details viewing, and feedback submission.

1. getLogin(String username, String password): Validates user login credentials against a database. If successful, it displays a welcome message, fetches user data, records attendance, and presents a menu to interact further.

2. display(): Provides a menu-driven interface allowing users to view their profile, attendance details, project information, apply for leave, submit feedback, or exit the application. Each option corresponds to specific database queries and user interactions.

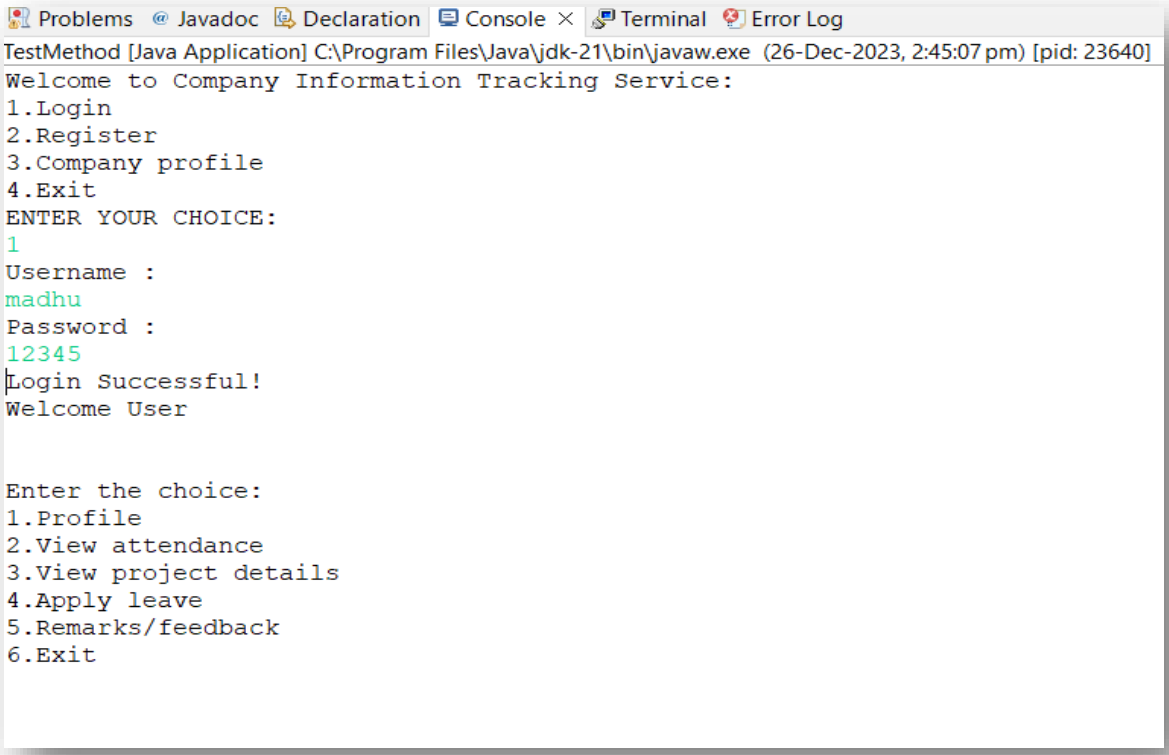
3. getAttendance(UserModule user): Records user attendance by inserting data into the database with the current timestamp when called.

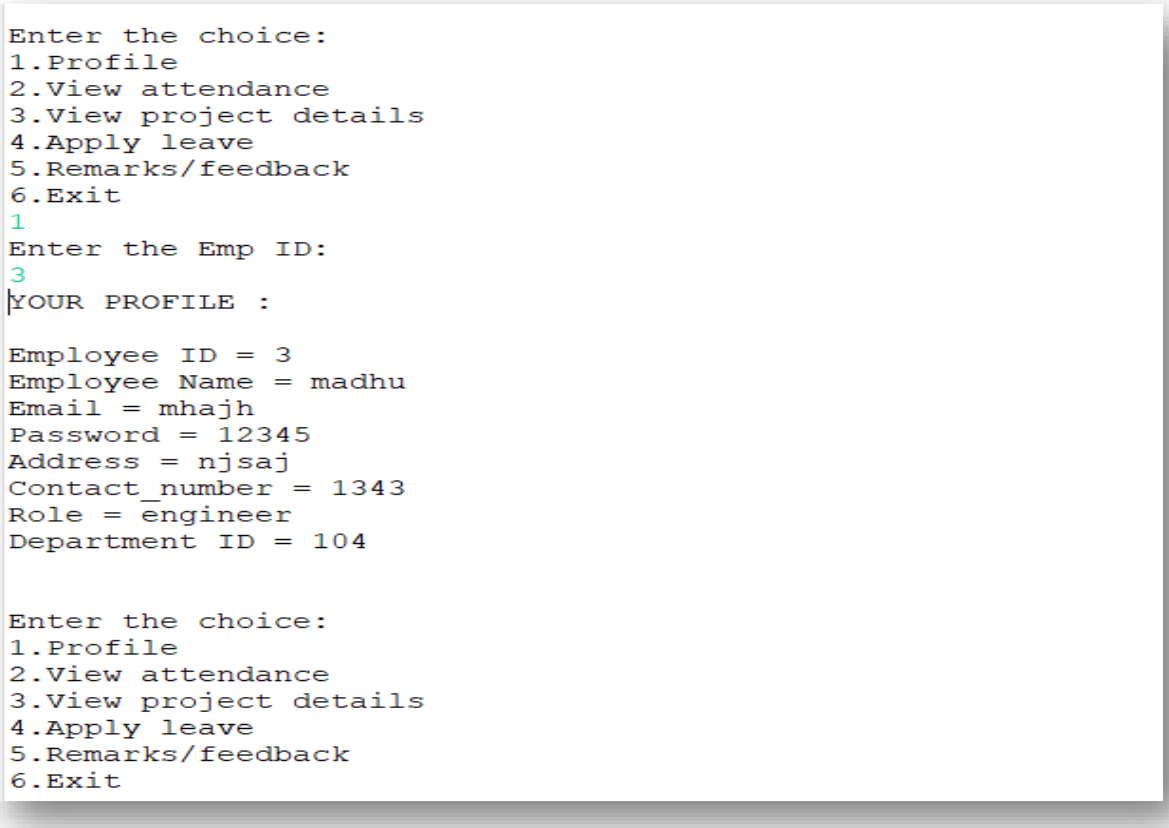
The class establishes database connections using the `DBconnection` class, executes SQL queries via prepared statements to retrieve or modify data, and handles user inputs via the `Scanner` class.

The `display()` method's menu-driven approach ensures user-friendly navigation, allowing individuals to access and interact with various aspects of their information and functionalities related to the company.

However, there are areas for potential enhancement, such as improved error handling, validation of user inputs, and better organization of methods for increased modularity and readability. Additionally, comments or method documentation could aid in understanding the code's functionality.

5.SCREENSHOTS





```
Enter the choice:
1.Profile
2.View attendance
3.View project details
4.Apply leave
5.Remarks/feedback
6.Exit
3
Enter the Emp ID:
3
PROJECT DETAILS :

Project ID =2
Project Name =ECE
Start_date =2001-01-20
End_date = 2001-05-20
Income = 2000000.0
Expense = 400000.0
```

```
Enter the choice:
1.Profile
2.View attendance
3.View project details
4.Apply leave
5.Remarks/feedback
6.Exit
2
Enter the Emp ID:
3
Emp ID:3
ATTENDANCE DETAILS :

check_column=present
login_date=2023-12-18

check_column=present
login_date=2023-12-18

check_column=present
login_date=2023-12-18

check_column=present
login_date=2023-12-18

check_column=present
login_date=2023-12-18

check_column=present
login_date=2023-12-18
```

```
Enter the choice:
1.Profile
2.View attendance
3.View project details
4.Apply leave
5.Remarks/feedback
6.Exit
4
Apply for leave.
yes/no:
yes
Enter Emp_id:
3
No.of.days:
1
Date[date/mon/year-date/mon/year]:
23/02/2005
Reason for applying leave
sick leave

Emp_id = 3
No.of.days = 1
Date/s = 23/02/2005
Reason = sick leave

Successfully applied
```

```
Welcome to Company Information Tracking Service:
1.Login
2.Register
3.Company profile
4.Exit
ENTER YOUR CHOICE:
1
Username :
admin
Password :
admin123
Welcome Admin!
1.List pending users
2.Approve pending users
3.Department
4.Assign Roles
5.Project
6.Finance
7.Company profile
8.exit
3
A.List Departments
B.Add Department
C.Assign Department
D.Delete Department
E.Exit
```

6.CONCLUSION

In conclusion, the Company Information Tracking System proves indispensable in enhancing organizational efficiency and decision-making. By centralizing data, streamlining processes, and providing real-time insights, it empowers businesses to navigate dynamic landscapes effectively. This system not only ensures accurate information retrieval but also facilitates strategic planning and resource optimization. With heightened data security and accessibility, it fosters a responsive and informed corporate environment. Overall, the implementation of this system is paramount for modern enterprises seeking a competitive edge through streamlined data management and actionable insights.

7.REFERENCE

YOUTUBE LINKS

- https://youtu.be/Gwwfetb5CJs?si=Iyjtd22wpyIRKQJQ
- https://youtu.be/rGysLrq9_QI?si=XOd-ZOm63vkyAifE
- https://www.youtube.com/watch?v=rpB1WWqZ6HQ
- https://www.youtube.com/watch?v=HmF3yQ_nj5Q

WEBSITE LINK

- https://www.seminarsonly.com/Engineering-Projects/Java/Company_Information_Tracking_System.php
- https://1000projects.org/company-information-tracking-system-cits-b-tech-cse-java-project-report.html
- https://www.scribd.com/document/329779762/Synopsis-yo
- https://freeprojectsforall.com/java-projects-company-information-tracking-system-2/