public class **Application**  
{

static User activeUser;

static boolean isJobWindowOpen;

public static User getActiveUser()

public static void setActiveUser(User user)

public static boolean login(String nid, String password)

public static void submitJobApplication(String name, int age, String nid, String location, String email, int phoneNumber, File cv)

}

public class **Party**

{

public static ArrayList<NLC> nlcMembers;

public static ArrayList<DLC> dlcMembers;

public static ArrayList<GM> gmMembers;

public static Manager manager;

public static FinancialManager financialManager;

public static Election dlcPromotionElection;

public static Election generalMemberPromotionElection;

private static ArrayList<PartyProject> onGoingPwp;

private static ArrayList<CommunityProject> onGoingCwp;

public static void addOnGoingProject(Project project)

public void createNewUser(String name, int age, String nid, String location, String email, int phoneNumber)

}

public interface **IProjectCreatable**

{

public void createProject(String projectTitle, String projectDescription, int expenditure, File attachment, String location, IProjectCreatable projectCreator);

}

public interface **IFundPayable**

{

public void donateFund(float amount);

}

public interface **IPromotable<T>**

{

public T promote();

}

public interface **IElectionCreatable**

{

public boolean createPromotionElection();

}

public class **JobApplication**

{

String name, nid, location, email, phoneNumber;

int age;

File cv;

}

public abstract class **Project**

{

String title, description, nlcFeedback, location;

int expenditure;

File file;

IProjectCreatable projectCreator;

GeneralMember appointee;

Report report;

public Project(String title, String description, String nlcFeedback, int expenditure, File attachment, String location)

public abstract Report getReport();

public abstract void setReport(Report report);

public abstract void setNlcFeedback(String feedback);

public abstract String getNlcFeedback();

public abstract void setAppointee(GeneralMember gm);

public abstract GeneralMember GetAppointee();

}

public class **PartyProject** extends Project

{

public void setNlcFeedback(String feedback)

public void setAppointee(GeneralMember gm)

public Report getReport();

public void setReport(Report report);

}

public class **CommunityProject** extends Project

{

DLC cwpFilter;

public void setNlcFeedback(String feedback)

public void setAppointee(GeneralMember gm)

public void setCwpFilter(DLC dlc)

public Report getReport();

public void setReport(Report report);

}

public class **Report**  
{  
 String message;

File file;

public Report(String message, File file)

public String GetMessage()

public File GetFile()  
}

public class **Donation**

{

float fund;

IFundPayable donator;

}

public class **Election**  
{  
 public IPromotable[] shortlistForPromotion;

public ArrayList<IElectionCreatable> voters;

public int[] votes;

public void createShortList(ArrayList<IPromotable> users)

public void castVote(int candidate, IElectionCreatable voter)

public void finaliseElection()

}

public class **ProjectApprovalSystem**

{  
 private Project project;

private int approvalCount, disapprovalCount;

private ArrayList<NLC> nlcSubmittedDecision;

public ProjectApproval(Project project)

public void approveProject()

public void disapproveProject()

public void forwardProjectToDlc()

public void removeProjectFromNlcList()  
}

public abstract class **User**

{

String name, NID, location, emailAddress;

int age, phoneNumber;

public User(String name, int age, String nid, String location, String email, int phoneNumber)

public void logout()

}

public class **NLC** extends User implements IProjectCreatable, IFundPayable, IElectionCreatable

{

private static ArrayList<ProjectApprovalSystem> financialApprovedPwp;

private static ArrayList<CommunityProject> dlcApprovedCwp;

private static ArrayList<ProjectApprovalSystem> financialApprovedCwp;

private ArrayList<Project> requestedProjectReport;

public void createProject(String projectTitle, String projectDescription, int expenditure, File attachment, IProjectCreatable projectCreator)

public static void addFinancialApprovedProject(Project project)

public static void removeFinancialApprovedProject(Project project)

public static void addDlcApprovedCwp(CommunityProject project)

public void requestFinancialApproval(CommunityProject projectForApproval)

public void projectImplementationApproval(ProjectApprovalSystem projectForApproval, boolean isApproved)

public void requestReport(Project project)

public void addRequestedProjectReport(Project project)

public void giveProjectFeedback(Project project, String message, int rating)

public void createPromotionElection();

public void donateFund(float amount)

}

public class **DLC** extends User implements IPromotable<NLC>, IFundPayable, IPromotionCreatable  
{

int overallRating, deliveredCwpCounter;

private static ArrayList<CommunityProject> dlcApprovalPendingCwpFromGm;

private static ArrayList<PartyProject> pwpToAssign;

private static ArrayList<CommunityProject> cwpToAssign;

public void assignProject(Project project, GeneralMember gm)

public static void addProjectToAssign(Project project)

public void updateOverallRating(int rating)

public NLC promote();

public static void addCwpForDlcApproval(CommunityProject project)

public void forwardCwpToNLC(CommunityProject cwp)

public boolean createPromotionElection();

public void donateFund(float amount)  
}

public class **GeneralMember** extends User implements IPromotable<DLC>, IProjectCreatable, IFundPayable

{

Project assignedProject;

int overallRating, currentProjectRating, completedProjectCount;

public void createProject(String projectTitle, String projectDescription, int expenditure, File attachment, IProjectCreatable projectCreator)

public void sendReport(String message, File attachment)

public void setCurrentProjectRating(int rating)

public void donateFund(float amount)

}

public class **Manager** extends User

{

private Hashtable<Project, Report> reportsRecievedFromGeneralMembers;

private Hashtable<Project, NLC> reportRequest;

private Hashtable<Project, int> nlcFeedbackForGeneralManager;

private ArrayList<JobApplication> jobApplications;

private ArrayList<JobApplication> interviewShortList;

private ArrayList<IPromotable> promotableUsers;

public void addProjectImplementationReport(Project project, Report report)

public void addReportRequest(Project project, NLC nlc)

public void forwardReportToNlc(Project project, NLC nlc)

public void processReceivedReports()

public void addNlcFeedback(Project project, int rating)

public void toggleJobWindow(boolean value)

public void addJobApplication(JobApplication application)

public void createInterviewShortList(ArrayList<JobApplication> applicants);

public void hireAsFinancialManager(JobApplication applicant);

public void sendJoiningLetter(String applicantEmailID);

public void addPromotionReport(IPromotable candidate)

public void processPromotableUsers()

public void sendPromotionNotice(String recipientEmail)

}

public class **FinancialManager** extends User

{

private static ArrayList<PartyProject> financialApprovalPendingPwp;

private static ArrayList<CommunityProject> financialApprovalPendingCwp;

private ArrayList<Donation> newDonations

private ArrayList<Donation> processedDonations;

private ArrayList<float> annualDonations;

private float partyFund;

public void addProjectForFinancialApproval(Project project)

public void modifyProjectExpenditure(Project project, float newExpenditure)

public void approveProject(Project project)

public void processDonation()

public void sendDonationNotices()

public void updateAnnualSheet()

}