|  |  |
| --- | --- |
| **Programmierschnittstellen und Softwarequalität**  **Prof. Dr. Ursula Oesing** |  |
| **Projekttermin 5: Akzeptanztests mit JBehave**  **und Use Case Test** |

**5. Akzeptanztests mit JBehave und Use Case Test**

**5.1 Informationen und Voraussetzungen zu diesem Projekttermin**

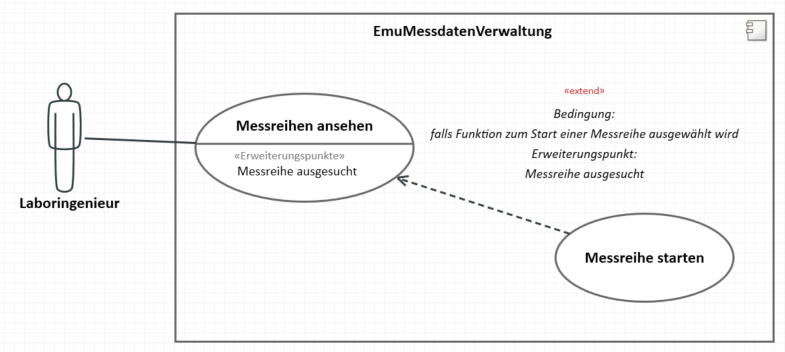
Vorhanden ist die Lösung des vierten Projekttermins. Es werden jetzt Akzeptanztests mit JBehave durchgeführt.

**5.2 Vorbereitungen zu diesem Projekttermin**

Erstellen Sie ein weiteres Testfolder *testJBehave.* In der Datenbank müssen  
 Messreihen vorhanden sein, davon mindestens eine mit Messungen und  
 mindestens zwei Messreihen ohne Messungen.

**5.3 Aufgaben dieses Projekttermins**

Erstellen Sie innerhalb von *testJBehave* einen Use Case Test mit JBehave zu dem folgenden Ausschnitt aus einem UML Use Case Diagramm.



Die User Story lautet:

*Als Laboringenieur möchte ich Messreihen starten können, um Messungen zu einer Messreihe zu erhalten.*

**Weitere Informationen zu der Aufgabe**

**Sie brauchen für diesen Projekttermin nur Methoden aus der Klasse *BasisModel* testen.** Legen Sie in der story-Datei zwei Scenarien an, *Starten einer Messreihe ohne Messungen* und *Starten einer Messreihe mit Messungen.*

*Messreihen ansehen* wird hier getestet, indem das Lesen von Mess-  
reihen getestet wird, hier die Methode *leseMess-reihenInklusiveMessungenAusDb* der Klasse *Basis-Model.*

*Messreihe ausgesucht* Implementieren Sie die Suche einer Messreihe  
ohne Messungen für das erste Szenario und mit  
 Messungen für das zweite Szenario innerhalb der  
 Step-Klasse.

*falls Funktion zum Start einer Messreihe ausgewählt wird*

bedeutet, dass die Methode *starteMessreihe* der Klasse *BasisControl* angestoßen wird. Das können Sie weglassen.

*Messreihe starten* wird hier getestet, indem der Start einer Messreihe   
 und damit das Aufnehmen von Messungen getestet  
 wird, hier die Methode *speichereMessungInDb* der  
 Klasse *BasisModel*.

Im Szenario *Starten einer Messreihe ohne Messungen* sollen 0, 1 oder 2 Messungen angelegt werden. Die anzulegenden Messungen können Sie mit beliebigen Messwerten erstellen (Dummy-Objekte, siehe Bemerkungen), ohne auf das Messgerät zuzugreifen. Benutzen Sie eine JBehave-Testmethode inklusive Variable und Examples. Überprüfen Sie, ob die richtige Anzahl von Messungen angelegt wurde.

Im Szenario *Starten einer Messreihe mit Messungen* soll versucht werden, eine Messung anzulegen. Es kommt zu einer Exeption, die Sie überprüfen sollen.

Ausnahmsweise dürfen Sie die Klasse *BasisModel* erweitern.

**public** Messreihe getMessreiheByMessreihenId(**int** messreihenId) {

Messreihe ergebnis = **null**;

**int** pos = 0;

**while**(ergebnis == **null**

&& pos < **this**.getMessreihen().size()) {

**if**(**this**.getMessreihen().get(pos).getMessreihenId()

== messreihenId) {

ergebnis = **this**.getMessreihen().get(pos);

}

pos++;

}

**return** ergebnis;

}

**Bemerkungen**

Üblicherweise müssen Datensätze, die zum Testen benötigt werden, hier die Messreihen ohne und mit Messungen, beim Testen zu Beginn selbst erstellt werden, damit sicher ist, dass sie in der Datenbank existieren. Sie müssen am Ende inklusive Messungen wieder gelöscht werden, da die Datenbank mit den Testdaten nach dem Test in demselben Zustand sein sollte wie vor dem Test.

In dieser Aufgabe läuft der Test vielleicht nur 1x erfolgreich und vollständig durch, da beim 2. Durchlauf keine Messreihe mehr ohne Messungen in der Datenbank vorhanden ist. Hier müssten nach jedem Testlauf gespeicherte Messungen wieder gelöscht werden.

Die Messung, die neu erstellt wird, ohne einen Messwert mittels Messgerät zu holen, ist ein einfaches Beispiel zu einem Dummy-Objekt (vgl. Mocking).