

Komponentenbasierte Entwicklung SoSe 2015

Übungsblatt 1: Entwicklungsumgebung/JSF

Bearbeitung in der Übung KW 15/16

Für die Bearbeitung der Aufgaben ist eine Entwicklungsumgebung (Betriebssystem, IDE etc.) festgelegt. Sie sollten sich weitgehend daran halten. Die Festlegung der Umgebung soll ermöglichen, dass wir am Thema bleiben können und nicht ständig mit Plattformspezifika „kämpfen“ müssen. Sollten Sie eine abweichende Umgebung verwenden, kann nur eingeschränkt Support gegeben werden.

Bearbeiten Sie alle Aufgaben immer in Paaren (Ausnahmen sind markiert). Suchen Sie sich hierfür einen festen Übungspartner. Wechseln Sie sich an der Tastatur ab.

Falls Sie in der Übung nicht fertig werden, empfiehlt es sich, die Aufgaben selbstständig bis zur nächsten Woche fertigzustellen. Auch dies sollten Sie in Paaren tun.

Aufgabe 1-1 (Entwicklungsumgebung)

Vorbemerkung:

- Setzen Sie sich als Paar an zwei benachbarte Rechner und bearbeiten Sie die Teilaufgaben einzeln.
- Fangen Sie nur dann mit einer neuen Teilaufgabe an, wenn beide mit der vorhergehenden Aufgabe fertig sind.
- Gleichen Sie sich nach jeder Teilaufgabe ab: Was haben Sie gelernt? Sind noch Fragen offen? Wenn ja, klären Sie diese gemeinsam.

Ziel: Vertraut machen mit der Entwicklungsumgebung auf den Laborrechnern und ggf. Einrichten einer Entwicklungsumgebung auf dem eigenen Rechner.

Im Rahmen der Veranstaltung werden wir unter der folgenden Umgebung arbeiten:

- a) **Betriebssystem:** Ubuntu 14.04.x
- b) **Java:** Java SE Development Kit 8 von Oracle
- c) **Maven** (aktuelle Version): Wofür wir das brauchen, wird später erläutert.
- d) **Entwicklungsumgebung:** Eclipse IDE for Java EE Developers in der Version Luna (Luna Service Release 1a (4.4.1))
- e) **Servlet Engine:** Apache Tomcat 7
- f) **Datenbank:** Hier können Sie verwenden, was Ihnen zusagt, z.B. eine In-Memory-Datenbank wie Apache Derby.

Machen Sie sich mit der Umgebung auf dem Poolrechner vertraut und beantworten Sie folgende Fragen bzw. bearbeiten folgende Teilaufgaben:

1. Ermitteln Sie die installierte Java-Version und wo die Runtime und der Compiler installiert sind.
2. Ermitteln Sie die installierte Maven-Version und wo Maven installiert ist.
3. Was ist ein Tomcat-Server?:
 - a. Lesen Sie nach: http://de.wikipedia.org/wiki/Apache_Tomcat
 - b. Ermitteln Sie:

- i. Wie lautet (unter Ubuntu) der Pfad zu den Skripten zum Starten (`startup`) und Stoppen (`shutdown`) des Servers.
 - ii. Wo findet man (unter Ubuntu) das Verzeichnis in dem die Webapplikationen, die der Tomcat bereitstellen soll, liegen (heißt `webapps`)?
 - iii. Wo findet man (unter Ubuntu) das Verzeichnis, in das der Tomcat Logging-Informationen schreibt?
 - c. Lesen Sie unter <http://tomcat.apache.org/tomcat-7.0-doc/appdev/deployment.html> nach, in welcher Weise Applikationen auf einem Tomcat-Server installiert werden (sollten Sie z.T. schon aus der Vorlesung wissen).
 - d. Private Tomcat-Instanz einrichten: Tomcat wird häufig im Bereich Entwicklung und Test verwendet, wenn viele Benutzer auf einem System parallel arbeiten. Das Ubuntu-Tomcat-Paket stellt ein Werkzeug zur Verfügung, Einzelnutzer-Instanzen von Tomcat zu installieren, die auf die zentralen System-Bibliotheken zurückgreifen, aber keine root-Rechte benötigen. Somit kann jeder Benutzer eine private Instanz von Tomcat verwenden. Folgen Sie hierzu dem Abschnitt „Private Instanzen verwenden“ unter <https://help.ubuntu.com/lts/serverguide/tomcat.html> (das Paket `tomcat7-user` ist bereits auf Ihrem Rechner installiert). Für Informationen zum Tool `tomcat7-instance-create` siehe auch <http://manpages.ubuntu.com/manpages/quantal/man2/tomcat7-instance-create.2.html>. Die private Instanz sollte in Ihrem Heimatbereich liegen.¹
 - e. Starten Sie ihre private Tomcat-Instanz. Achten Sie dabei auf die Ausgaben auf der Konsole. Rufen Sie danach im Browser <http://localhost:8080/> auf. Was sehen Sie? Fahren Sie den Tomcat nun wieder runter. Rufen Sie danach im Browser wieder <http://localhost:8080/> auf. Was sehen Sie?
4. Starten Sie Eclipse
- a. Wählen Sie eine Workspace in Ihrem Heimatbereich. By the way: Klären Sie, was ein Eclipse-Workspace eigentlich ist.
 - b. Mit der *Eclipse IDE for Java EE Developers* ist es möglich, Java-Webapplikationen während der Entwicklung direkt auf einem Server, z.B. einem Tomcat, zu testen. Wir wollen hier auf die eingerichtete private Tomcat-Instanz zugreifen. Hierzu müssen wir diese in Eclipse bekanntgeben.

Damit wir das tun können, müssen wir vorher ein paar Dinge nachholen, die `tomcat7-instance-create` ausgelassen hat. Alle Informationen hierzu finden Sie im Abschnitt „Missing files“ unter <http://askubuntu.com/questions/313103/cannot-start-tomcat-after-installing-a-private-instance>. Beachten Sie auch die Fußnote auf der Seite (beginnt mit „I just wanted to say that this answer worked great...“). Dem hier beschriebenen Hinweis müssen Sie folgen. Alle anderen Abschnitte auf dieser Seite (z.B. „New server bug“ sollten Sie erst dann beachten, wenn die nächsten Schritte nicht zum Erfolg führen).²

¹ Wechseln Sie vor dem Ausführen des Kommandos in das Zielverzeichnis in Ihrem Heimatbereich.

² Achten Sie darauf, immer den Pfad zu Ihrer privaten Tomcat-Instanz zu verwenden.

Nun können wir in Eclipse unseren Tomcat-Server einrichten: Unter dem Reiter "Server" auf "Click this link to create a new server" klicken. Im sich öffnenden Dialog "Server" folgende Einträge machen:

- Unter Apache "Tomcat v7.0 Server" auswählen
- "Server name" editieren, z.B. "My private Tomcat v7.0 Server at localhost"
- "Next" wählen
- Bei "Tomcat installation directory" das Verzeichnis auswählen, in dem die private Tomcat-Instanz liegt (z.B. `~/my-tomcat`)
- "Finish" wählen

Nun können Sie den Tomcat von Eclipse aus starten: Unter dem Reiter "Server" auf "My private Tomcat v7.0 Server at localhost" (oder wie auch immer Sie Ihren Server genannt haben) das Kontextmenu öffnen und „Start“ wählen. Rufen Sie danach noch einmal probeweise `http://localhost:8080/` im Browser auf. Falls Sie eine Fehlermeldung bekommen (z.B. HTTP Status 404) ist alles ok. Sollte der Server nicht zu erreichen sein: Ein Problem existiert, um das Sie sich kümmern müssen. Als letztes: Stoppen Sie den Tomcat-Server (Kontextmenü).

Hier nun ein paar Hinweise zur Installation unserer Entwicklungsumgebung unter Ubuntu 14.04.02:

Vorbemerkung: Sie benötigen Root-Rechte.

Zu b) **Java:** Sie sollten das Java SE Development Kit 8 von Oracle installieren (Achtung: Das Java Runtime Environment (JRE) reicht nicht aus). Oracle Java (früher Sun Java), auch als Java SE (Standard Edition) bezeichnet, muss manuell – praktisch an der Paketverwaltung von Ubuntu vorbei – installiert werden (siehe auch http://wiki.ubuntuusers.de/Java/Oracle_Java). Folgen Sie hierzu der Anleitung unter http://wiki.ubuntuusers.de/Java/Installation/Oracle_Java/Java_8#Java-8-JDK.

Testen Sie Ihre Installation wie oben unter 1) beschrieben.

Zu c) **Maven:** Folgen Sie der Anleitung unter <http://www.mkyong.com/maven/how-to-install-maven-in-ubuntu/> (im Prinzip nur ein Befehl).

Testen Sie Ihre Installation wie oben unter 2) beschrieben.

Zu d) **Eclipse IDE for Java EE Developers:** Sie sollten die aktuelle Version Luna installieren. Diese finden Sie hier: <http://eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/lunasr1a>. Im Prinzip ist das Installieren nur ein Entpacken und Verschieben (für Details siehe z.B. https://www.youtube.com/watch?v=7kR7a_Z9NIQ)

Zu e) **Apache Tomcat:** Hier sollten Sie die Version 7 installieren. Beachten Sie hierzu den Abschnitt „Systemweite Installation“ unter <https://help.ubuntu.com/lts/serverguide/tomcat.html>.

Die Installation von Tomcat unter Ubuntu 14.04.02 bewirkt, dass Tomcat als Dienst eingerichtet und beim Systemstart mitgestartet wird (Details zu Diensten unter Ubuntu siehe <http://wiki.ubuntuusers.de/Dienste>). Sie sollten dies abschalten. Komfortabel geht dies mit dem Tool `sysv-rc-conf` (GUI zum Bearbeiten von Diensten; müssen Sie evtl. erst installieren:

`sudo apt-get install sysv-rc-conf`). Starten Sie das Tool und deselektieren Sie beim Tomcat die Einträge für die Runlevel 2-5.

Private Instanzen von Tomcat verwenden (optional - falls Sie es brauchen): Hierfür benötigen Sie das Tool `tomcat7-user` (Installation: `sudo apt-get install tomcat7-user`) Für alle weiteren Schritte siehe oben unter 3d.

Tomcat unter Eclipse einrichten: Siehe oben unter 4.

Zu f) In späteren Übungen benötigen Sie eine relationale Datenbank. Sie können hier frei wählen (z.B. MySQL oder Apache Derby). Eine Anleitung zur Installation von Apache Derby finden Sie unter http://db.apache.org/derby/papers/DerbyTut/install_software.html.

Aufgabe 1-2 (JSF-Applikation *Hello World*)

Vorbemerkung: Führen Sie diese Aufgabe zu zweit an einem Rechner durch.

Ziel: Eine erste einfache JSF-Applikation zum Laufen bringen und verstehen.

Hilfsmittel: JSF-Dokumentation unter <http://jsfatwork.irian.at>

Importieren Sie eine fertige Hello-World-Anwendung in Ihr Eclipse und beantworten Sie die Fragen:

1. Laden Sie die auf Moodle bereitgestellte Datei `kbe-ue1-2.zip` herunter.
2. Starten Sie Ihr Eclipse.
3. Wählen Sie *File->Import...*
4. Wählen Sie *General->Existing Projects into Workspaces* (und *Next*)
5. Wählen Sie *Select archive file* und dann das heruntergeladene Zip.
6. Wählen Sie *Finish*.
7. Warten Sie bis der Workspace-Build fertig ist (siehe in Eclipse in der rechten unteren Ecke)
8. Nun können Sie die Applikation auf Ihrem Tomcat-Server laufen lassen:
 - a. Öffnen Sie im Project Explorer von Eclipse das Kontextmenü auf dem Projekt `kbe-ue1-2`.
 - b. Wählen Sie *Run As->Run on Server*
 - c. Wählen Sie *Choose an existing Server* und Ihren Server aus der Liste darunter.
 - d. Wählen Sie *Finished*. Der Server wird gestartet. Außerdem sollte auch Firefox (bzw. der eingerichtete Default-Browser) gestartet werden und die Seite „Hello World“ erscheinen (<http://localhost:8080/kbe-ue1-2/hello.jsf>).
9. Laden Sie die Seite im Browser neu. Beobachten Sie dabei die Konsole in Eclipse. Was sehen Sie? Finden Sie heraus, warum das passiert.
10. Ändern Sie die Managed Bean in der Applikation dahin, dass Sie zum Session-Scope gehört. Rufen Sie die Seite ein weiteres mal auf. Was können Sie beobachten? Warum?
11. Ist Ihnen sonst noch etwas aufgefallen?: Es war nicht gefordert den Server neu zu starten. Was erschien auf der Konsole, nachdem Sie Ihre Änderung abgespeichert hatten?
12. By the way: Geben Sie im Browser nur `localhost:8080/kbe-ue1-2/` ein. Sie laden auf der Hello-World-Seite. Wie kommt das?

13. Für alle, die sich jetzt wundern, warum wir eigentlich keine JSF-JARs installiert haben. Schauen Sie in die Datei `pom.xml` (Hier geht es um das Buildwerkzeug Maven. Wenn Sie nicht verstehen, was hier passiert: Kein Problem. Wir werden es demnächst behandeln).

Aufgabe 1-3 (Eclipse)

Vorbemerkung: Diese Aufgabe müssen Sie nicht im Paar bearbeiten – können Sie aber.

Ziel: Eclipse kennenlernen.

Falls Eclipse für Sie neu ist oder Sie sich nicht sicher im Umgang mit Eclipse fühlen, arbeiten Sie den Abschnitt *Basic tutorial* in der Eclipse-Dokumentation unter <http://help.eclipse.org/luna/index.jsp> durch. Sie finden das *Basic tutorial* unter *Workbench User Guide->Getting Started->Basic tutorial*.

Aufgabe 1-4 (Saros)

Vorbemerkung: Diese Aufgabe müssen Sie nicht im Paar bearbeiten. Das Ergebnis sollten Sie aber im Paar testen.

Für Eclipse gibt es das Plugin Saros. Es macht es möglich, dass Sie mit einem Partner in Eclipse zusammen arbeiten können, auch wenn Sie nicht am selben Rechner oder in der selben Stadt sitzen. Unter <http://www.saros-project.org> finden Sie alle notwendigen Informationen.