

## Projektphase

Die Gesellschaft für Informatik, kurz GI, erwartet von Absolventen/innen eines Bachelorstudiums der Informatik verschiedene Kompetenzen, unter anderem Analyse-Kompetenz: Informatiker/-innen müssen sich schnell in neue Anwendungen einarbeiten können und Probleme im Gesamtzusammenhang erkennen können und mit den zugehörigen Lösungsmustern vertraut sein. Dabei müssen sie Inkonsistenzen erkennen und mit unklaren Anforderungen umgehen können. Sie müssen Zerlegungsstrategien anwenden, um komplexe Domänen modellieren zu können und in der Lage sein, Schnittstellen so zu definieren, dass die Systeme wartbar, erweiterbar und zuverlässig sind.

Außerdem erwartet man von Informatikern und Informatikerinnen die Fähigkeit zur Konstruktion von Systemen aus Hard- und Software, welche die Anforderungen vollständig erfüllen. Um sich dabei nicht im Detail zu verlieren, ist die Fähigkeit zur Abstraktion genauso unverzichtbar wie solide Kenntnisse im Bereich Software-Architektur und über den Einsatz von Mustern und Bibliotheken. Wichtig sind außerdem Kenntnisse, wie Mensch-Maschine-Schnittstellen anwendungsgerecht und ergonomisch modelliert werden.

Es werden noch viele andere Kompetenzen erwartet, die hier nicht alle aufgezählt werden sollen. Um solche Kenntnisse zu erwerben, soll in dieser Veranstaltung die Übung in Form eines Projektes durchgeführt werden.

## 1 Lernziele

Ziel der Projektphase ist es, die in der Vorlesung erworbenen Kenntnisse zu vertiefen und praktisch anzuwenden. Außerdem sollen Sie Ihre Erfahrungen im gemeinsamen Erstellen von Software erweitern: Man einigt sich auf eine Schnittstelle und jeder Beteiligte löst unabhängig vom Anderen einen Teil der Aufgabe. Zur Software-Entwicklung gehört auch das systematische Testen und eine zielgerichtete Fehlersuche: Sind die einzelnen Komponenten fehlerhaft oder arbeiten sie nur nicht korrekt zusammen. Es sind daher zunächst Komponententests zu erstellen, und erst danach ist die Software als Ganzes zu testen.

Alle Projekte müssen Ausfallsicher implementiert sein. Dazu ist an allen Stellen Redundanz vorzusehen, sowohl bei den Daten als auch bei den Prozessen. Server, Balancer oder auch Namensdienst müssen mehrfach vorhanden sein und die Replikatе müssen konsistent gehalten werden. Außerdem sind die Server als parallele Server zu implementieren: Mehrere gleichzeitige Anfragen werden parallel bearbeitet, nicht nacheinander.

## 2 Projektvorschläge

Die folgenden Projektvorschläge sind nur als Angebot zu verstehen. Sie können in Ihrer Gruppe auch eigene Vorschläge diskutieren und umsetzen. Eine Projektgruppe besteht mindestens aus zwei, maximal aus vier Studierenden.

### 2.1 Terminkalender

In einem zentral verwalteten Kalender können Termine eingetragen und abgefragt werden. Zu jedem Termin können Tagesordnungen abgelegt werden, die aber nur von den dazu Berechtigten eingesehen werden können. Die Administratoren können neue Benutzer anlegen. Die Benutzer können Gruppen bilden, wobei jeder Benutzer mehreren Gruppen angehören kann. Die Termine werden Gruppen oder Benutzern zugeordnet.

### 2.2 Dateidienst

Dateien sollen zentral verwaltet werden. Der Administrator kann neue Benutzer anlegen, die Benutzer können Gruppen bilden. Die Dateien werden Gruppen oder Benutzern zugeordnet, so dass nicht jeder alle Dateien sehen kann, sondern nur die dazu Berechtigten. Schreib- und Leseberechtigung sind zu unterscheiden: Der Server erlaubt gleichzeitigen Lesezugriff auf Dateien, aber ein Schreibzugriff wird exklusiv gewährt.

### 2.3 Spiele

Implementieren Sie ein Spiel wie Vier Gewinnt, Othello oder Dame, das zentral auf einem Server gespielt werden kann. Neue Benutzer können sich beim System registrieren; jeder Benutzer ist einer Gruppe zugeordnet. Die Einteilung in Gruppen erfolgt automatisch anhand der Spielstärke: Je höher der Prozentsatz an gewonnenen Partien ist, umso höher der Rang.

Für jede Spielklasse wird eine Liste aller augenblicklich gespielten Partien geführt. Bei der Anmeldung an das System werden diese Listen angezeigt. Jeder Benutzer kann ein Spiel anmelden und auf einen Partner warten oder man kann einem Spiel als Zuschauer beiwohnen. Schließlich kann man einem noch nicht gestartetem Spiel als Gegenspieler beitreten.

### 2.4 Instant Messenger

Das Grundprinzip von Instant Messaging ist die sofortige Übermittlung von Nachrichten. Jeder Benutzer pflegt eine Buddy-Liste (zu Deutsch Kontaktliste). Bei der Anmeldung ans System öffnet sich die Buddy-Liste und es ist mit einem Blick ersichtlich, welcher Freund online ist und für Gespräche bereit steht. Diese Liste wird natürlich stets aktualisiert.

Für unsere Zwecke ist es völlig ausreichend, lediglich kurze Textnachrichten von User zu User zu versenden. Durch die Buddy-Liste entscheiden die Nutzer selbst, wem sie das Senden von Nachrichten gestatten bzw. für wen sie sichtbar sind oder nicht.

### 3 Literatur

- Günter Bengel: *Verteilte Systeme*. Vieweg Verlag.
- G. Coulouris, J. Dollimore, T. Kindberg: *Verteilte Systeme*. Pearson Studium.
- Ulrike Hammerschall: *Verteilte Systeme und Anwendungen*. Pearson Studium.
- A. Schill, T. Springer: *Verteilte Systeme*. Springer Verlag.
- A.S. Tanenbaum, M. van Steen: *Verteilte Systeme*. Pearson Studium.
- Andrew S. Tanenbaum: *Moderne Betriebssysteme*. Pearson Studium.