**class ConjuntoUsuarios{**

private HashMap<String, Usuario> cjt;

**//CONSTRUCTORAS**

//Post:crea el conjunt i afegeix l’admin(usuari\_privilegiat)

**public ConjuntoUsuarios()**

**//CONSULTORAS**

//Pre: a cjt hi ha un usuari amb this.username = username si no retorna null.

//Post: retorna aquest usuari.

**public usuari\_estandard getUsuario(String username)**

**//MODIFICADORAS**

//Post: afegeix l’usuari user a cjt amb clau nom si ja n’hi havia un amb aquesta clau el substitueix.

**public boolean** addUsuario(usuari\_estandard user)

//Pre: a cjt hi ha un usuari amb username = nom si no retorna false..

//Post esborra aquest usuari i retorna true.

**public boolean** deleteUsuario(String username)

**public class usuari\_privilegiat extends usuari\_estandard{**

**//ATRIBUTOS**

**//CONSTRUCTORAS**

//Post : crea un usuari privilegiat amb username = user i password = pass

**public usuari\_privilegiat(String user, String pass)**

//crea un usuari privilegiat amb els atributs i les relacions de standard user.

**public usuari\_privilegiat(usuari\_estandard standard\_user)**

**//CONSULTORAS**

**//MODIFICADORAS**

//Post: modifica els atributs de l’usuari passat com a paràmetre amb els valors passats com a paràmetres o no modifica els que es passa null.

**public boolean** modificar\_usuari(String username, String pass, String nom, String sexe, Date naix, ConjuntoUsuarios cjt)

//Pre: hi ha un usuari a cjt\_Usuarios amb clau username sinó retorna false

//Post: esborra aquest usuari i retorna true.

**public boolean** borrar\_usuari(String username, ConjuntoUsuarios cjt)

//Post: transforma user en un usuari privilegiat

**public void** donar\_privilegis(usuari\_estandard ue, ConjuntoUsuarios cjt)

//Post: retorna el map amb tots els usuaris

**public Map<String, usuari\_estandard>**

**informacio\_usuaris(ConjuntoUsuarios cjt)**

**public class usuari\_estandard extends Usuario\_generico{**

**//ATRIBUTOS**

//CONSTRUCTORAS

//Post: crea un usuari buit

public usuari\_estandard()

//Post: crea un usuari amb username = user i password = pass.

public usuari\_estandard(String user, String pass)

//CONSULTORAS

//Post retorna la data de naixement

**public** Date getFechaN()

//MODIFICADORAS

//Post: canvia la data de naixement per fecha

**public void** setFechaN(Date fecha)

//Pre: existeix una relacio a Relacions amb clau = antic si no retorna false

//Post: modifica la relacio amb els atributs passats com a paràmetres i retorna true, no modifica els que es passa null.

public boolean modificar\_relacio(String antic, String nom, String descripcio)

//Post retorna totes les relacions de l’usuari.

**public** Map<String, TipoRelacion> getRelacions()

//Pre : oldPass és la contrasenya d’user sino retorna false

//Post: modifica user amb els paràmetres passats i retorna true.

**protected boolean** modificar\_usuari\_aux(usuari\_estandard user, String oldPass, String pass, String nom, String sexe, Date naix)

//Pre: oldPass és la contrasenya de l’usuari en execució

//Post: modifica l’usuari amb els paràmetres passats.

**public boolean** modificar\_usuari(String oldPass, String pass, String nom, String sexe, Date naix)

//Post: esborra l’usuari en execució de cjt

**public void** borrar\_usuari(ConjuntoUsuarios cjt)

**public class transformar\_data{**

**private static final** SimpleDateFormat formatter = **new** SimpleDateFormat(**"dd/MM/yyyy"**);

//Pre: s té el format SimpleDateFormat

//Post transforma s en una data

**public static** Date stringToDate(String s)

//Post transforma data en un string amb el format SimpleDateFormat

**public static** String dateToString(Date data)

//Pre: hi ha una entitat al graf amb nom = nom i classe = tipus.

//Post: retorna tots els elements directament relacionats amb aquest.

**public** Vector<Entidad> relacions\_directes(String nom, String tipus, Grafo graf)

**public class ctr\_usuari\_dom**

**private Scanner sc = new Scanner(System.*in*);**

**private ConjuntoUsuarios cjt;**

**private usuari\_estandard user;**

**private Grafo graf;**

//Post: this.cjt = cjt

**public void carregar\_cjt\_usuaris(ConjuntoUsuarios cjt)**

//Post: this graf = graf

**public void carregar\_graf(Grafo graf)**

//Pre no existeix cap usuari amb nom = username sinó retorna false.

//Post crea un usuari amb els atributs passats com a paràmetres i retorna true. El crea privilegiat si privilegiat = true.

**public boolean** crear\_usuari(String username, String pass, **boolean** privilegiat)

//Post crea una relació a l’usuari amb els atributs passats com a paràmetres si ja n’hi havia una amb el mateix nom la substitueix

**public void** afegir\_relacio(String nom, String path, String descripcio)

//Pre Existeix una relació de l’usuari actual amb this.nom = nom. Si no retorna false.

//Post Modifica la relació amb els paràmetres passats i retorna true.

**public boolean** modificar\_relacio(String nom, String nomNou, String descripcio)

//Pre Existeix una relació de l’usuari actual amb nom = nom. Si no retorna false.

//Post: esborra la relacio de l’usuari i retorna true.

**public boolean** esborrar\_relacio(String nom)

//Pre: existeix una relació a l’usuari actual amb nom = nom si no retorna null.

//Post: la retorna.

**public** TipoRelacion consultar\_relacio(String nom)

//Post: Retorna tots els usuaris creats amb els seus atributs(les relacions no)

**public** ArrayList<ArrayList<String>> informacio\_usuaris()

//Post: retorna totes les relacions de l’usuari amb els seus atributs

**public** ArrayList<ArrayList<String>> informacio\_relacions()

//Pre: existeix un usuari al conjunt amb el nom i la contrasenya passats com a paràmetres sinó retorna “0”.

//Post: S’actualitzen les variables del controlador de domini d’usuari i retorna la classe de l’usuari

**public** String loggin(String nom, String pass)

//Post: esborra l’usuari actual del conjunt

**public void** borrar\_usuari\_estandard()

//Pre: O bé l’usuari amb nom = username existeix al conjunt i esPrivilegiat és true, oldPass és null i user(l’atribut) és privilegiat o bé username és null, oldPass és la contrasenya de l’usuari actual i esPrivilegiat és false; sinó retorna false.

//Post: modifica l’usuari amb nom = username si esPrivilegiat es true o l’usuari actual si és false.

**public boolean** modificar\_usuari(String username, String oldPass, String pass, String nom, String sexe, Date data, **boolean** esPrivilegiat)

//Pre: user(l’atribut) és privilegiat i existeix un usuari al cjt amb nom = user\_borrar sinó retorna false.

//Post: l’esborra del cjt

**public boolean** borrar\_usuari(String user\_borrar)

//Pre: user(l’atribut) és privilegiat i existeix un usuari estàndard al cjt amb nom = username

/Post: el transforma en privilegiat

**public boolean** donar\_privilegis(String username)

//Pre: user(l’atribut) és privilegiat

//Post: afegeix una nova entitat amb nom=nom, de la classe tipus i amb etiqueta = etiq

**public void** afegir\_element(String nom, String tipus, String etiq)

//Pre: user(l’atribut) és privilegiat, primer i segon són entitats ja existents al graf i

tipus ={AP|CP|TP} segon la classe de les 2 entitats

//Post: afegeix la relació entre primer i segon al graf

**public void** afegir\_relacio\_graf(String primer, String segon, String tipus)

//Pre: user(l’atribut) és privilegiat i hi ha una entitat al graf amb nom = nom i de la classe tipus

//Post: l’esborra

**public void** esborrar\_element(String nom, String tipus)

//Pre: user(l’atribut) és privilegiat i existeix una relació al graf entre primer i segon i

tipus ={AP|CP|TP} segon la classe de les 2 entitats

//Post: esborra la relació(no esborra les entitats)

**public void** esborrar\_relacio\_graf(String primer, String segon, String tipus)

//Pre: hi ha una entitat al graf amb nom = nom i de la classe tipus.

//Post: retorna totes les entitats(i els seus atributs) relacionades amb aquest

**public** ArrayList<ArrayList<String>> relacions\_directes(String nom, String tipus)