Password Manager:
CS665 Project

Package

login

Package

passwordManager

<<interface>>
LockState

quit(PasswordManager):void
login(PasswordManager,String):boolean

Login

-ciphertext:char[]
-encrypted:boolean
-password:Password
-rpg:RandomPasswordGenerator
-username:String

+createNewPassword():void
-getCharacterSet():PasswordCharacterSets
-getPasswordLength():int
+getUser():String
+getPass():String
+encrypt(String):void
+decrypt(String):void
+toCharArray():char[]

Password

-passwordText:char[]

+getChar():char[]

PasswordManager

-_DUMMY_USER_:String = "sAmPlEuSEr"
-_DUMMY_PASS_:char[] =
{'p','a','S','S','w','0','R','d'}
-encrypted:boolean
-loginFile:File
-logins:ArrayList<Login>
-loginsWithNames:HashMap<String,Login>
-master:String
-mementos:Stack<RPGMemento>
-rpg:RandomPasswordGenerator
-state:LockState
-userId:int

-createMasterPassword():void
-generateUserId():void
#getDummyLogin():Login
-getFile():void
+lockManager():void
-encryptLogins():void
-writeEncryptedLogins():void
+unlockManager(String):void
-decryptLogins():void
-populateLoginLists():void
+setState(LockState):void
+run():void
-getLogin():Login
-accessLogin(Login):void
-deleteLogin(Login):void
-createLogin():void

UnlockedState

+quit(PasswordManager):void
+login(PasswordManager,String):boolean

LockedState

+quit(PasswordManager):void
+login(PasswordManager,String):boolean

Package

pseudorandomGenerator

<<interface>>
RPGMemento

getCharSet():PasswordCharacterSets
getLength():int
getRandom():Random
getSeed():int
setCharSet(PasswordCharacterSets):void
setLength(int):void
setSeed(int):void
setRandom(Random):void

Package

encryption

<<enumeration>>
PasswordCharacterSets

ALPHA("ABCDEFGHIJKLMNOP
QRSTUVWXYZabcdefghijklmnop
qrstuvwxyz")

ALPHA_NUMERIC("ABCDEFGHIJ
KLMNOPQRSTUVWXYZabcdefghi
jklmnopqrstuvwxyz0123456789")

ALPHA_NUMERIC_WITH_
SYMBOLS("ABCDEFGHIJKLMNO
PQRSTUVWXYZabcdefghijklmnop
qrstuvwxyz0123456789!@#$%^&
*();:?><")

RandomPasswordGenerator

-length:int
-charSet:PasswordCharactterSets
-random:Random
-seed:int

+getMemento():RPGMemento
+setMemento(RPGMemento):void
+setCharSet(PasswordCharacterSets):void
+setLength(int):void
+setSeed(int):void
+generateRandomPassword():Password
-RPGMemento:Class
  getCharSet():PasswordCharacterSets
  getLength():int
  getRandom():Random
  getSeed():int
  setCharSet(PasswordCharacterSets):
void
  setLength(int):void
  setSeed(int):void
  setRandom(Random):void

AESLoginEncryption

-*algorithm:String* = "AES/GCM/NoPadding"
-*LENGTH_OF_TAG_BIT: int = 120*
-*LENTH_OF_SALT: int = 16*
-*LENGTH_OF_IV: int = 12*

+encrypt(byte[],String):String
+decrypt(String,String):String