

JAVA PROGRAMMING LAB

MINI PROJECT

(CA 3307)

TOPIC – CHARITY MANAGEMENT SYSTEM



DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

NIT JAMSHEDPUR

Name - Aman

Reg no - 2021PGCACA034

Course - MCA (3rd Sem)

Faculty-in-charge – Dr. Alekha Kumar Mishra

CHARITY MANAGEMENT SYSTEM

Overview

Charity management System is a project in JAVA where a candidate can give donations online on this portal. In this mini project there are two portals, first is admin portal: where admin can add organization and view donations by each users. Second is user portal : user can give donations to any organization as they want.

Code:

```
import java.util.*;

class CharityUsers {
    private String Name;
    private String userName;
    private String passCode;

    public String getName() {
        return this.Name;
    }
    public String getUserName() {
        return this.userName;
    }
    public String getPassCode() {
        return this.passCode;
    }
    CharityUsers(String N, String userName, String passCode) {
        this.Name = N;
        this.userName = userName;
        this.passCode = passCode;
    }
}

class CharityUserCollection {
    public static String userID;
    Scanner s = new Scanner(System.in);
    private static ArrayList<CharityUsers> users = new ArrayList<>();

    public void userAdd(CharityUsers user) {
        this.users.add(user);
        //System.out.println(this.users.get(0).getName());
    }

    public boolean Login(String userName,String passCode) {

        for(CharityUsers u : this.users) {
            if((u.getUserName().compareTo(userName)) == 0 && (u.getPassCode().compareTo(passCode))==0) {
                userID = userName;
                return true;
            }
        }
        return false;
    }
}
```

```

public CharityUsers findUserObject(String userName,String passCode) {

    for(CharityUsers u : this.users) {
        if((u.getUserName().compareTo(userName)) == 0 && (u.getPassCode().compareTo(passCode))==0) {

            return u;
        }
    }
    CharityUsers x = null;
    return x;
}

public void userView() {
    System.out.println("USERS LIST");
    int id=0;
    if(this.users.size()<1) {
        System.out.println("***NO RECORDS FOUND!***");
    }
    for(CharityUsers u : this.users) {
        System.out.println("ID : "+id++);
        System.out.println("NAME : "+u.getName());
        System.out.println("_____");
    }
}

public void regUser() {

    System.out.println("PROVIDE YOUR FULL NAME : ");
    String N = s.nextLine();
    System.out.println("PROVIDE YOUR USERNAME : ");
    String userName = s.nextLine();
    System.out.println("PROVIDE YOUR PASSCODE : ");
    String passCode = s.nextLine();
    for(CharityUsers checkUser : this.users) {
        if(checkUser.getUserName().compareTo(userName)==0) {
            System.out.println("USERNAME ALREADY TAKEN!");
            return;
        }
    }

    CharityUsers user = new CharityUsers(N,userName,passCode);
    this.userAdd(user);
    System.out.println("***ACCOUNT CREATION SUCCESSFULL!***");
}

class CharityDetails {
    public String OrganizationName;
    public String Location;
    public int donations;
    public ArrayList<String> donars;
    public CharityDetails(String name, String Location) {
        this.OrganizationName = name;
        this.Location = Location;
        this.donations = 0;
        this.donars = new ArrayList<>();
    }
    public void donate(int donation,String user) {
        this.donations = this.donations+donation;
        this.donars.add(user);
        System.out.println("***DONATION SUCCESSFULL!***");
    }
}

```

```

class CharityCollection {
    public static ArrayList<CharityDetails> charities = new ArrayList<>();

    Scanner s = new Scanner(System.in);
    public void donate(CharityUsers user) {
        if(this.charities.size()<1) {
            System.out.println("***NO organizationS TO DONATE***");
        } else {
            int oid=-1;
            while(oid!=0) {
                System.out.println("\tCHOOSE organization");
                System.out.println("\t_____");
                int id = 1;
                for(CharityDetails e : this.charities) {
                    System.out.println("\tID : "+id++);
                    System.out.println("\tNAME : "+e.OrganizationName);
                    System.out.println("\tLOCATION : "+e.Location);
                    System.out.println("\t_____");
                }
                System.out.println("\tCHOOSE ANY organization (PRESS 0 to EXIT) : ");
                oid = s.nextInt();
                if(oid ==0) {
                    break;
                }
                int tempId = oid;
                tempId--;
                System.out.println("PROVIDE DONATION AMOUNT : ");
                int don = s.nextInt();
                try {
                    charities.get(tempId).donate(don,user.getName());

                } catch(Exception e) {
                    System.out.println("***CHOOSE VALID ID***");
                }
            }
        }
    }

    public void charityAdd() {

        System.out.println("PROVIDE organization NAME : ");
        String OrganizationName = s.nextLine();
        System.out.println("PROVIDE LOCATION : ");
        String Location = s.nextLine();
        CharityDetails val = new CharityDetails(OrganizationName,Location);
        this.charities.add(val);
        System.out.println("***ADDED SUCCESSFULLY!***");
    }

    public void viewCharity() {
        if(this.charities.size()<1) {
            System.out.println("***NO RECORDS FOUND!***");
        } else {
            System.out.println("CURRENT LIST OF DONATIONS");
            System.out.println("_____");
            int id = 1;
            for(CharityDetails e : this.charities) {
                System.out.println("ID : "+id++);
                System.out.println("NAME : "+e.OrganizationName);
                System.out.println("LOCATION : "+e.Location);
            }
        }
    }
}

```

```
System.out.println("TOTAL DONATION AMOUNT : "+e.donations);
System.out.println("DONATIONS BY : "+e.donars);
```

```
S System.out.println("_____");
    }
    }
    }
}
```

```
class Aman_Project {
    public static Scanner s = new Scanner(System.in);
    static int f=0;
    public static void frontEnd() {
        CharityUserCollection user = new CharityUserCollection();
        CharityCollection charity = new CharityCollection();
        int opt = 0;

        while(opt!=4) {
            try {
                if(f==0) {
                    System.out.println("**first You have to add organization through admin login."
                        + "\n(admin user & pass is: 'admin' **");
                    f=1;
                }

                System.out.println("\n\n__WELCOME TO CHARITY MANAGEMENT SYSTEM__");
                System.out.println("1.USER LOGIN");
                System.out.println("2.ADMIN LOGIN");
                System.out.println("3.USER REGISTER");
                System.out.println("4.EXIT");
                System.out.print("PLEASE PROVIDE AN OPTION : ");
                opt = s.nextInt();

                if(opt==2) {
                    System.out.println("PROVIDE ADMIN USERCODE : ");
                    s.nextLine();
                    String userName = s.nextLine();
                    System.out.println("PROVIDE ADMIN PASSCODE : ");
                    String passCode = s.nextLine();
                    if(userName.compareTo("admin")==0 && passCode.compareTo("admin")==0) {
                        int spt=0;
                        while(spt!=4) {
                            System.out.println("\n__WELCOME ADMIN!__\n");

                            System.out.println("\t1.ADD organization ");
                            System.out.println("\t2.VIEW DONATIONS ");
                            System.out.println("\t3.VIEW USERS: ");
                            System.out.println("\t4.BACK: ");
                            System.out.print("\tCHOOSE ANY : ");
                            spt = s.nextInt();
                            if(spt==1) {
                                System.out.println("\t\tPROVIDE organization DETAILS : ");
                                charity.charityAdd();
                            }
                            if(spt==2) {
                                charity.viewCharity();
                            }
                            if(spt==3) {
                                user.userView();
                            }
                        }
                    }
                }
            }
        }
    }
}
```

```

    } else {
        System.out.print("\t**AUTHORIZATION FAILED** ");
    }

    if(opt == 1) {
        System.out.println("PROVIDE YOUR USERNAME : ");
        s.nextLine();
        String userName = s.nextLine();
        System.out.println("PROVIDE YOUR PASSCODE : ");
        String passCode = s.nextLine();

        if(user.Login(userName,passCode)) {
            int spt = 0;

            CharityUsers currentUser = user.findUserObject(userName,passCode);
            while(spt!=2) {
                System.out.println("\n__LOGIN SUCCESS__\n");
                System.out.println("\t1.DONATE ");
                System.out.println("\t2.EXIT: ");
                System.out.print("\tCHOOSE ANY : ");
                spt = s.nextInt();
                if(spt==1) {

                    charity.donate(currentUser);
                }
            }
        } else {
            System.out.println("**CREDENTIALS NOT RECOGNISED BY OUR INTERNAL
SYSTEM**");
        }
    }

    if(opt == 3) {
        user.regUser();
    }

    } catch (Exception e) {
        System.out.println("PLEASE PROVIDE VALID INPUT!");
        continue;
    }
}
System.out.print("THANK YOU FOR USING OUR SERVICE");
}

public static void main(String[] args) {
    frontEnd();
}
}

```