# EncryptedData

- encryptedData: String
- + EncryptedData(encryptedData : String)
- + getEncryptedData(): String
- + setEncryptedData(encryptedData : String) : void

# DecryptedData

- decryptedData: String
- + DecryptedData(decryptedData : String)
- + getDecryptedData() : String
- + setDecryptedData(decryptedData : String) : void

### RandomNumber

- randomNumber: String
- + RandomNumber(randomNumber: String)
- + getRandomNumber(): String
- + setRandomNumber(randomNumber: String) : void

## SignedMessage

- signedMessage: String
- + SignedMessage(signedMessage: String)
- + getSignedMessage() : String
- + setSignedMessage(signedMessage: String) : void

# MiddlewareApplication

- + restC : RESTController
- + main(args : String[]) . void



## RESTController

- ${\sf scp}: {\sf ServerCommandProxy}$
- key : byte[]
- aesMasterKey : Key
- + RESTController()
- + getUserID(userName : String) : ResponseEntity<UserID>
- + getUserList(): ResponseEntity<UserList>
- + userExists(userName : String) : ResponseEntity<UserExists>
- + getUserBalance(userName : String) : ResponseEntity<UserBalance
- + aesEncryptedData(data: String) : ResponseEntity<EncryptedData>
- + aesDecryptedData(data: String) : ResponseEntity<DecryptedData>
- \+ getRandNum(): ResponseEntity<RandNum>
- + getSignedMessage(): ResponseEntity<SignedMessage>



# ServerCommandProxy

- xmlRpcClient : final XmlRpcClient
- \_authToken : final String
- + ServerCommandProxy(server : String, port : int. authToken : String) : void
- + getUserAccountBalance(userName : String) : double
- + getUserProperty(userName : String, propertyName: String) : String
- + isUserExists(userName : String) : Boolean

only the used methods are shown

#### UserID

- userID: String
- + UserID(userID : String)
- + getUserID() : String
- + setUserID(userID : String) : void

#### UserList

- userList : Vector<String>
- + UserList(userList : Vector<String>)
- + getUserList() : Vector<String>
- + setUserList(userList: Vector<String>): void

## UserExists

- userExists: Boolean
- + UserList(userExists : Boolean)
- + getUserExists(): Boolean
- + setUserExists(userExists : Boolean) : void

## UserBalance

- UserBalance: double
- + userBalance(userBalance : double)
- + getUserBalance() : double
- + setUserBalance(userBalance : double) : void