Einrichtung Entwicklungsumgebung  
(Eclipse + Git)  
  
für *Intern*

vertraulich

Autor: Daniel Knauth (Scheer GmbH)

Version: 0.0.1

Datum: 2016-10-18

M&M-Werbeagentur GmbH

Saarbrücker Allee 5a

D-66663 Merzig

Tel.: +49 68 61 – 93 92 8 – 0

Fax: +49 68 61 – 93 92 8 – 19

© M&M-Werbeagentur GmbH

Unauthorised copies are prohibited. Each unauthorised use and transmission will cause claim in tort. Although this documentation was created with all conscientious care, the whole risk of consequences of use will be borne by the user himself. The authors exclude liability, guarantees, or any kind of responsibility for the use or the consequences of use as far as correctness, exactness, reliability, actuality or others are concerned.

Inhalt

Revisionsdaten 2

Abkürzungen 2

Übersicht 3

Voraussetzungen 3

Vorbereitung 3

Ersteinrichtung von Eclipse und Git-Checkout 3

Konfiguration der Build- und Laufzeitumgebung 4

Einrichtung und Test des Tomcat-Servers 5

# Revisionsdaten

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Status** | **Datum** | **Autor** | **Änderungen** |
| 0.0.2 | Update | 2016-11-03 | Daniel Knauth | Git-Einstellungen ergänzt |
| 0.0.1 | Entwurf | 2016-10-18 | Daniel Knauth | Erste Fassung |
|  |  |  |  |  |
|  |  |  |  |  |

# Abkürzungen

|  |  |
| --- | --- |
| M&M | M&M Werbeagentur GmbH |
| Eclipse | Eine quelloffene Programmierumgebung; <http://eclipse.org/> |
| Git | Verteiltes System zur Software-Versionskontrolle; <https://git-scm.com/> |
| Putty | Secure-Shell-Ersatz für Windows PCs |
| SSH | Secure Shell, Protokoll zur Fernwartung von (Unix-) Rechnern |

# Übersicht

Dieses Dokument beschreibt die Erst-Einrichtung der Entwicklungswerkzeuge auf einem PC im Netz der M&M.

# Voraussetzungen

Die Anmeldung am PC in der Domäne *mm-intranet.de* muss möglich sein, die Java-Laufzeitumgebung *JDK 1.8* muss durch den PC-Support vorinstalliert sein.

# Vorbereitung

Die folgenden Komponenten können direkt aus dem Internet heruntergeladen durch den Endanwender lokal installiert werden. Dafür sind keine Administrationsrechte notwendig.

|  |  |  |
| --- | --- | --- |
| Name | Version | Beschreibung |
| [Eclipse](http://www.eclipse.org/downloads/) | >= 4.6.0 | Java-Entwicklungsumgebung |
| [Tomcat](https://tomcat.apache.org/) | >= 8.0 | Web-Applikations-Server |
| [Putty](http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html) | >= beta 0.6.7 | Secure-Shell-Client für MS Windows |

Eclipse und Tomcat werden üblicherweise in C:\ entpackt. Putty kann an beliebiger Stelle abgelegt werden, also i.d.R. unter *C:\Program Files*.

# Ersteinrichtung von Eclipse und Git-Checkout

Als Arbeitsbereich für Git-Repositories wird ülicherweise **C:\SRV** benutzt. Dieser Ordner muss ggf. angelegt und beim ersten Start von Eclipse als *Workspace* festgelegt werden.

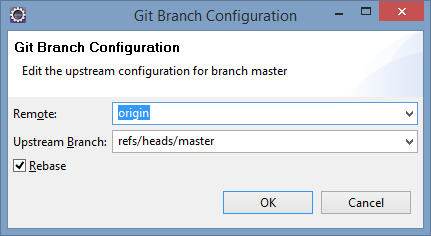
Zunächst sind in Eclipse einige Grundeinstellungen vorzunehmen, diese finden sich im Menü unter *Window 🡪 Preferences*:

1. Unter *General 🡪 Startup and Shutdown 🡪 Workspaces* ist C:\SRV einzutragen, falls nicht beim ersten Start bereits geschehen.
2. Unter *Team 🡪 Git 🡪 Configuration* wird der aktuelle Benutzer eingetragen, und zwar mit
   1. *user.name*: Aktueller Domänen-Anmeldename, z.b. *nkarpe* oder *ext0239*
   2. *user.email*: Email-Adresse @mm-worldnet.de
3. Unter *Java 🡪 Installed JREs* muss das aktuelle JDK 1.8 als Standard-VM eingetragen sein. Etwaige Vorgängerversionen können entfernt werden.

Die Quelltexte der Anwendungen werden in Git versioniert. Eine lokale Arbeitskopie kann in C:\SRV wie folgt angelegt werden:

1. Die Git Repository-Ansicht öffnen: *Window 🡪 Perspective 🡪 Open Perspective 🡪 Other… 🡪 Git*.
2. Im Bereich *Git Repositories* den Punkt  *Clone a Git repository* wählen:
   * Host: *192.168.0.80*
   * Git repository: */srv/ph\_git*
   * Protocol: *ssh*
   * User: Aktueller Domänen-Anmeldename, z.b. *nkarpe* oder *ext0239*

Für die tägliche Entwicklung wird der Branch: *master* genutzt.  
Als Zielordner für die lokale Arbeitskopie bitte *C:\SRV\ph* angeben

1. Unter *Branches* 🡪 *Local* 🡪 *Master* mit Rechtsklick *Configure Branch…* wählen und dort *Rebase* aktivieren, um unnötige Merge-Operationen zu vermeiden (siehe Bild)  
   
2. Importieren der Projekte: *File* 🡪 *Import…* 🡪 *Git* 🡪 *Projects from Git* 🡪 *Existing loal repository* 🡪 *ph\_git 🡪 Import existing Eclipse projects*
3. Teilprojekte auswählen: *A-D, R, Z, images*

# Konfiguration der Build- und Laufzeitumgebung

In einer frischen Arbeitskopie müssen zuerst die Abhängigkeiten zwischen den Teilprojekten eingerichtet werden, um einen sauberen Build in Eclipse zu erhalten. Diese sind:

A

B

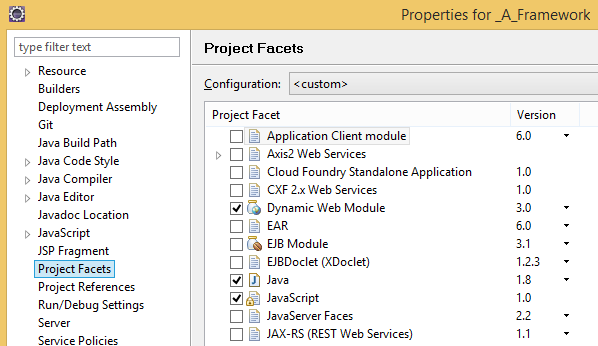
C

D

* A hängt von B ab,
* A und B hängen von C ab,
* A, B, und C hängen von D ab.

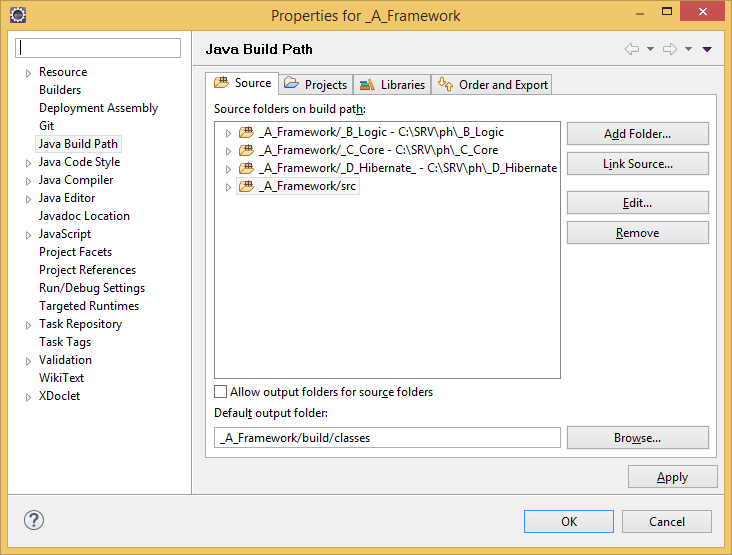
Die Laufzeitbibliotheken von *\_A\_Framework* werden korrekt gefunden, wenn folgende *Project Facets* gewählt sind:

* Java (1.8)
* JavaScript
* Dynamic Web Module



Die *Web App Libraries* von Projekt A sind als Abhängigkeiten in B, C und D bereits eingetragen.

Im Projekt *\_A\_Framework* werden die Abhängigkeiten unter *Properties 🡪 Java Build Path 🡪 Source* als Verknüpfung *(„Link Source…“)* geführt. Wenn wie oben beschrieben die lokale Arbeitskopie in *C:\SRV\ph* liegt, stimmen die Pfade bereits. Andernfalls müssen sie korrigiert werden.



Die Anwendung erwartet 2 Konfigurationsdateien in C:\SRV\conf[[1]](#footnote-1). Diese Dateien sind nicht in Git eingecheckt; Beispiele anbei.

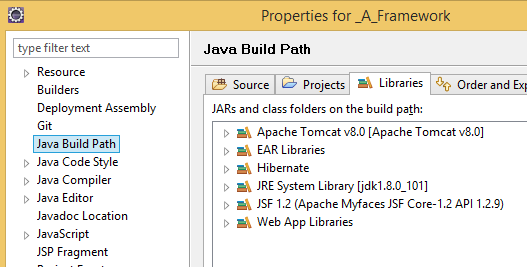
  
In environment.ini muß die aktuelle IP und der Name des Entwicklungs-Rechners in dev\_ips und dev\_hosts eingetragen werden, andernfalls startet die Anwendung nicht.

# Einrichtung und Test des Tomcat-Servers

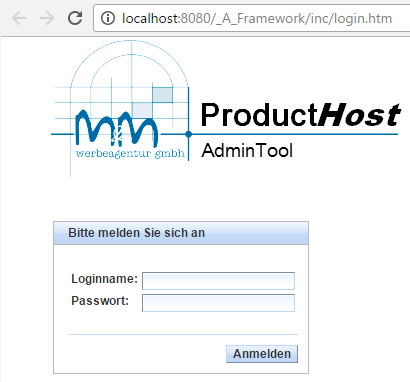
1. Im Menü *Window 🡪 Show View 🡪 Servers* aufrufen, dort den Link *No servers available…* wählen, dann *Apache 🡪 Tomcat v8.0 Server*.
2. Dann das Installationsverzeichnis auswählen, also *C:\apache-tomcat-8.0.x*
3. Im letzten Schritt wird die Anwendung *\_A\_Framework* konfiguriert.

Im Projekt *\_A\_Framework* den Server unter *Properties 🡪 Java Build Path 🡪 Libraries* mit *Add Library 🡪 Server Runtime 🡪 Apache Tomcat v8.0* eintragen.

Eventuell bestehende Abhängigkeiten auf frühere Tomcat-Versionen können entfernt werden.



Nach Start von Tomcat in der *Servers-View* kann die Anwendung unter <http://localhost:8080/_A_Framework/inc/login.htm> aufgerufen werden.



1. Die Pfade für die Konfigurationsdateien sind hart codiert in \_D\_Hibernate\_/com/pinhost/app/lib/config/Environment.java [↑](#footnote-ref-1)