

View

LoginScreenController

-mainApp: MainApp
-clientControl: ClientControl

+initialize(): Void
+LoginController()
+setPrevStage(stage: Stage): Void
+doLoginButton(): Void
+setMainApp(mainApp: MainApp): Void

MainScreenLoginScreen

-control: ClientControl
-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

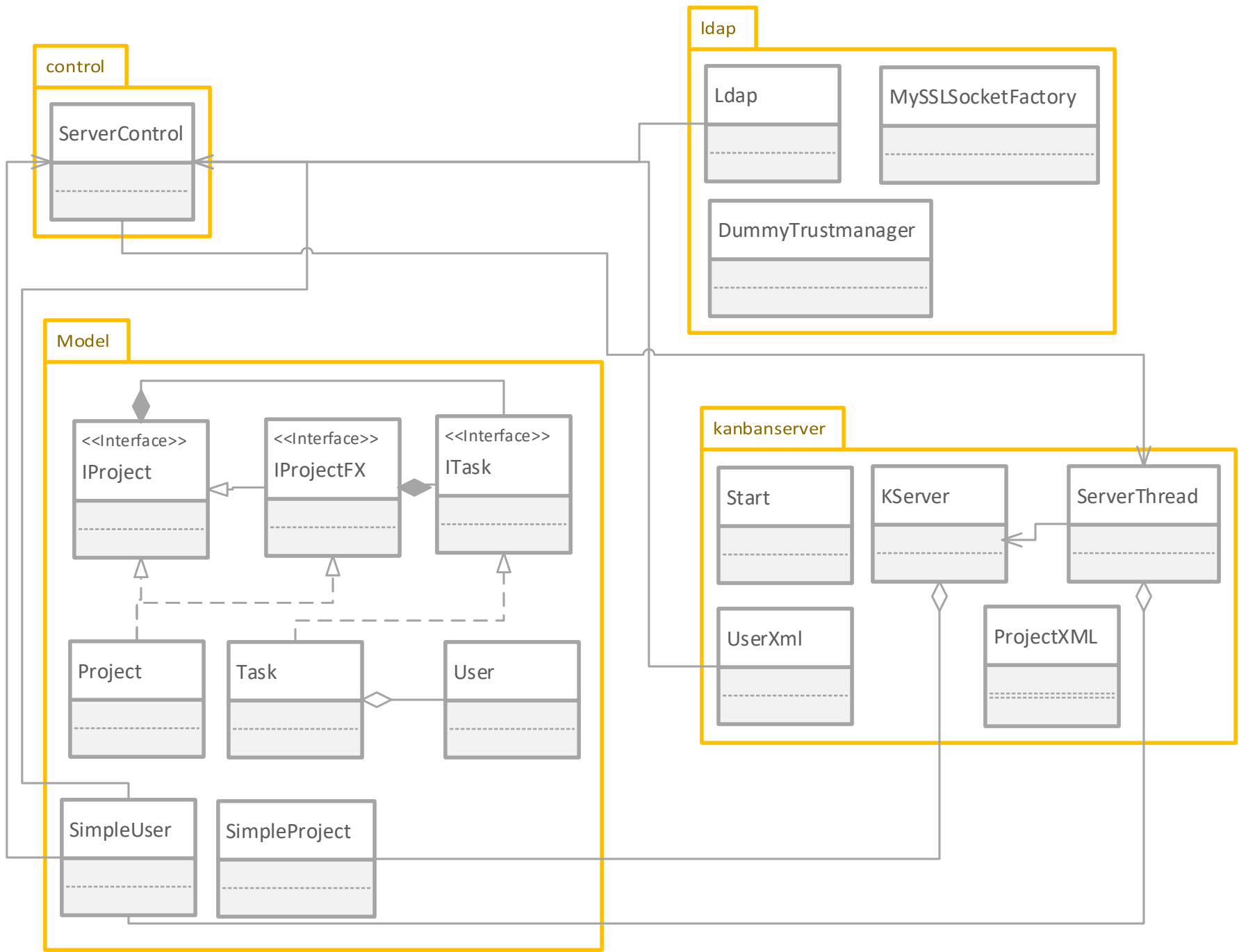
-okClicked: Boolean

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

ProjectSelectionScreenController

-mainApp: MainApp
-listOfProjectName: List(SimpleProject)

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



control

ServerControl

-ldap: Ldap

-userXML: UserXML

+userLogin(userName: String): Boolean

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



Ldap

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, host: String, num: Integer, bool: Boolean): Socket
+createSocket(string: String, host: String, num: Integer): Socket
+createSocket(string: String, host: String, num: Integer, netAdd: InetAddress): Socket
+createSocket(netAdd: InetAddress, host: String, num: Integer): Socket
+createSocket(netAdd1: InetAddress, host: String, num: Integer, netAdd: InetAddress): Socket
```


kanbanserver

Start

+main(args: String[]): Void

KServer

#serverThreads: List(ServerThread)

-listofObjects: List(SimpleProject)

+main(args: String[]): Void

ServerThread

#receivingMessages: Boolean

#currentObject: Object

#objectOutputStream: ObjectOutputStream

#objectInputStream: ObjectInputStream

#socket: Socket

+ServerThread(socket: Socket)

+sendSimpleProject(userName: String): Void

+sendSimpleUser(userName: String): Void

+sendString(str: String)

+run(): Void

UserXml

-userXMLDirectory: String

+userXML(directory: String, login: String)

+readUserXML(): User

+writeUserXML(user: User): Void

ProjectXML

-projectXMLDirectory: String

+projectXML(directory: String, login: String)

+readProjectXML(): Project

+writeProjectXML(project: Project): Void