

Dynamisches Web - Übung 4

17. Dezember 2016

Inhalt der Übung

- Übungsaufgabe
 - express.js HelloWorld
 - Router in ein node.js Modul auslagern
 - Vorstellung Anwendungsgerüst
 - Beginn Umsetzung Übung

Übungsaufgabe

Abgabeschluss: **Sonntag, 8. Jänner um 22:00 Uhr.**

Bitte alle Dateien zu einem ZIP-Archiv zusammenfügen und in Moodle mit dem Dateinamen-Muster `ue4-<ihrnachname>.zip` hochladen. Verwenden Sie wie in den bisherigen Übungen das Port 3000. Der vorgezeigte Beispielcode wird auf der E-Learning Plattform zur Verfügung gestellt.

Generelle Aufgabenstellung:

Ziel der vierten Übung ist die Umsetzung des AboutMe Klons aus Übung 3 (allerdings ohne Caching) mit dem node.js basierten Webframework Express, erreichbar unter <http://expressjs.com>. Express übernimmt viele allgemeine Aufgaben rund um HTTP und soll so die Arbeit eines Webentwicklers erleichtern und damit beschleunigen.

Hinweis: Caching wird deshalb nicht noch einmal implementiert, weil Express automatisch ETags generiert. Wie der Last-Modified Response Header, ist der ETag Header Teil des Caching Validierungsmodells.

Aufgaben:

1. Umsetzung Punkte 1, 2 und 3 aus Übung 3 (8P)

- Verwenden Sie das zur Verfügung gestellte Anwendungsgerüst, um die Angaben aus den Unterpunkten 1, 2 und 3 aus Übung 3 umzusetzen.
- **Wichtig:** Beachten Sie die leichten Änderungen im URI-Schema der Applikation. Statt <http://localhost:3000/<kürzel>> ist gemäß dem Anwendungsgerüst nun <http://localhost:3000/profiles/<kürzel>> usw. zu verwenden.
- Ein entscheidender Faktor bei der **Benotung** der Übung ist nicht nur die Erfüllung der funktionalen Anforderungen gemäß Übungsblatt 3, sondern insbesondere die richtige Strukturierung nach dem MVC Pattern.

2. Umsetzung Punkt 5 aus Übung 3 – Profilbild Upload (1P)

- Ermöglichen Sie dem Benutzer so wie in Übung 3 den Upload eines Profilbilds, das selbstverständlich auch in der Ansicht des Profils angezeigt werden soll.

- Mit einer Middleware-Funktion erleichtert Express das Ausliefern statischer Ressourcen wie Bilder enorm. Die Vorgabe ist, dass Sie diese Funktion benutzen. Mehr Infos dazu sind in der Dokumentation von express.js zu finden.

3. Liste aller Profile (1P)

- Als einzige neue Anforderung kommt hinzu, dass unter der URI <http://localhost:3000/profiles> alle am Server angelegten Profile aufgelistet werden sollen. Zur Erinnerung: man nennt eine solche Ressource auch Listenressource (vgl. Block 2).
- Natürlich ist die reine Auflistung von Profilnamen nur bedingt nutzenbringend. Sorgen Sie dafür, dass Besucher der Seite über die Liste zu den einzelnen Profilen gelangen.