Part 4

The document contains the link to the group video and the presentation slides use in the presentation.

The Link to the group video is:

https://www.youtube.com/watch?v=yqSy5RGeRGk&feature=youtu.be
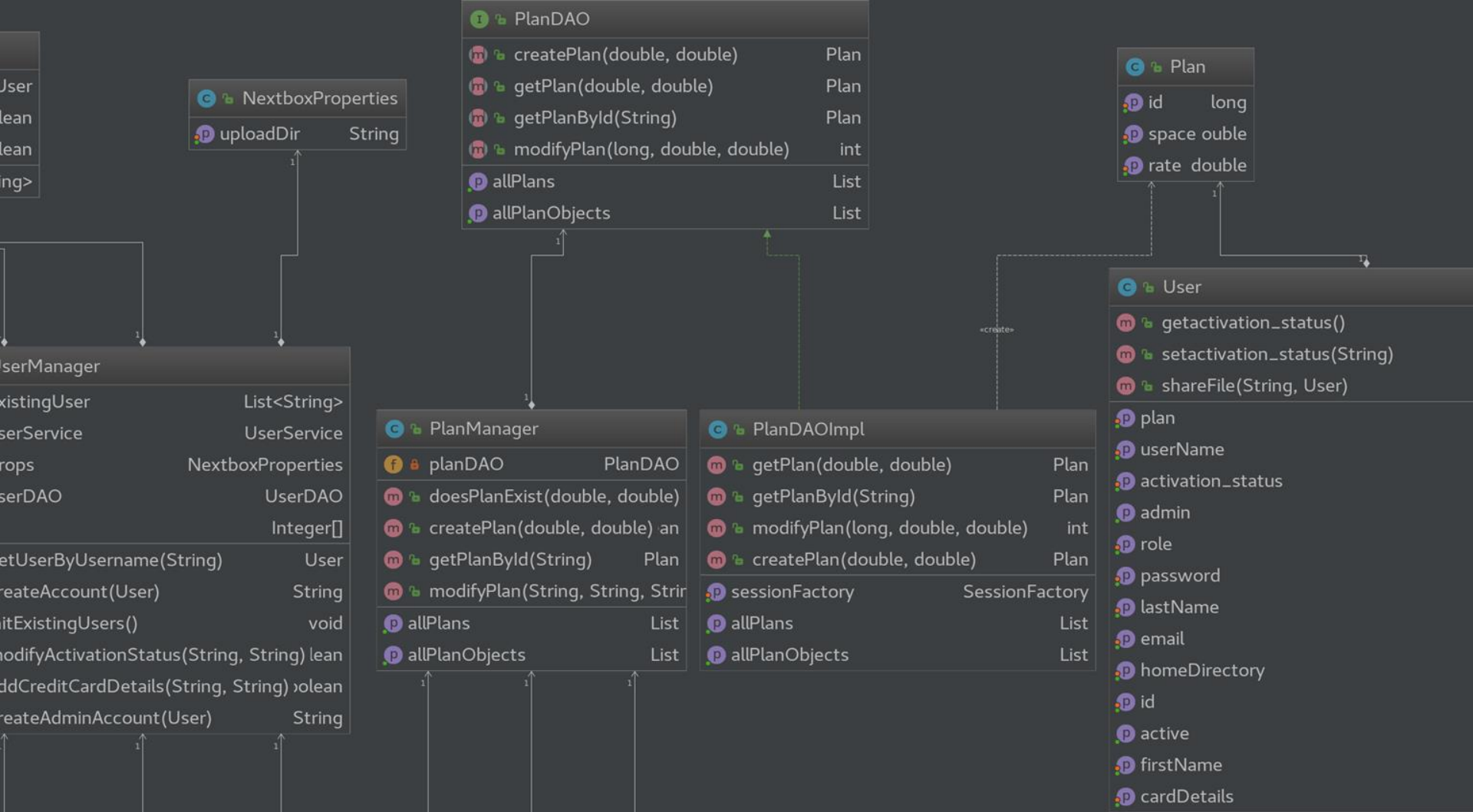
NextBox - Easily Manage Files
(Team 43)

Anirudh Tiwari
Daniel Milroy
Saurabh Sood
Sneha Parmar

Use cases

1. Create user account, create admin account: Anirudh

2. View photo and recursively delete directory: Daniel

3. Change activation status and view usage: Sneha

The Proxy Design Pattern

NextboxProperties
- uploadDir : String

PlanDAO (Interface)
- createPlan(double, double) : Plan
- getPlan(double, double) : Plan
- getPlanById(String) : Plan
- modifyPlan(long, double, double) : int
- allPlans : List
- allPlanObjects : List

Plan
- id : long
- space : ouble
- rate : double

UserManager
- existingUser : List<String>
- serService : UserService
- rops : NextboxProperties
- serDAO : UserDAO
- : Integer[]
- etUserByUsername(String) : User
- reateAccount(User) : String
- itExistingUsers() : void
- modifyActivationStatus(String, String) : lean
- ddCreditCardDetails(String, String) : oolean
- reateAdminAccount(User) : String

PlanManager
- planDAO : PlanDAO
- doesPlanExist(double, double)
- createPlan(double, double) an
- getPlanById(String) : Plan
- modifyPlan(String, String, Strir
- allPlans : List
- allPlanObjects : List

PlanDAOImpl
- getPlan(double, double) : Plan
- getPlanById(String) : Plan
- modifyPlan(long, double, double) : int
- createPlan(double, double) : Plan
- sessionFactory : SessionFactory
- allPlans : List
- allPlanObjects : List

«create»

User
- getactivation_status()
- setactivation_status(String)
- shareFile(String, User)
- plan
- userName
- activation_status
- admin
- role
- password
- lastName
- email
- homeDirectory
- id
- active
- firstName
- cardDetails

Use Case Diagrams

Admin Use Case Diagram
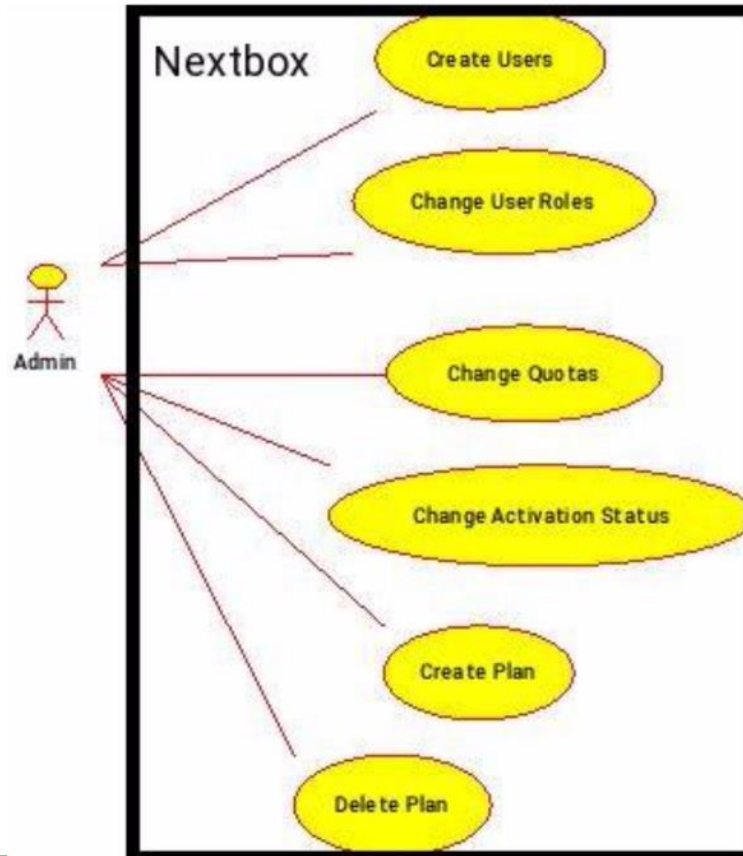
User Use Case Diagram-1

User Use Case Diagram-2

Final Class Diagram

UserDAO
- getUserByUsername(String) : User
- modifyActivationStatus(String, String) : boolean
- createAccount(User) : boolean
- createAdminAccount(User) : boolean
- updateCreditCardDetails(long, String) : boolean
- existingUsernames : List<String>

UserService
- getUserByUsername(String) : User
- createAccount(User) : boolean
- createAdminAccount(User) : boolean
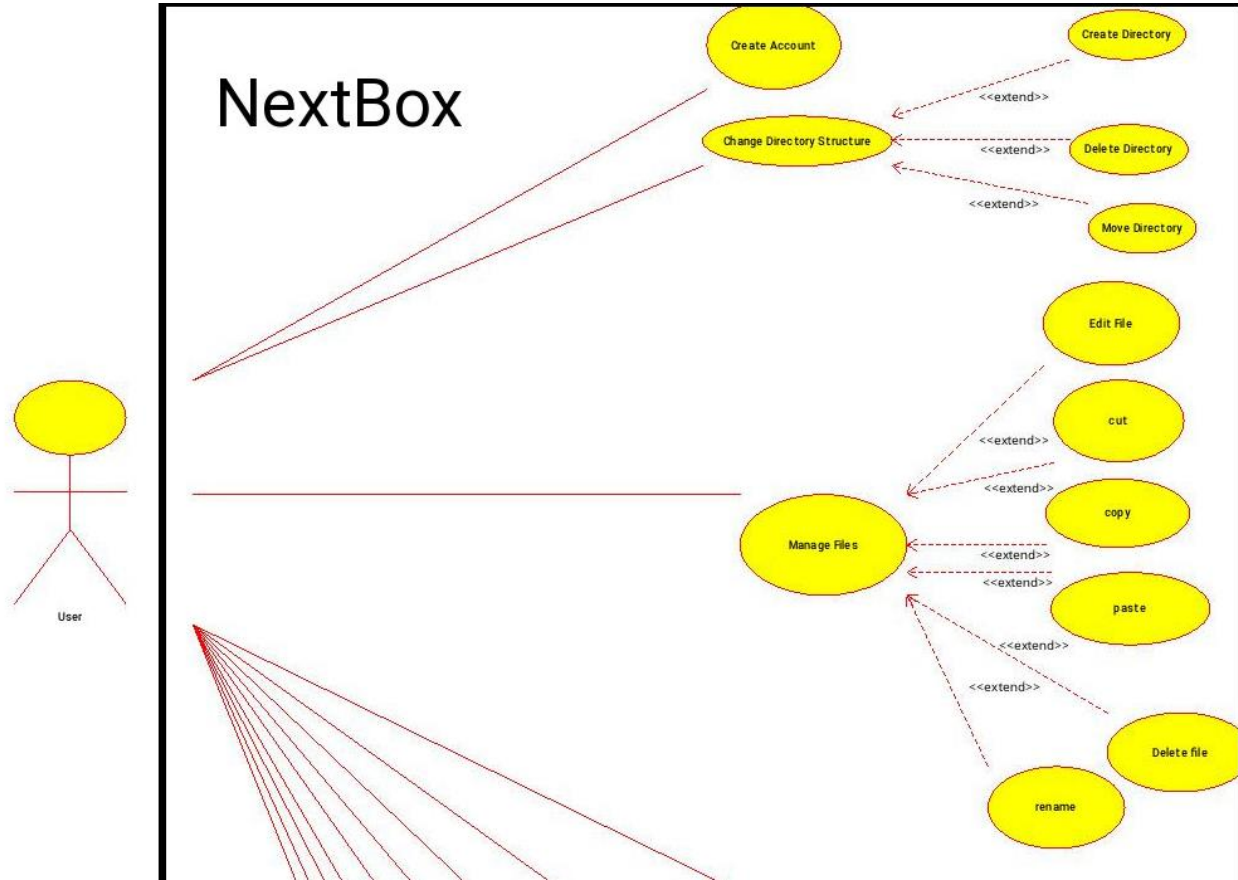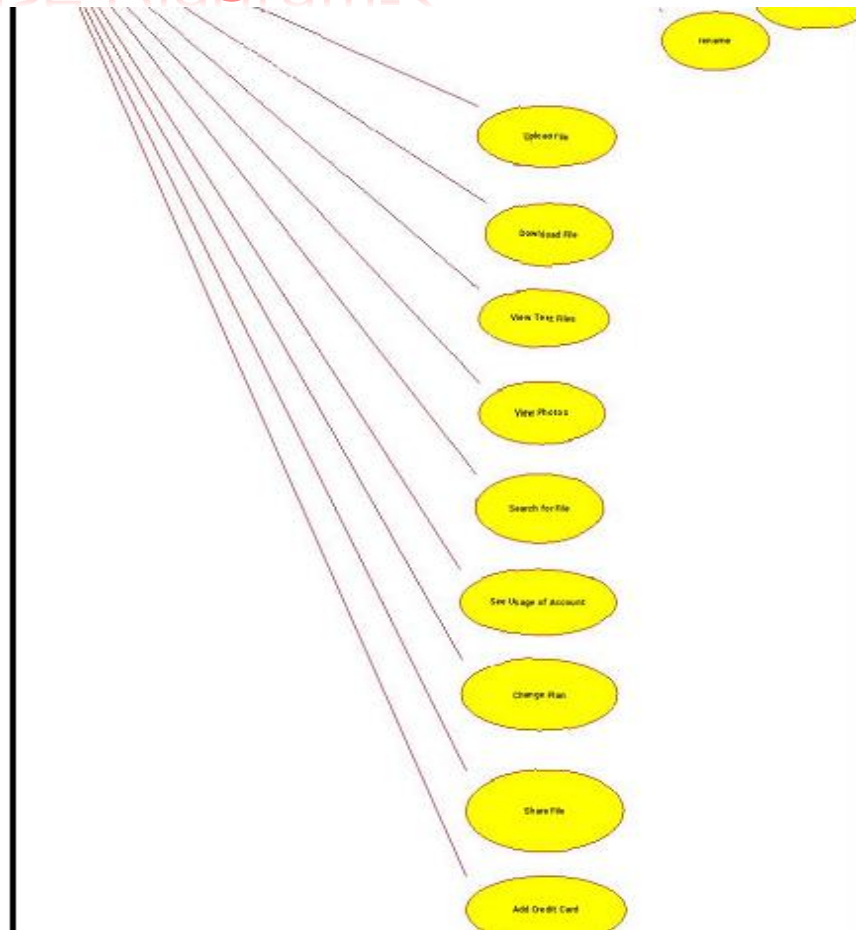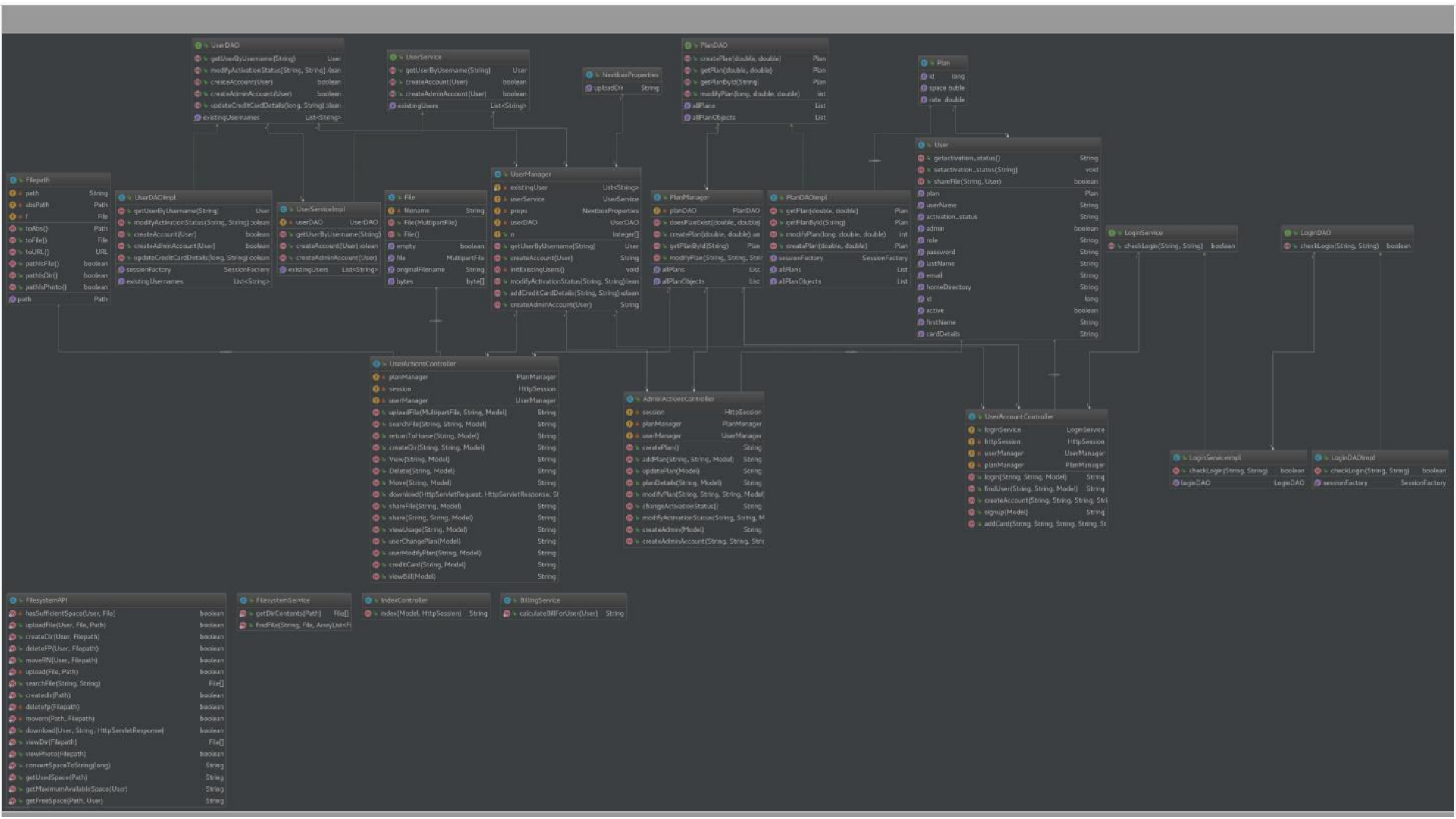- existingUsers : List<String>

NextboxProperties
- uploadDir : String

PlanDAO
- createPlan(double, double) : Plan
- getPlan(double, double) : Plan
- getPlanById(String) : Plan
- modifyPlan(long, double, double) : int
- allPlans : List
- allPlanObjects : List

Plan
- id : long
- space : double
- rate : double

Filepath
- path : String
- absPath : Path
- f : File
- toAbs() : Path
- toFile() : File
- toURL() : URL
- pathIsFile() : boolean
- pathIsDir() : boolean
- pathIsPhoto() : boolean
- path : Path

UserDAOImpl
- getUserByUsername(String) : User
- modifyActivationStatus(String, String) : boolean
- createAccount(User) : boolean
- createAdminAccount(User) : boolean
- updateCreditCardDetails(long, String) : boolean
- sessionFactory : SessionFactory
- existingUsernames : List<String>

UserServiceImpl
- userDAO : UserDAO
- getUserByUsername(String) : User
- createAccount(User) : boolean
- createAdminAccount(User) : boolean
- existingUsers : List<String>

File
- filename : String
- File(MultipartFile)
- File()
- empty : boolean
- file : MultipartFile
- originalFilename : String
- bytes : byte[]

UserManager
- existingUser : List<String>
- userService : UserService
- props : NextboxProperties
- userDAO : UserDAO
- n : Integer[]
- getUserByUsername(String) : User
- createAccount(User) : boolean
- initExistingUsers() : void
- modifyActivationStatus(String, String) : boolean
- addCreditCardDetails(String, String) : boolean
- createAdminAccount(User) : String

PlanManager
- planDAO : PlanDAO
- doesPlanExist(double, double) : boolean
- createPlan(double, double) : Plan
- getPlanById(String) : Plan
- modifyPlan(String, String, String) : String
- allPlans : List
- allPlanObjects : List

PlanDAOImpl
- getPlan(double, double) : Plan
- getPlanById(String) : Plan
- modifyPlan(long, double, double) : int
- createPlan(double, double) : Plan
- sessionFactory : SessionFactory
- allPlans : List
- allPlanObjects : List

User
- getactivation_status() : String
- setactivation_status(String) : void
- shareFile(String, User) : boolean
- plan : Plan
- userName : String
- activation_status : String
- admin : boolean
- role : String
- password : String
- lastName : String
- email : String
- homeDirectory : String
- id : long
- active : boolean
- firstName : String
- cardDetails : String

LoginService
- checkLogin(String, String) : boolean

LoginDAO
- checkLogin(String, String) : boolean

UserActionsController
- planManager : PlanManager
- session : HttpSession
- userManager : UserManager
- uploadFile(MultipartFile, String, Model) : String
- searchFile(String, String, Model) : String
- returnToHome(String, Model) : String
- createDir(String, String, Model) : String
- View(String, Model) : String
- Delete(String, Model) : String
- Move(String, Model) : String
- download(HttpServletRequest, HttpServletResponse, St
- shareFile(String, Model) : String
- share(String, String, Model) : String
- viewUsage(String, Model) : String
- userChangePlan(Model) : String
- userModifyPlan(String, Model) : String
- creditCard(String, Model) : String
- viewBill(Model) : String

AdminActionsController
- session : HttpSession
- planManager : PlanManager
- userManager : UserManager
- createPlan() : String
- addPlan(String, String, Model) : String
- updatePlan(Model) : String
- planDetails(String, Model) : String
- modifyPlan(String, String, String, Model)
- changeActivationStatus() : String
- modifyActivationStatus(String, M
- createAdmin(Model) : String
- createAdminAccount(String, String, Strin

UserAccountController
- loginService : LoginService
- httpSession : HttpSession
- userManager : UserManager
- planManager : PlanManager
- login(String, String, Model) : String
- findUser(String, String, Model) : String
- createAccount(String, String, String, Stri
- signup(Model) : String
- addCard(String, String, String, String, St

LoginServiceImpl
- checkLogin(String, String) : boolean
- loginDAO : LoginDAO

LoginDAOImpl
- checkLogin(String, String) : boolean
- sessionFactory : SessionFactory

FilesystemAPI
- hasSufficientSpace(User, File) : boolean
- uploadFile(User, File, Path) : boolean
- createDir(User, Filepath) : boolean
- deleteFP(User, Filepath) : boolean
- moveFN(User, Filepath) : boolean
- upload(File, Path) : boolean
- searchFile(String, String) : File[]
- createdir(Path) : boolean
- deletefp(Filepath) : boolean
- movern(Path, Filepath) : boolean
- download(User, String, HttpServletResponse) : boolean
- viewDir(Filepath) : File[]
- viewPhoto(Filepath) : boolean
- convertSpaceToString(long) : String
- getUsedSpace(Path) : String
- getMaximumAvailableSpace(User) : String
- getFreeSpace(Path, User) : String

FilesystemService
- getDirContents(Path) : File[]
- findFile(String, File, ArrayList<Fi

IndexController
- index(Model, HttpSession) : String

BillingService
- calculateBillForUser(User) : String

Demo