

**TFH-Berlin  
Fachbereich VI  
Seminar: SWP2**

**Dozent: Prof. Knabe**

# **Instant Messenger-System**

Schlussbewertung

**Melanie Hirsch, s726183  
Christoph Goerlich, s729964**

**Datum: 10.07.2008**



## **Inhaltsverzeichnis**

<b>1</b>	<b>Schlussbewertung</b> .....	<b>1</b>
1.1	Schlussbewertung der Entwicklung des Servers: .....	1
1.2	Schlussbewertung der Entwicklung des Clients .....	1



## **1 Schlussbewertung**

### **1.1 Schlussbewertung der Entwicklung des Servers:**

Das Teil-Projekt Server ging insgesamt über die Dauer von 2 Semestern und ist insgesamt als Erfolg zu betrachten. Alle Muss-Kriterien wurde erfolgreich umgesetzt.

Die Entwicklung nach dem Wasserfallmodell war nicht optimal und hatte zu lange gedauert. Gerade das ständige Pflegen und Überarbeiten der Diagramme ist nicht vorteilhaft, gerade weil manche Änderungen erst während der Entwicklung notwendig wurden.

Bei größeren Projekten mag das Vorgehen nach dem Wasserfallmodell seine Berechtigung haben. In unserem Fall hätte man sicherlich schneller zu Erfolg kommen können.

Die Entwicklung mit Ruby on Rails hat Spaß gemacht und war sehr erfolgreich. Die Möglichkeiten verschiedenen Paradigmen der Programmierung einsetzen zu können ist ein enormer Vorteil zu anderen Sprachen. Auch die Verwendung des Webframeworks Rails, war optimal. Mit Rails lassen sich hochwertige Webapplikationen sehr schnell umsetzen.

### **1.2 Schlussbewertung der Entwicklung des Clients**

Das Teil-Projekt Client ging insgesamt über die Dauer von 2 Semestern und ist insgesamt als Erfolg zu betrachten. Alle Muss-Kriterien wurde erfolgreich umgesetzt.

Die Entwicklung mit Java war langwierig und doch ist das Projektende als Erfolg zu werten, da auch in diesem Bereich alle Muss-Kriterien zum Abgabedatum erfüllt wurden.

Es ist zu überlegen, ob eine Vorgehensweise nach Agilen Methoden evtl. schneller zum Erfolg geführt hätte.

Leider war die Wahl der Basistechniken für den Client, nachträglich betrachtet, nicht optimal und führte zu Komplikationen und letztlich auch zu Verzögerungen während der Implementierung. So ist das GUI-Toolkit SWT ausserhalb des Eclipse-RCP-Frameworks relativ unhandlich und unflexibel, z.b. was den Umgang mit Threads betrifft. Auch die ursprünglich geplante Nutzung eines Messenger-

---

Protokolls, etwa Jabber, war nicht möglich, da alle Instant-Messenger-Protokolle stark Serverausgerichtet sind und wir damit die Kriterien des Pflichtenheftes, direkte Client-Client-Kommunikation, verletzt hätten.