Internet-Learning
Definitionsphase

Marco Rademacher

16. Mai 2000

Zusammenfassung

Nach der Planungsphase setzt in der Softwareentwicklung die Definitionsphase ein. Hier werden Ziele definiert, das System analysiert und abgegrenzt.

1 Überblick zur Definitionsphase

Während grundsätzliche Planungsarbeit (Kundenbeziehung, Entwicklungsmodell, Zeitplan, Team- und Ressourcenstrukturierung) in der Planungsphase abgelaufen sind, wird in der Definitionsphase die Basis für die Systementwicklung gelegt. Hier sind Informationen über den derzeitigen Zustand des Systems und die Ziele des Kunden zu beschaffen und Entscheidungen über den Umfang der Arbeit zu treffen.

Insgesamt kann man folgende Teilschritte ausmachen:

- Systembeschreibung
- Zielbildung
- Erneute Systemabgrenzung
- Systemmodellierung (Abläufe, Objekte)

1.1 Systembeschreibung

Die zu erstellende Software wird meist in einem bestimmten, bereits bestehendem System eingesetzt. Damit sie mit dem System harmoniert und dies hinsichtlich der Ziele des Kunden optimiert, ist das System zunächst zu beschreiben.

Beschrieben werden können z.B.:

- Aufbauorganisation (bestimmte Bereiche einer Firma, die Hierarchie)
- Ablauforganisation (wiederkehrende Arbeitsabläufe)
- Probleme (bei der bisherigen Abwicklung)
- Ziele (zur Behebung der Probleme)

1.2 Zielbildung

Der Kunde verfolgt mit dem Einsatz eines neuen IT-Systems bestimmte Ziele. Diese sind zu identifizieren und zu verfeiniern. Es entsteht eine Zielhierarchie, an deren unterster Stufe operationale Ziele stehen. Dies sind Ziele, die unmittelbar aussagen, welcher Teil des Systems verbessert werden muss.

Beispiel: Das Hauptziel einer Firma ist der Erhalt bzw. die Erhöhug ihres Marktanteils. Dieses Hauptziel kann auf verschiedene Arten erreicht werden, z.B. durch die Erhöhung des Umsatzes. Der Umsatz wiederum kann z.B. durch bessere Kalkulation oder Kostensenkung erreicht werden. Bessere Kalkulation kann dann schließlich als operationales Ziel in der Gestaltung einer übersichtlicheren Oberfläche bestehen.

1.3 Erneute Systemabgrenzung

Durch die Beschreibung des bestehenden Systems ist es bereits eingegrenzt worden, da es unmöglich ist, ein reales System vollständig zu beschreiben. Nun muss jedoch entschieden werden, auf welchen Teil sich das neu zu entwickelnde IT-System beschränken soll. Schließlich ist die Analysearbeit kein Selbstzweck, sondern soll als Basis der Systementwicklung dienen.

1.4 Objektorientierte Modellierung

Der abgegrenzte Bereich des Systems wird nun modelliert. Betrachtet werden hier diejenigen "Dingeöder Personen, die sich in der Software wiederfinden sollen. Ist z.B. die Eingabe einer Nutzerkennung erforderlich, wird im objektorientierten Modell (OOM) auch ein Objekt »Person« geschaffen, das die Eigenschaften »Nutzername« und »Passwort« erhält, da diese Daten im Programm später benutzt werden.

Diese Modellierung erleichtert die Programmentwicklung enorm, da nun bereits der Modellierungsbereich in Objekte geteilt wird, die in objektorientierten Programmiersprachen (wie z.B. Java) einfach übernommen werden können. Jedes Objekt wird später als eine neue Objektklasse in Java eingeführt.

In der Modellierung kann man noch mehr objektorientierte Konzepte einfließen lassen. So können Vererbungshierarchien modelliert werden, wenn sich mehrere Objektklassen ähnlich sehen. Sollen z.B. Schüler und Lehrer in unterschiedlichen Objekten modelliert werden, weil sie unterschiedliche Tätigkeiten durchführen, so haben doch beide Gruppen Namen. Dies kann dadurch modelliert werden, dass eine Klasse »Person« modelliert wird, die als Attribut den Namen enthält. Aus dieser Klasse werden nun die Klassen »Lehrer« und »Schüler« abgeleitet. Diese Art der Beziehung zwischen Vater- und Kind-Klassen wird auch als »is-a« oder »ist-ein«-Beziehung genannt. Den Mechanismus, dass die Kindklasse dieselber Eingenschaften enthält, wie die Vaterklasse, wird dann als Vererbung bezeichnet.[1] Vererbung

[1] Mehr zur OOM in [2].

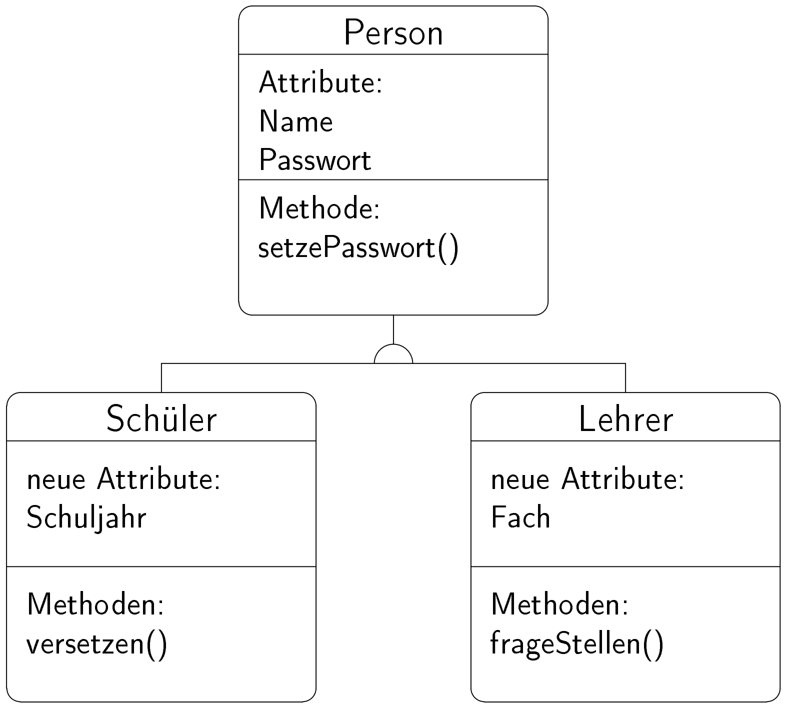

Abbildung 1: Objekte mit einfacher Vererbungshierarchie

Aufgaben

1. Welche wichtigen Fragen sind aus den vorangegangenen Aufgaben übrig geblieben? Welche Beantwortung erscheint nun doch wichtig?

2. Beschreibt das vorhandene System. Geht dabei insbesondere ein auf

   - die Technik an der Schule
   - den Kenntnisstand der Schüler in Bezug auf Internet/Computer
   - die Erwartungen der Schüler an Lernsoftware
   - die Erwartungen der Schüler und des Lehrers an die Wunschsoftware

   Tragt Eure eigenen Beobachtungen zusammen und bezieht auch den Fragebogen und die E-Mail-Nachricht von Herrn Lischka mit ein.

3. Welche Konsequenzen hat die erste Analyse für die weitere Projektplanung? Überarbeitet und konkretisiert den Projektablauf (Info: die Semesterferien beginnen am Sa, den 22.7., die Schulferien am Donnerstag, den 20.7. Es stehen also noch 9 Wochen zu Verfügung.)

4. Führt eine Zielanalyse durch. Welche Ziele könnt Ihr auf der Seite des Kunden erkennen? Brecht die Hauptziele bis zu operationalen Zielen herunter und stellt sie in einem Grafen dar.

5. Trefft Euch zu einer Brainstormingsitzung. Erörtert die Frage: Wie kann die von Euch gestaltete Software aus Anwendersicht aussehen (Funktionalität steht im Vordergrund)? Wie immer beim Brainstorming kommt es *nicht* auf *gute* Ideen, sonder auf *viele* Ideen an. Daher solltet Ihr Euch und Eure Vorschläge für 20 Minuten nicht kritisieren, sondern Euch auch von den abwegigen Vorschlägen der anderen inspirieren lassen. Einer von Euch schreibt alle Ideen mit. Nach dem Brainstorming werdet Ihr sachlicher und überlegt, was man wirklich machen kann.

6. Findet eine kurze Formel zur Bezeichnung der Software, die Ihr entwickeln wollt. Auch ein netter, kurzer Name kann Euch eine Orientierung dafür geben, was das Ziel der Entwicklung ist.

7. Grenzt den Modellbereich ein. Was soll bei der Software im folgenden nur noch betrachtet werden?

8. Versucht erste Objektklassen aus dem Modellbereich zu identifizieren und stellt sie und die Beziehungen zwischen ihnen grafisch oder tabellarisch dar.

Literatur

[1] Helmut Balzert
Lehrbuch der Software-Technik
Spektrum Akademischer Verlag, Heidelberg, 1998

[2] Martin Schader, Michael Rundshagen
Objektorientierte Systemanalyse – Eine Einführung
Springer-Verlag, Mannheim, 1994

[3] David Siegel
Das Geheimnis erfolgreicher Websites – Business, Budget, Manpower, Lizenzen, Design
Markt und Technik Verlag, München, 1998

[4] Don Tapscott
Net Kids – Die digitale Generation erobert Wirtschaft und Gesellschaft
Gabler, Wiesbaden, 1998

In den USA ist die Netznutzung weiter fortgeschritten, als bei uns. Aus einer Analyse der Nutzungsgewohnheiten von Kindern und Jugendlichen wird gefolgert, welche Einstellungen und Erwartungen eine durch das Internet geprägte Generation hat und wie sie dadurch die Gesellschaft verändern wird.

Ein durchaus visionäres, aber leicht zu lesendes Buch, das etwas Licht auf die Anwenderseite wirft. Der Originaltitel lautet »Growing up Digital«. Außerdem gibt es eine Website zum Buch mit einigen Buchausschnitten und Hintergrundinfos:
http://www.growingupdigital.com