

Operations-Spezifikationen mittels Schablone

Hier sind einige der Operationen der Muß-Funktionalität, geordnet nach Häufigkeit, spezifiziert. Klassenoperationen sind unterstrichen.

Klasse: **Nachricht**

Zweck: Eine Nachricht, die eine Lehrkraft betrifft und an alle Interessenten bezüglich dieser Lehrkraft versandt werden soll. **Bsp.:** Krankmeldung

Signatur:	<u>lehrkraftbezogeneNachrichtVersenden(</u> <u>user: Person</u> <u>nachrichtenquelle: Person</u> <u>inhalt: String</u> <u>): Nachricht</u>
Effect:	Versendet eine eMail mit dem Titel „Nachricht betreffend die Lehrkraft <i>user.name</i> “ an alle Personen, die als Interessent bei der Lehrkraft <i>nachrichtenquelle</i> abonniert haben. Darin wird auch darauf hingewiesen, unter welcher URL die Interessenten ihre Lehrkraftabonnements verwalten können. Die Nachricht wird gespeichert mit den Angaben <i>inhalt</i> als <i>inhalt</i> , sowie zwecks Nachvollziehbarkeit mit <i>user.username</i> als <i>editUsername</i> und aktuellem Datum/Uhrzeit als <i>editTimestamp</i> .
In:	<i>user</i> - die Person, die die Nachricht versenden will, typischerweise eine Mitarbeiterin des Dekanats.
In:	<i>nachrichtenquelle</i> - die Person, betreffend welche eine Nachricht versendet werden soll, typischerweise eine Lehrkraft.
In:	<i>inhalt</i> - der frei formulierte Inhalt der Nachricht mit einer typischen Länge von 2-4 Zeilen. Bsp.: „Prof. Knabe ist krank vom 28.11.03 – 03.12.03. Alle seine Lehrveranstaltungen fallen aus.“
Out:	Die erzeugte und versandte Nachricht.
Except:	<ul style="list-style-type: none"> • Aktueller Benutzer {0} hat weder die Rolle Admin noch Dekanat: <i>UserNichtBerechtigtExc</i> • Die angegebene Nachrichtenquelle ist keine Lehrkraft: <i>NichtLehrkraftExc</i> wenn <i>nachrichtenquelle.status</i> keine Lehrkraft ist. Siehe <i>PersonStatus.istLehrkraft()</i> • Der Nachrichteninhalt ist länger als 10 Zeilen oder 10*80 Zeichen: <i>InhaltZuLangExc</i> Begründung: Die Nachrichten sollen auch in Tabellenform noch sinnvoll darstellbar sein. Statt eines längeren

Textes ist auf dessen URL im WWW zu verweisen.

Signatur:	<u>lehrkraftbezogeneNachrichtenSuchen(</u> <u> lehrkraft: Person</u> <u>): Nachricht_List</u>
Effect:	Sucht alle Nachrichten, die <code>lehrkraft</code> betreffen.
Out:	Eine Liste der gefundenen Nachrichten
Except:	<ul style="list-style-type: none"> Die angegebene Nachrichtenquelle ist keine Lehrkraft: <code>NichtLehrkraftExc</code> wenn <code>nachrichtenquelle.status</code> keine Lehrkraft ist. Siehe <code>PersonStatus.istLehrkraft()</code>

Klasse: **Abonnement**

Zweck: Drückt aus, daß eine Person als Interessent Nachrichten bezüglich einer Person als Nachrichtenquelle abonniert hat.

Signatur:	<u>beiLehrkraeftenAbonnieren(</u> <u> user: Person</u> <u> nachrichtenquellen: Person_List</u> <u>): Person_List</u>
Effect:	<p>Abonniert die Nachrichten bei allen in <code>nachrichtenquellen</code> aufgeführten Personen, typischerweise Lehrkräften, und storniert eventuell bestehende Abonnements bei allen anderen Personen. Speichert die neuen Abonnements mit <code>user.username</code> als <code>editUsername</code> und aktuellem Datum/Uhrzeit als <code>editTimestamp</code>.</p> <p>Der <code>user</code> wird per eMail benachrichtigt über die stornierten, die neu getätigten und die dadurch bestehenden Abonnements.</p>
In:	<code>user</code> - die Person, die Nachrichten abonnieren will, typischerweise ein Student.
In:	<code>nachrichtenquellen</code> - Liste der Personen, typischerweise Lehrkräfte, bei denen <code>user</code> Nachrichten abonnieren will.
Out:	Neu berechnete Liste aller Lehrkräfte, bei denen der <code>user</code> keine Nachrichten abonniert hat.

Klasse: Person

Zweck: Darstellung einer im Dekanat erfaßten Person oder eines authentifizierten Systembenutzers oder beides in einem. Die Verbindung erfolgt über das Attribut `username`.

Signatur:	<u>alleLehrkraefteSuchen(): Person_List</u>
Effect:	Sucht alle gespeicherten Personen, die ihr <code>status</code> als Lehrkraft ausweist. Siehe <code>PersonStatus.istLehrkraft()</code> .
Out:	Liste der gefundenen Lehrkräfte

Signatur:	<u>eigeneNachrichtenquellenSuchen(): Person_List</u>
Effect:	Sucht alle Personen, bei denen von der aktuellen Person als Interessent ein Nachrichten-Abonnement besteht.
Out:	Eine Liste der gefundenen Nachrichtenquellen

Signatur:	<u>personenAusDateiEintragen(</u> <u> user: Person,</u> <u> personendatei: InputStream</u> <u>): Exception_List</u>
Effect:	Liest <code>personendatei</code> als XML-Text und erzeugt aus jedem <code><person></code> -Element eine persistente Person, wobei der Inhalt jedes Kindelements von <code><person></code> , z. B. <code><name></code> in das gleichnamige Attribut als Wert übernommen wird. Wenn eine Person schon existiert, festzustellen anhand der Gleichheit der <code>dekanatsId</code> , werden ihre Attributwerte aktualisiert. Beim Erzeugen oder Aktualisieren einer Person wird <code>user.username</code> als <code>editUsername</code> und aktuelles Datum/Uhrzeit als <code>editTimestamp</code> eingetragen.
Out:	Liste der Ausnahmen, die bei der Verarbeitung der XML-Datei auftraten.

Signatur:	<u>allePersonenLoeschen()</u>
Effect:	Benachrichtigt mit einer eMail alle Interessenten, die bei mindestens einer Person Nachrichten abonniert haben, von dem Ende der Abonnements. Löscht alle Personen, Abonnements und Nachrichten.