

## MainApp

```
-primaryStage: Stage
-loginLayout: BorderPane
-newTaskLayout: AnchorPane
-clienControl: ClientControl
+loginName: String
-----
+MainApp()

+start(primaryStage: Stage): Void
+initRootLayout(): Void
+showLoginScreen(): Void
+showMainScreen(): Void
+showProjectScreen(): Void
+showNewTaskDialog(): Boolean
+getPrimaryStage(): Stage
+getInstance(): MainApp
+showEditTaskDialog(selectedTask: ITask): Boolean
+showProjectInsertDialog(): Void
+main(args: String[]): Void
```

## ClientControl

```
-instance: ClientControl
-openProject: IProject
-sleepingProjects: List(IProject)
-receivedSimpleProjects: List(SimpleProject)
-mainApp: MainApp
-clientThread: Thread
-kClient: KClient
-----
+ClientControl()
+getInstance(): ClientControl
+userLogin(name: String): Void
+simpleUserReturnedFromLogin(simpleUserObj: SimpleUser)
+simpleProjectsReturnedFromLogin(return List: List(SimpleProject)
+projectCreate(): Boolean
+projectNewTask(): Boolean
+getOpenProject(): IProject
+setOpenProject(openProject): void
+getUser(): User
+getSimpleProjects(): List(SimpleProjects)
+getReceivedSimpleProjects(): List(SimpleProject)
+setReceivedSimpleProjects(receivedSimpleProjects: List(SimpleProject)
```

## KClient

```
-socket: Socket
-objectOutputStream: ObjectOutputStream
-objectInputStream: ObjectInputStream
-port: Integer
-url: String
-clientControl: ClientControl
-message: Object
-----
+KClient(url: String, port: Integer)
+requestSimpleProjects(): Void
+connectToServer(addr: InetAddress, port Integer) Void
+sendNewProject(project: Project): Void
+run(): Void
```

### User

-uid: String

-name: String

-pw: String

-projects: List(SimpleProjects)

-projectCurrent: Project

---

+User(uid: String, name: String, pw: String,  
projects: List(Projects), projectCurrent: Project)

+getUID(): String

+setUID(uid: String): Void

+getName(): String

+setName(name: String): Void

+getPw(): String

+setPw(pw String): void

+getProjects(): List(String)

+setProjects(projects: List(String)): Void

+getProjectCurrent():String

+setProjectCurrent(projectCurrent: String):  
Void

### LoginScreenController

-mainApp: MainApp  
-clientControl: ClientControl

---

+initialize(): Void  
+LoginController()  
+setPrevStage(szage: Stage): Void  
+dopLoginButton(): Void  
+setMainApp(mainApp: MainApp): Void

### MainScreenController

-control: ClientContro-  
-mainApp: MainApp

---

+initialize(): Void  
+MainScreenController()  
+handleNewTask(): Void  
+handleEditPerson(): Void  
+setMainApp(mainApp: MainApp): Void  
showTaskDetails(task: ITask): Void

### NewTaskDialogController

-task: ITask  
-control: ClientControl  
-okClicked: Boolean

---

+initialize(): Void  
+setTask(task: ITask): Void

### NewProjectDialogController

-clientControl: ClientControl  
-okClicked: Boolean

---

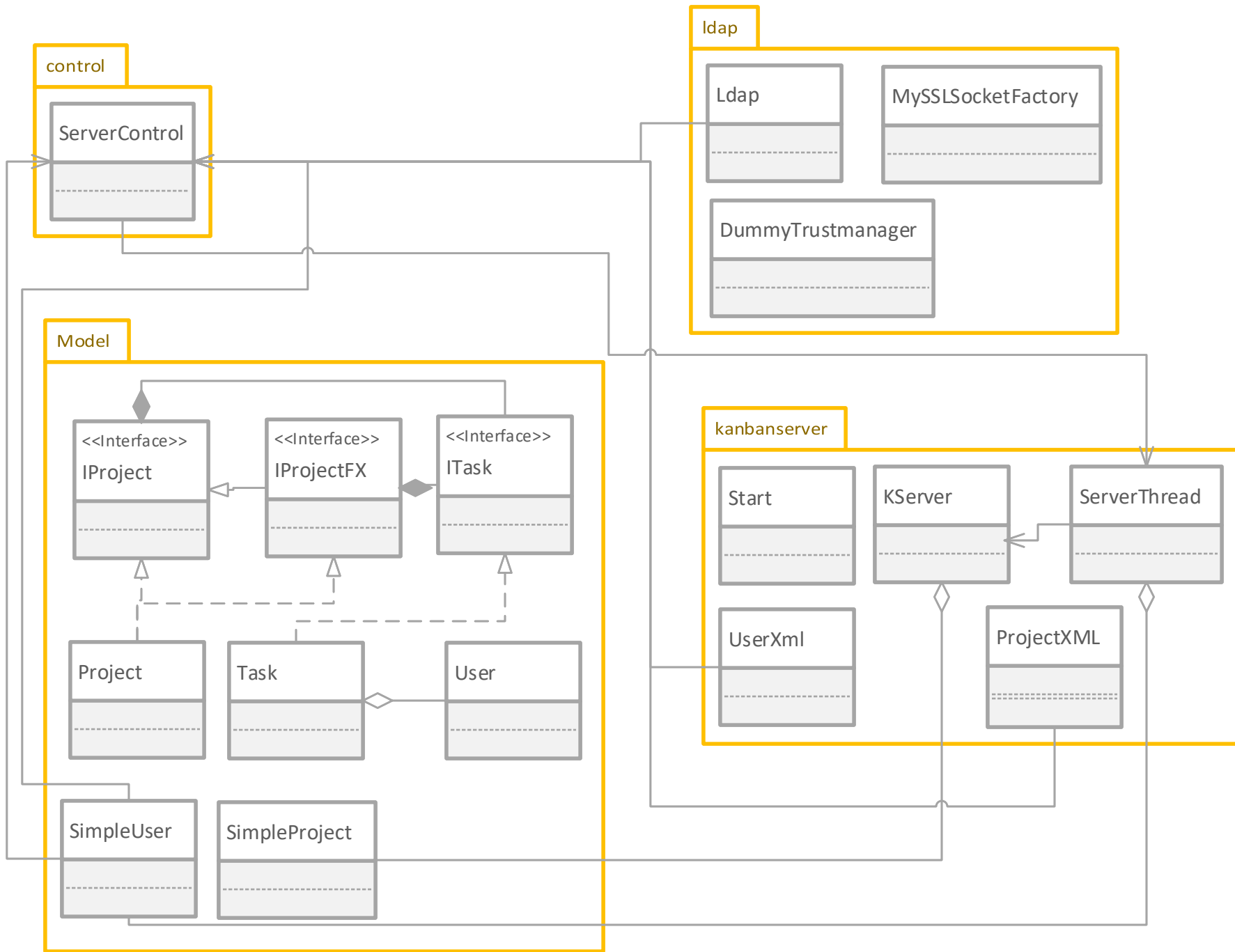
+initialize(): Void  
+setProject(): Void  
+handleOK(): Void  
+handleCancel(): Void

### ProjectSelectionScreenController

-mainApp: MainApp  
-clientControl: ClientControl  
-listOfProjectName: List(SimpleProject)

---

+initialize(): Void  
+ProjectSelectionScreenController()  
+handleProjectSelection(): Void  
+handleNewProject(): Void  
+handleAccept(): Void  
+setMainApp(mainApp: Mainpp): Void



control

## ServerControl

-ldap: Ldap

-userXML: UserXML

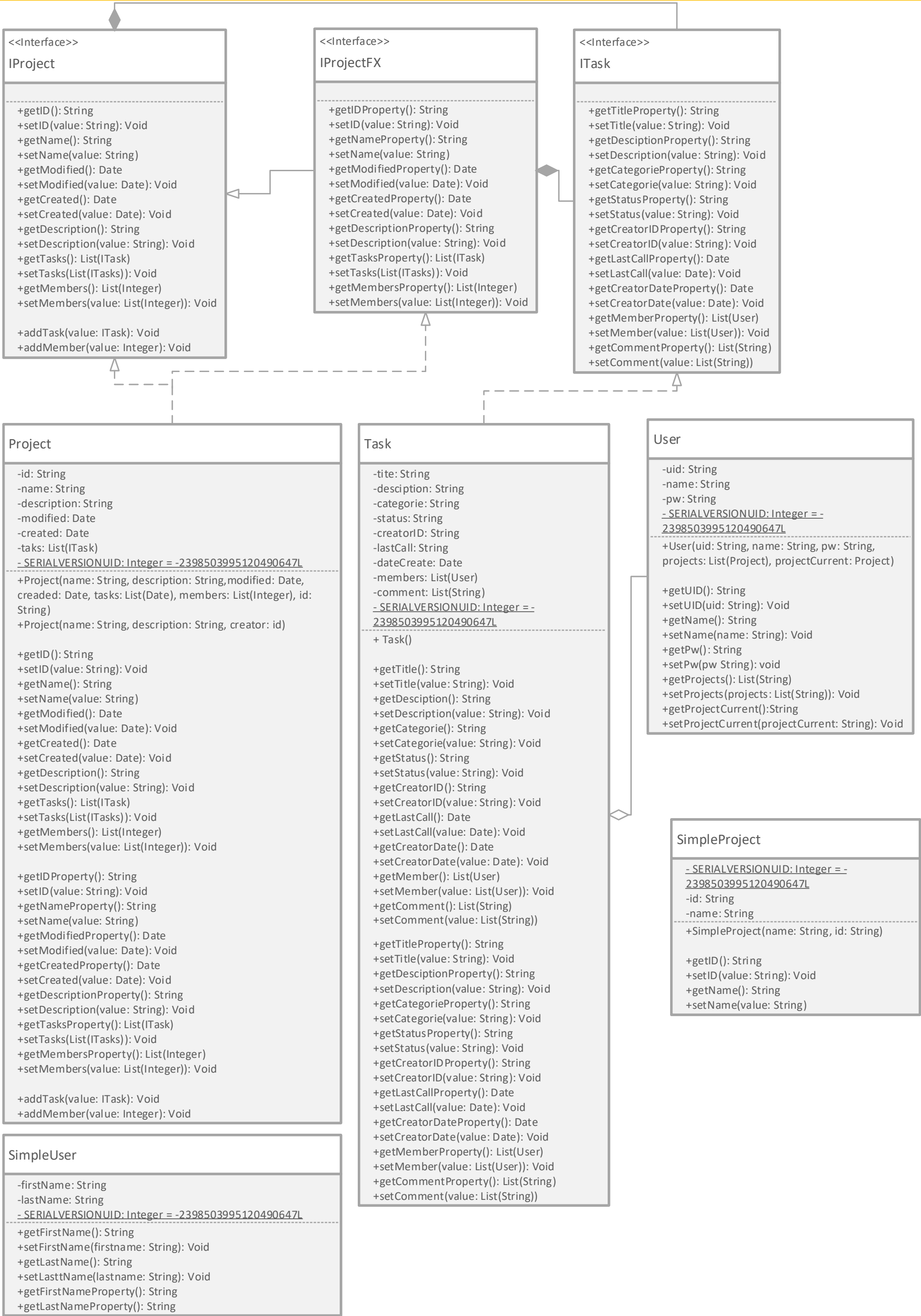
-projectXML: ProjectXML

---

+userLogin(userName: String): Boolean

+getProjectsForUserLogin(userName: String):  
ArrayList(SimpleProject)

+createNewProject(newProject: Project): Boolean



## Ldap

```
-Host: String
-SearchBase: String
- LdapContext: DirContext
-----
+Ldap(Hostname: String, LdapDirectory: String)

- connect: Boolean
-disconnect: Void
-search( SearchBase: String, SearchFilter: String, User: SimpleUser): Boolean
+login(HemsUid: String, User: SimpleUser): Boolean
```

## DummyTrustmanager

```
-----
+checkClientTrusted(cert: Certificate[], strin: String) Void
+checkServerTrusted(cert: Certificate[], strin: String): Void
+checkServerTrusted(cert: Certificate, string: String): Certificate[]
```

## MySQLSocketFactory

```
- socketFactory: SSLSocketFactory
-----
+MySQLSocketFactory()
+getDefault(): SocketFactory
+getDefaultCipherSuites(): String[]
+getSupportedCipherSuites(): String[]
+createSocket(socket: Socket, string:
String, num: Integer, bool: Boolean):
Socket
+createSocket(string: String, num:
Integer): Socket
+createSocket(string: String, num:
Integer, netAdd: InetAddress, i: Integer):
Socket
+createSocket(netAdd: InetAddress,
num: Integer): Socket
+createSocket(netAdd1: InetAddress,
num: Integer, netAdd: InetAddress, i:
Integer): Socket
```



