Nutzer Anmeldung / Registrierung

Nutzen Sie JSF-Templates, um ein einheitliches Erscheinungsbild für die Aufgabe zu erzeugen.

Bitte führen Sie die einzelnen Aufgaben nacheinander durch.

In der Applikation sollen sich Benutzer registrieren können.

Für die Benutzer sind mindestens folgende Eigenschaften vorgesehen:

* Name  
  Maximale Länge 20 Zeichen
* Passwort  
  Regeln für das Passwort
  + Min. 1 Kleinbuchstabe
  + Min. 1 Großbuchstabe
  + Min. 1 Zahl

Für das Password kann folgendes Pattern helfen [A-Z]+[a-z]+[0-9]+  
Die Klasse Benutzer kann und soll bei Bedarf angepasst werden.

# Übersichtsseite

Erzeugen Sie eine Übersichtsseite für diese Übung. Die Seite soll über einen Bereich header, content und footer verfügen. Im Header Bereich erzeigen Sie bitte eine Navigation für die einzelnen Abschnitte der Aufgabe.

Im footer lassen Sie sich die aktuelle View ausgeben. Beispiel: #{view.viewId}

# Validierung mit JSF

Erzeugen Sie eine JSF-Seite für die Nutzerregistrierung.

Validieren Sie die Daten innerhalb der JSF mit den <f:validate…. Tags.

Lassen Sie sich die Nachrichten für fehleingaben ausgeben.

# Validierung durch die Bean

Legen Sie hierfür eine neue JSF-Seite für die Nutzerregistrierung an.

Lassen Sie die Daten nun innerhalb der Bean mit den Built-In Validierungs Annotations prüfen.

Lassen Sie sich auf der JSF-Seite wieder die Nachrichten für Fehleingaben ausgeben.

# Validierung der gesamten Bean

Legen Sie erneut eine neue JSF-Seite für die Nutzerregistrierung an.

Jetzt sollen die Daten nicht mehr einzeln geprüft werden, sondern die Bean soll als Gesamtkonstrukt geprüft werden.

Dazu schreiben Sie einen eigenen BeanValidator.  
Die WholeBeanValidation muss per JSF Tag, Per Annotation und per Konfiguration aktiviert werden.

# Erweitern der Applikation I

Für den Benutzer soll es möglich sein eine E-Mail anzugeben. Bei der E-Mail wird unterschieden zwischen interne und externe E-Mail.

* Interne E-Mails dürfen ohne @provider.de angegeben werden.  
  Max.Muster
* Externe E-Mails müssen mit einem @provider.de angegeben werden.  
  Max.Muster@example.com

Für diese Eingabefeld erzeugen Sie bitte einen eigenen Converter. Dieser soll E-Mails, welche ohne Provider angegeben wurden, um den Provider @intern.cc erweitern. Im Beispiel von Max.Muster wäre das Ergebnis [Max.Muster@intern.cc](mailto:Max.Muster@intern.cc) welches dann auch in das Benutzer Objekt gespeichert werden soll.

# Erweitern der Applikation II

Erweitern Sie die Applikation um eine Controller Klasse für die Benutzer.

Diese Controllerklasse soll eine Liste aller registrierten Benutzer enthalten. Dafür soll die Controllerklasse beim Start der Applikation initialisiert werden.

Alle Benutzer, die sich erfolgreich registriert haben, werden in die Liste aufgenommen.

Nun erzeugen Sie eine JSF-Seite, wo alle Registrierten Benutzer aufgelistet werden.

# Erweitern der Applikation III

In einer JSF-Seite soll ein Benutzer sich anmelden können. Dazu werden die eingegebenen Daten mit den Daten aus der Benutzerliste im Controller verglichen. Sollte Name und Passwort übereinstimmen war die Anmeldung erfolgreich falls nicht wird der Benutzer zu einer Fehlerseite geleitet.

Erstellen Sie eine JSF-Seite, wo alle angemeldeten Benutzer aufgelistet werden.

# Erweitern der Applikation IV

Erstellen Sie eine JSF-Seite für die Benutzerdaten. Hier soll neben dem Namen und dem Passwort (Ja darf im Klartext sein) eine Selektion erstellt werden.  
Die Selektion soll folgende Werte anbieten

* Ofen A
* Ofen B
* Ofen A und B

Die Daten, wenn die diese geändert wurden, sollen über einen Speichern Button in das Benutzerobjekt übertragen werden.

Im footer soll ab jetzt immer ein Hinweis ausgegeben werden, welche Selektion der User angewählt hat.