

Übung 09: REST-Server und REST-Client

Abgabetermin: 13. 6. 2022, 8:30

Aufgabe	Punkte	Abzugeben über Moodle
Übung 09	24	Java-Programm in Zip-Datei

Übung 08: HTML-Server mit REST (24 Points)

Ihr Webinterface ist auf breite Zustimmung gestoßen und wird bereits an vielen Instituten der Johannes Kepler Universität¹ erfolgreich verwendet. Von diesem allgemeinen Zuspruch sehr beeindruckt, haben sich nun auch Rektorat und Institute auf die Einrichtung eines gemeinsamen Servers für einen universitätsweiten Kalender geeinigt. Während man dafür zwar Ihre zuvor entwickelte Datenbank und Ihre JavaFX-GUI weiterverwenden möchte, möchte man sich aber durch eine plattformunabhängige Schnittstelle für die Zukunft absichern. Natürlich erhalten ein weiteres Mal Sie den Auftrag für die Implementierung dieser neuen REST-basierten Schnittstelle zwischen Kalender-Datenmodell und Kalender-GUI.

Auf Moodle ist eine aktualisierte Version der *pom.xml* für das Framework aus UE08 verfügbar. Nachdem Sie den Inhalt dieser Datei mit der neuen Versionen ersetzt haben, laden Sie bitte das Maven Projekt neu (*pom.xml* -> *Maven* -> *Reload Project*), um die hinzugekommenen Bibliotheken zu importieren.

Erweitern Sie das Framework aus UE08 um eine REST-Schnittstelle für das Datenmodell und schließen Sie die Kalender-GUI über diese an das Datenmodell am Server an. Konkret ist folgendes nötig:

- Implementieren Sie eine Klasse *CalendarServlet*, in der Sie Ihre REST-Schnittstelle implementieren. Das Design der Schnittstelle ist Ihnen überlassen. Sie können in Ihrer REST-Schnittstelle die Methoden *addListener*, *removeListener* und *close* aus *CalendarModel* ignorieren. Erstellen Sie weiterhin geeignete *ExceptionHandler* Implementierungen für *DateTimeException* (geworfen z.B. in *DateTime.of(...)* oder *YearMonth.of(...)*), *NoSuchTaskException* und *CalendarException*.
- Übertragen Sie in Ihrer REST-Schnittstelle *Task* und die diversen Sammlungen, in denen es enthalten ist, in Form von JSON. Modifizieren Sie dazu *Task* soweit nötig.
- Erstellen Sie weiterhin eine Klasse *CalendarClient*, welche das *CalendarModel* Interface implementiert und Methodenaufrufe über Ihre REST-Schnittstelle an den Server weiterleitet. *CalendarClient* soll hinzugefügte *CalendarListener* selbst speichern und bei Ausführung seiner Methoden entsprechend benachrichtigen. Es reicht insofern, wenn in der GUI eines Clients durch andere Clients durchgeführte Änderungen erst durch einen Monatswechsel nachgeladen und dadurch ersichtlich werden.

Sie dürfen jeden Teil des zur Verfügung gestellten Codes verändern.

Hinweise:

- Achten Sie sowohl am Server als auch am Client auf geeignete Fehlerbehandlung.
- Achten Sie auf die Verwendung geeigneter HTTP-Statuscodes und deren Prüfung.
- Sie müssen sich nicht darum kümmern, das Modell-Objekt am Server freizugeben und den Server zu terminieren. Der Server kann endlos laufen, bis er manuell terminiert wird.
- Achten Sie auf eine saubere Implementierung.

¹ Disclaimer: Die Johannes Kepler Universität ist eine fiktive Hochschule, die in keinem Zusammenhang zur Johannes Kepler Universität steht.