Architektur

**View**

**XHTML**

**CSS**

**JS**

**Images**

**Beans**

**Dao**

**Common**

**DbConnect**

**Util**

**webapp**

**java**

**TO**

**DB TO**

**DB**

**DB DataBse**

**Table**

**User**

**CUSTOM-xhtml**

**LoginBeanString** username

**String** password

**UserDao**

**webapp**

**java**

**login.xhtml**

**UserTO**

**DBUserConnect**

**UserTO**

**String** loginname

**String** password

**String** firstname

**String** lastname

**java**

**Common**

**UserTO**

**UserBean String** loginname

**String** password

**String** firstname

**String** lastname

void setUser(UserTO userTO)

String login()

UserTO getUserForLogin()

UserTO readUserForLogin()

**UserTO**

**DB**

**DataBase**

ResultSet readUserForLogin()

**user**

**String** loginname

**String** password

**String** firstname

**String** lastname

Beispielhaft zeigen die obigen Diagramme den Weg der Datenanforderung bis zur Datenlieferung anhand des Logins.

Beim starten der Anwendung erscheint die Kundenspezifische Login Datei ( z.b mm.xhtml ) die direkt die login.xhtml includiert.

Die beiden Eingabefelder Loginname und Passoword respresentiert durch die beidem Inputfelder

<p:inputText id=*"id\_loginpage\_loginname\_input"*

onchange=*"js\_common\_check\_login()"*

onblur="js\_common\_check\_login()"

onkeypress="js\_common\_check\_login()"

value=*"#{loginBean.loginName}"* size=*"30"* />

<p:password id=*"id\_loginpage\_password\_input"*

onchange=*"js\_common\_check\_login()"*

onkeypress="js\_common\_check\_login()"

onblur="js\_common\_check\_login()"

value=*"#{loginBean.password}"* size=*"30"* />

Diese sind gebunden an die LoginBean. Und an die Properties

**private** String loginName="";

**private** String password="";

Der Button der login.xhtml ist gebunden an die Methode

**public** String login()

Diese holt über den Aufruf UserTO userTO = UserDao.*getUserForLogin*(loginName, password); der Klasse **UserDao** ein **UserTO** Object.

Die Klasse **UserDao** holt in der Methode **getUserForLogin(loginName, password)**

über die Klasse **DBUserConnect** in der Methode **readUserForLogin(loginName, password)** das **UserTO** Object.