

Schlussbewertung „eFit“

Allgemeines

Da der Zeitrahmen in diesem Semester sehr eng war, konnte das Team eFit die Kann-Kriterien nur teilweise wie z.B. die Darstellung von Diagrammen realisieren. Auf der Strecke ist dabei die Implementierung für Mobilgeräte geblieben.

Endabgabe

Aufteilung der Klassen bei der Endabgabe ist suboptimal. Viele Komponenten sind in Teamarbeit entstanden, so dass eine eindeutige Zuordnung nicht möglich ist.

Vorlesung

Themen wie AspectJ hätten früher angesprochen werden müssen, evtl. sogar im ersten Semester des Softwareprojekts, da es eine sinnvolle Erweiterung darstellt und unsere Softwarearchitektur im Gesamtkontext vereinfacht hätte. Auch das Testen wurde zwar von der technischen Seite detailliert erläutert, doch blieben anfangs viele Fragen offen, die allerdings alle in der Besprechung geklärt werden konnten.

Technologien

Auf Grund der Zusammengehörigkeit der Produkte für die Entwicklung, sind wir davon ausgegangen, dass diese einwandfrei zusammenarbeiten. Wir mussten feststellen, dass es trotzdem zu Unstimmigkeiten der Techniken gekommen ist.

IDE

NetBeans lief teilweise sehr instabil. So mußten wir während des Projekts die IDE mehrmals neu installieren und nach dem Hinzufügen von externen Bibliotheken liefen Komponenten der IDE nicht mehr. Allerdings war NetBeans eine große Hilfe bei der Erstellung von JSF basierten Masken

Server

GlassFish Enterprise Server v2.1 lief immer stabil und ist als ApplicationServer nur zu empfehlen.

APIs

VisualJSF war problemlos zu benutzen, insb. in Verbindung mit NetBeans. JFreeChart war auch einfach zu integrieren. Multex war eine anfangs eine kleine Hürde, nach einer Explorationsphase aber sinnvoll einzusetzen. Mit JUnit gab es Probleme in der Persistenzschicht, da der Applikationsserver via JNDI auf die Entities im Session-Context zugreift, eine externe J2SE basierte

Testanwendung darauf aber keinen Zugriff hatte. Als Workaround wurde deswegen eine JPA Lösung implementiert.

SVN

Subversion war für unsere Arbeit unerlässlich, allerdings muß man etwas Zeit aufbringen, um die Funktionalität sicher zu beherrschen.

Zusammenfassung

Wir würden nicht noch einmal mit NetBeans arbeiten, höchstens um die JSF basierten Masken zu erzeugen. Würden wir das Projekt noch einmal machen, würden wir von Anfang an ein Bugtrackingsystem benutzen um Fehler zu dokumentieren und permanent sichtbar zu haben.

Alles in Allem haben wir sehr viel gelernt und wir haben mehr erreicht, als wir uns im Laufe des Semesters vorstellen konnten. Es sind nach wie vor einige Inkonsistenzen und Fehler im System, doch das Gesamtsystem läuft stabil und der Endanwender sieht nichts von den Fehlern.