**JAVA MINI PROJECT**

**GRIEVANCES HANDLING SYSTEM**

Submitted by:

TRYPHENA R (312323205236)

Of

BACHELOR OF TECHNOLOGY (B.Tech)

In

INFORMATION TECHNOLOGY



St. JOSEPH’S COLLEGE OF ENGINEERING

(An Autonomous Institution)

St. JOSEPH’S GROUP OF INSTITUTIONS

OMR, Chennai – 600119

**ABSTRACT:**

The **Grievance Handling System** is a Java-based application designed to streamline the process of registering, tracking, and resolving complaints within an organization. Users can easily submit grievances and check the status of their complaints, while administrators can view and update complaint statuses, ensuring an organized resolution process.

The main goal is to provide a transparent and efficient way to manage complaints, reducing paperwork and improving communication. This small-scale system is ideal for institutions, companies, and communities, helping them maintain a record of complaints and their resolutions in a simplified manner.

**INTRODUCTION:**

The **Grievance Handling System** is designed to provide an efficient solution for managing complaints in organizations, institutions, or any setting where user concerns need to be addressed. It allows users to easily register complaints and track their status, while administrators can review and resolve issues. This system streamlines the grievance process by offering a user-friendly interface, ensuring both users and administrators can navigate and use it effortlessly. It fosters a transparent communication channel between users and administrators, enabling timely resolution of complaints.

The system also offers scalability to accommodate growing user bases and increasing complaints, while ensuring no grievance is overlooked. By organizing and tracking complaints, the system helps identify recurring issues, providing valuable insights for continuous improvement. This solution is reliable for both small and large organizations, ensuring a fair and transparent grievance resolution process that enhances overall user satisfaction.

**OBJECTIVE:**

The primary objectives of the Grievance Handling System are:

1. To provide users with an easy-to-use platform to file complaints.
2. To allow users to view the status of their complaints in real-time.
3. To offer administrators a tool for reviewing complaints and updating their statuses.
4. To ensure that the grievance process is streamlined and transparent.
5. To enhance communication between users and the administration, ensuring quick resolutions.
6. To provide an effective solution for managing and resolving grievances, reducing delays and confusion.

**SCOPE OF THIS PROJECT:**

1. Enabling users to register complaints with relevant details and track their status.

2. Allowing administrators to view all complaints and update their resolution status.

3. Automating the complaint management process to improve efficiency and transparency.

4. Applicable for institutions, companies, and communities to manage and resolve grievances effectively.

**PROJECT DESCRIPTION:**

The Grievance Handling System is a user-friendly application designed to allow users to register complaints and track their status, while also enabling administrators to manage and resolve these complaints. This system serves as an efficient platform for handling grievances, improving transparency, and providing timely responses.

Features:

1. User Functionality:
   * Login: Users can log in to the system.
   * Register Complaint: Users can register a complaint by providing their name and a description of the issue.
   * View Complaint Status: Users can check the status of their registered complaints by entering the complaint ID.
2. Admin Functionality:
   * Login: Admins can log in using a secured username and password.
   * View Complaints: Admins can view a list of all registered complaints.
   * Update Complaint Status: Admins can update the status of a complaint (e.g., "Resolved", "In Progress", etc.).
3. Complaint Tracking: Each complaint is assigned a unique ID, making it easier for both users and admins to track its progress.
4. Status Management: Initially, complaints have a "Pending" status, and the admin has the ability to update it as necessary.
5. Simple User Interface: The system uses a simple command-line interface (CLI), making it easy to interact with the application.

Technologies Used:

* Java for core logic and implementation
* Scanner for taking user input
* List to store and manage complaints

Purpose:

The project aims to automate the process of complaint registration and status management in organizations, providing both users and admins with a convenient and efficient system for grievance handling.

**SOURCE CODE:**

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

class Complaint {

int id;

String userName;

String description;

String status;

Complaint(int id, String userName, String description) {

this.id = id;

this.userName = userName;

this.description = description;

this.status = "Pending";

}

@Override

public String toString() {

return "Complaint ID: " + id + ", User: " + userName + ", Status: " + status;

}

}

public class GrievanceHandlingSystem {

private static List<Complaint> complaints = new ArrayList<>();

private static int idCounter = 1;

private static final String ADMIN\_USER = "admin";

private static final String ADMIN\_PASS = "password";

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

while (true) {

System.out.println("1. User Login 2. Admin Login 3. Exit");

System.out.print("Choose: ");

int choice = sc.nextInt();

sc.nextLine();

if (choice == 1) {

userMenu(sc);

} else if (choice == 2) {

if (adminLogin(sc)) {

adminMenu(sc);

} else {

System.out.println("Invalid admin credentials.");

}

} else if (choice == 3) {

System.out.println("Exiting...");

break;

} else {

System.out.println("Invalid choice. Try again.");

}

}

sc.close();

}

private static boolean adminLogin(Scanner sc) {

System.out.print("Admin Username: ");

String username = sc.nextLine();

System.out.print("Admin Password: ");

String password = sc.nextLine();

return ADMIN\_USER.equals(username) && ADMIN\_PASS.equals(password);

}

private static void userMenu(Scanner sc) {

while (true) {

System.out.println("1. Register Complaint 2. View Status 3. Back");

System.out.print("Choose: ");

int choice = sc.nextInt();

sc.nextLine(); // Consume newline

if (choice == 1) {

registerComplaint(sc);

} else if (choice == 2) {

viewStatus(sc);

} else if (choice == 3) {

break;

} else {

System.out.println("Invalid choice. Try again.");

}

}

}

private static void adminMenu(Scanner sc) {

while (true) {

System.out.println("1. View Complaints 2. Update Status 3. Back");

System.out.print("Choose: ");

int choice = sc.nextInt();

sc.nextLine(); // Consume newline

if (choice == 1) {

viewAllComplaints();

} else if (choice == 2) {

updateStatus(sc);

} else if (choice == 3) {

break;

} else {

System.out.println("Invalid choice. Try again.");

}

}

}

private static void registerComplaint(Scanner sc) {

System.out.print("Your Name: ");

String name = sc.nextLine();

System.out.print("Description: ");

String description = sc.nextLine();

Complaint complaint = new Complaint(idCounter++, name, description);

complaints.add(complaint);

System.out.println("Complaint registered with ID: " + complaint.id);

}

private static void viewStatus(Scanner sc) {

System.out.print("Enter Complaint ID: ");

int id = sc.nextInt();

sc.nextLine(); // Consume newline

for (Complaint complaint : complaints) {

if (complaint.id == id) {

System.out.println("Status of Complaint ID " + id + ": " + complaint.status);

return;

}

}

System.out.println("Complaint ID not found.");

}

private static void viewAllComplaints() {

if (complaints.isEmpty()) {

System.out.println("No complaints registered.");

} else {

for (Complaint complaint : complaints) {

System.out.println(complaint);

}

}

}

private static void updateStatus(Scanner sc) {

System.out.print("Enter Complaint ID to update: ");

int id = sc.nextInt();

sc.nextLine(); // Consume newline

for (Complaint complaint : complaints) {

if (complaint.id == id) {

System.out.print("Enter new status: ");

complaint.status = sc.nextLine();

System.out.println("Complaint status updated.");

return;

}

}

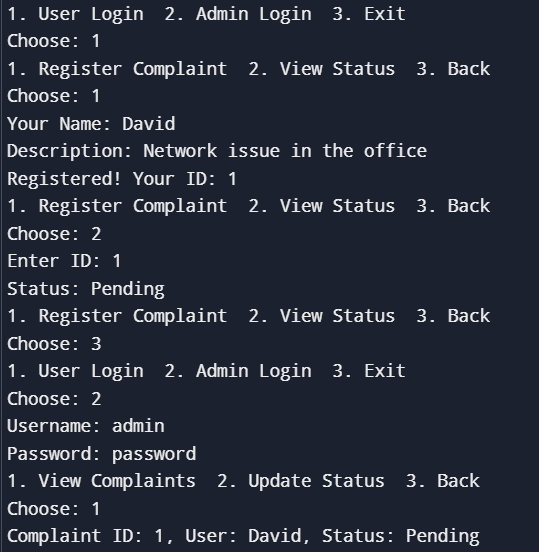
System.out.println("Complaint ID not found.");

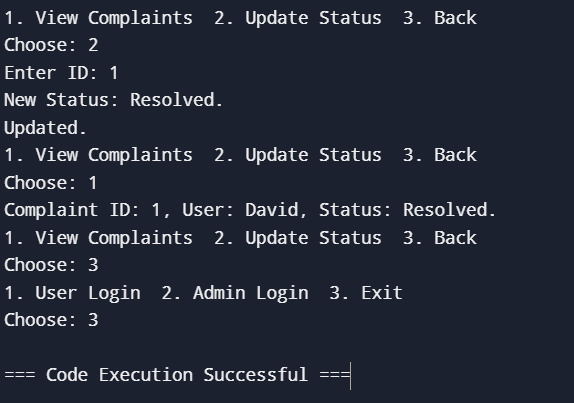
}

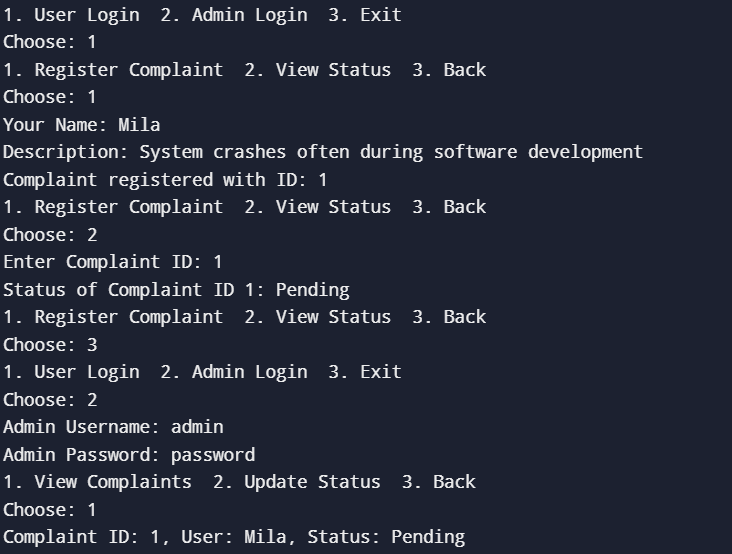
}

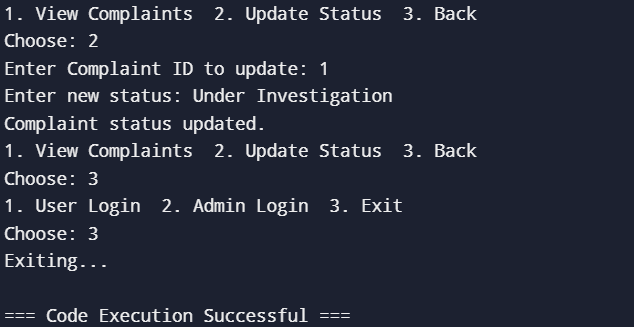
**OUTPUT:**

**SAMPLE 1:**

****

****

**SAMPLE 2:**

****

**CODE IMPLEMENATION:**

The Grievance Handling System allows users to register complaints and view their status, while an admin can manage complaints by viewing and updating their status.

Key Features:

1. Complaint ID: Each complaint is assigned a unique ID (automatically incremented) for tracking.
2. Complaint Status: Initially, every complaint has a "Pending" status, which the admin can change (e.g., to "Resolved" or "In Progress").
3. Admin Login: Admin access is secured with a username ("admin") and password ("password").
4. User Interaction:
   * Register Complaint: Users provide their name and complaint description to register a new complaint.
   * View Status: Users can check the status of their complaint by entering the complaint ID.
5. Admin Functions:
   * View Complaints: Admin can see all registered complaints.
   * Update Status: Admin can update the status of a complaint by entering the complaint ID.
6. Input Validation: The program ensures that the user's choice is valid and that the entered complaint ID exists for updates and status checks.
7. Exit Option: The program runs in a loop, providing menus for user and admin actions until the user chooses to exit.

The system is implemented using basic Java programming concepts like classes, methods, loops, and conditionals, enabling simple interaction between users and the admin for grievance handling.

**FUTURE ENHANCEMENTS:**

1. Data Persistence: Store complaints in a file or database instead of in memory, so data is saved between sessions.

2. User Authentication: Allow multiple users with unique credentials for secure access.

3. Email Notifications: Notify users when their complaint status is updated.

**CONCLUSION:**

The Grievance Handling System project offers a streamlined approach to registering and managing complaints, with distinct functionalities for users and administrators. Users can log complaints and track their status, while administrators are granted secure access to view and update complaint records. This console-based application highlights essential Java programming concepts, such as object-oriented design and role-based access control, while ensuring user-friendly interactions through a simple menu interface. Overall, the project effectively demonstrates how basic programming principles can be applied to solve real-world problems in complaint management.