

Personenverwaltung mit mehreren Ansichten



Beschreibung

Dieses Projekt beinhaltet die Aufgabe zur Entwicklung einer Personenverwaltung mit Spring Boot und Thymeleaf. Das Projekt beinhaltet mehrere Fenster für die Darstellung und Bearbeitung der Personen. Das Projekt erfordert den Umgang mit Beans und Modells sowie den Einsatz einer Datenbank.

Form: Einzelarbeit

Zeit: ca. 4 Lektionen

Abgaben: <https://www.lanza.ch/5IA20c.html>

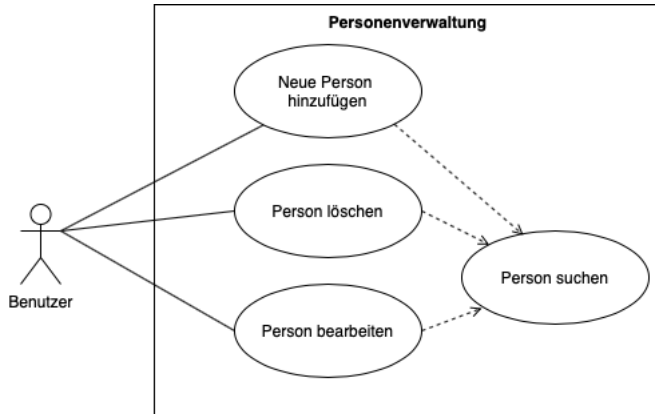
Inhaltsverzeichnis

1	PERSONENVERWALTUNG.....	2
1.1	EINLEITUNG	2
1.2	GRAFISCHE DARSTELLUNG	2
2	RAHMENBEDINGUNGEN	3
2.1	TECHNISCHE RAHMENBEDINGUNGEN	3
2.2	TECHNISCHER KRAM UND TUTORIALS	3
2.3	FUNKTIONALE ANFORDERUNGEN AN DIE WEB-APPLIKATION	3
3	BEWERTUNG.....	5
3.1	BEWERTUNGSKRITERIEN	5

1 Personenverwaltung

1.1 Einleitung

Für die Verwaltung von Personen soll eine Web-Applikation mit Spring-Boot und Thymeleaf erstellt werden. Die folgenden Anwendungsfälle (ohne Personensuche) sollen dabei umgesetzt werden:



1.2 Grafische Darstellung

Die Personen können beispielsweise in einer tabellarischen Liste erscheinen. Auf jeder Zeile könnte der Vor- und der Nachname, sowie die E-Mail-Adresse der Person stehen.

Personenverwaltung

Übersicht

Vorname	Nachname	E-Mail		
Yoko	Jenkins	lorem@vitaeeratVivamus.org	bearbeiten	löschen
Zelenia	Crosby	neque.sed.dictum@Sedcongue.com	bearbeiten	löschen
Ali	Tran	arcu@tempor.ca	bearbeiten	löschen
Raymond	Gentry	turpis@velitdui.net	bearbeiten	löschen
MacKenzie	Chan	in@ornareFuscemollis.org	bearbeiten	löschen

Person hinzufügen

Musterfenster zur Bearbeitung der Personendaten.

Personenverwaltung

Person bearbeiten

Vorname

Lorem ipsum

Nachname

Lorem ipsum

Email address

Lorem ipsum

2 Rahmenbedingungen

2.1 Technische Rahmenbedingungen

Die folgenden technischen Rahmenbedingungen sollen im Projekt eingehalten werden:

- Spring Boot und Thymeleaf als Basistechnologie
- Bootstrap als Layout und Gestaltungsmittel
- CDI für die Umsetzung einer mehrschichtigen Architektur
- JPA für die Speicherung der Personendaten
- MySQL bzw. MariaDB als persistenten Datenspeicher

Die Applikation soll nach den Prinzipien der objektorientierten Programmierung (SOLID) umgesetzt werden.

<https://www.baeldung.com/solid-principles>

Spring Boot Dependency Injection.

<https://www.baeldung.com/spring-dependency-injection>

2.2 Technischer Kram und Tutorials

Forms mit Bootstrap gestalten:

<https://getbootstrap.com/docs/4.0/components/forms/>

Buttons mit Bootstrap gestalten:

<https://getbootstrap.com/docs/4.0/components/buttons/>

Hintergrundinformation zur Standard URL Syntax von Thymeleaf:

<https://www.thymeleaf.org/doc/articles/standardurlsyntax.html>

2.3 Funktionale Anforderungen an die Web-Applikation

Bearbeitungsformular umsetzen

Wenn eine Person angewählt wurde, erscheint das Bearbeitungsformular. Achten Sie darauf, dass Änderungen an den Daten nur übernommen werden, wenn tatsächlich speichern gedrückt wird. Beim Drücken des Buttons «abbrechen» dürfen Änderungen nicht übernommen werden.

Person hinzufügen

Die Webapplikation soll das Hinzufügen einer neuen Person ermöglichen. Sie dürfen hierfür dasselbe Formular wie für die Bearbeitung der Personendaten verwenden oder ein anderes programmieren. Achten Sie darauf, dass die Benutzerführung klar ist.

Funktion löschen

Implementieren Sie die Funktion, um eine Person zu löschen.

Fortgeschrittene: Ergänzen Sie die Lösch-Funktion mit einer «sind Sie sicher» Abfrage. Orientieren Sie sich dabei an den Möglichkeiten die Bootstrap bietet.

Personendaten

Die Personen sollen folgende Eigenschaften erhalten:

- Vorname
- Nachname
- E-Mail-Adresse
- Geburtsdatum
- Geschlecht (M/F)

Speicherung

Die Daten der Personen sind persistent zu speichern.

Validierung der Werte

Die Eigenschaften der Personen sollen bei der Bearbeitung und Speicherung überprüft werden. Dabei kommen die folgenden Validierungsregeln zu tragen:

- Vorname und Nachname: 2 bis 30 Zeichen
- E-Mail-Adresse: gültige Adresse (<https://www.baeldung.com/java-email-validation-regex>)
- Geburtsdatum: gültiges Datum vor dem 1. Januar 2005
- Geschlecht: Eingaben wie M/F oder m/f, beim Speichern immer m/f kleingeschrieben

Anzeige der entsprechenden Fehlermeldungen.

Responsive Design

Die Web-Applikation soll für die Nutzung auf einem Computer sowie auf dem Smartphone optimiert werden. Die Ansichten und Formulare sollen entsprechend gestaltet werden.

3 Bewertung

3.1 Bewertungskriterien

Programmierung nach dem SOLID Ansatz (4 Punkte)

- Klassen haben genau eine Verantwortung und (open for extensions/close for modifications)
- Dependency Injection eingesetzt

Einsatz der Beans und Scopes (6 Punkte)

- Annotationen
- Nutzung (Eigenschaften und Konstruktoren)
- Architektur der Web-Applikation

Einsatz von Bootstrap, Darstellung der App (4 Punkte)

- Responsive Design
- Formularfelder

Bearbeitung der Personendaten (8 Punkte)

- Einsatz und Auswahl der Felder
- Validierung der Felder
- Anzeige von Meldungen
- Verarbeitung des Modells

Neue Person hinzufügen (2 Punkte)

Navigation zwischen den Fenstern (2 Punkte)

Bestehende Person Löschen (2+4 Punkte)

- Funktion löschen
- Bestätigung beim Löschen

Persistente Speicherung der Daten (4 Punkte)

- Nutzung von JPA
- Korrekte Umsetzung

Total 36 Punkte