Die Softwaretechnik-Ausbildung in der Wirtschaftsinformatik an der WWU Münster

Herbert Kuchen Michael Nietsch

Westfälische Wilhelms-Universität Münster
Institut für Wirtschaftsinformatik
http://www-wi.uni-muenster.de

1 Einführung

An der Westfälischen Wilhelms-Universität Münster kann Wirtschaftsinformatik mit den Abschlüssen Bachelor und Master bzw. Diplom studiert werden. In allen diesen Studiengängen bildet die Softwaretechnik einen zentralen Bestandteil. Neben der Vermittlung theoretischer Kenntnisse wird deren praktischer Umsetzung und insbesondere der Verwendung von CASE-Tools ein hoher Stellenwert beigemessen. Die Ausbildung umfasst im wesentlichen die folgenden Veranstaltungen. Sie sind in allen genannten Studiengängen enthalten.

Grundstudium:

1) Programmierung
2) Software-Praktikum

Hauptstudium:

1) Software-Engineering I
2) Software-Engineering II
3) Projektseminar
4) Seminare zu ausgewählten Aspekten des Software-Engineering

Auch in einer Reihe anderer Veranstaltungen werden Softwaretechnik-Aspekte gestreift. Zu nennen sind hier die Vorlesungen Nicht-Standard-Datenbanken, Wissensbasierte Systeme, Inner- und Zwischenbetriebliche Informationssysteme und Verteilte Systeme, auf die hier nicht im Detail eingegangen werden soll. Die meisten der genannten Veranstaltungen stehen neben Studierenden der Wirtschaftsinformatik auch Studierenden anderer Fachrichtungen (wie z. B. Mathematik) offen. Die Softwaretechnik-Veranstaltungen werden im folgenden im einzelnen dargestellt. Da bei den Seminaren ausgewählten Aspekte des Software-Engineerings von Veranstaltung zu Veranstaltung unterschiedliche, aktuelle Themen aufgegriffen und vertieft werden, und der Ablauf eines Seminars allgemein bekannt ist, wird hierauf nicht im Detail eingegangen.

2 Veranstaltungen des Grundstudiums

2.1 Programmierung

Die 4-stündige Grundvorlesung "Programmierung" richtet sich an Erstsemester. Die Veranstaltung hat ca. 200 Hörer(innen), die sich zur Hälfte aus Wirtschaftsinformak-Studierenden und zur Hälfte aus Studierenden anderer Fachrichtungen (meist Mathematik (Diplom oder Lehramt) oder Informatik-Lehramt) zusammensetzen. Sie wird durch eine 2-stündige Übung in 8 Kleingruppen ergänzt, die von studentischen Hilfskräften geleitet werden. Ziel ist, das Programmieren-im-Kleinen zu vermitteln. Neben den wesentlichen Konzepten imperativer und objektorientierter Sprachen werden auch deklarative Programmiersprachen, d.h. funktionale und logische Sprachen, und die entsprechenden Programmiertechniken vorgestellt. Der Schwerpunkt liegt hierbei, insbesondere was die Übungen betrifft, auf der objektorientierten Sprache Java. Weiterhin wird das Beweisen von Programmeigenschaften behandelt. Der Lernerfolg wird im Rahmen einer 2-stündigen, für das Vordiplom relevanten Klausur abgeprüft.

2.2 Software-Praktikum

Das Software-Praktikum wird als 4-wöchige, ganztägige Blockveranstaltung in der vorlesungsfreien Zeit im Frühjahr durchgeführt und richtet sich an Studierende nach dem 2. Fachsemester. Aufbauend auf den Grundvorlesungen Programmierung sowie Datenstrukturen und Algorithmen soll im Rahmen dieser Veranstaltung das Programmieren-im-Kleinen vertieft werden. Die Aufgabe besteht darin, ausgehend von einer vorgegebenen Spezifikation ein Softwaresystem zu implementieren und zu testen. Ein typisches Projekt ist die Entwicklung eines Systems zum Buchen von Bahnverbindungen und zur zugehörigen Stammdatenverwaltung.

Da die Anzahl (ca. 100) der teilnehmenden Studierenden bei der Entwicklung eines einzelnen Systems zu groß ist, werden Gruppen gebildet, von denen jede ein komplettes System erstellt. Jede Gruppe besteht wiederum aus 5 Teams, die jeweils einige vorgegebene Klassen des Gesamtsystems zu entwickeln haben. Neben der rein fachlichen Ausbildung liegt ein Aspekt des Praktikums daher auch in der Einübung von Teamarbeit.

Das zu entwickelnde System umfasst u.a. eine zeitgemäße graphische Benutzerschnittstelle, die von je einem Team jeder Gruppe mit Java AWT erstellt wird. Jedes der übrigen Teams ist für die Implementierung einiger Basisklassen (z.B. AVL-Baum, Heap) und Fachkkonzeptklassen verantwortlich.

Um das Projekt bei der hohen Teilnehmeranzahl trotz des engen Zeitrahmens erfolgreich durchziehen zu können, hat sich eine straffe Organisation als notwendig erwiesen. Für jede zu entwickelnde Klasse werden rigorose Deadlines vorgegeben, bis zu denen diese von einem Betreuer abgenommen worden sein müssen. Gleiches gilt auch für die Integrationstests. Bei der Abnahme ist nicht nur der erstellte Code vorzuführen, sondern es müssen auch die Test-Treiber und -Stümpfe sowie systematisch erstellte Testdaten vorgelegt werden, wobei für jede Komponente vorgegeben wird, ob Blackbox- und/oder Glassboxtesten eingesetzt werden soll. Jede Gruppe wird von einem Mitarbeiter betreut. Die notwendigen, grundlegende Kenntnisse über das Testen von Software werden in einem Crash-Kurs zu Anfang des Praktikums vermittelt. Hierbei wird auch Java AWT vorgestellt.

Als Werkzeug wird das Java-Development Kit verwendet. Das Softwarepraktikum wird als Teil der Vordiplomsklausur Informatik abgeprüft. Es ist geplant, die Veranstaltung in Zukunft in die Software-Engineering- Vorlesung (s.u.) zu integrieren.

3 Veranstaltungen des Hauptstudiums

3.1 Software-Engineering I

Als erste Vorlesung zum Thema Software-Engineering behandelt die 4-stündige Hauptstudiumsveranstaltung die frühen Phasen der Softwareentwicklung von der Vorstudie über die Anforderungsanalyse bis hin zum Grobentwurf. Sie bildet damit die Basis für die im darauffolgendem Semester stattfindende Vorlesung Software Engineering II.

Ausgehend von einer systemtheoretischen Fundierung, in der Systemdenken als Hilfsmittel sowohl zur Erklärung als auch zur Gestaltung von Informationssystemen vermittelt wird, werden Methoden und Werkzeuge für die Anforderungsanalyse behandelt. Hierbei wird besonderer Wert auf ein klares Verständnis der behandelten Begriffe, wie System, Modell, Methode, Qualität, Anwendungs- und Informationssystem etc. gelegt. Im Kontext dieses mit thematischer Entfaltung überschriebenen Teils wird sowohl eine prozess- als auch gegenstandsorientierte Abgrenzung der Softwareentwicklung und der Entwicklung industrieller Güter durchgeführt.

Während der Anforderungsanalyse geht es nicht nur um das Aufzeigen unterschiedlicher Ansätze und einer ersten technischen Gestaltung von Systemen, sondern auch um die Planung und geordnete Abwicklung von Projekten. Für die Informationssystemgestaltung werden Referenzmodelle unterschiedlicher Abstraktionsebenen und Anwendungsgebiete besprochen und auf ihre Tauglichkeit hin untersucht. Für klassische und moderne Vorgehensmodelle werden sowohl die Charakteristika ihrer Einsatzbereiche anhand von Fallbeispielen erarbeitet, als auch die unterschiedlichen Perspektiven des Managements und der Entwickler thematisiert. In etwa gleichgewichtig werden psychologische Aspekte der Projektarbeit im Plenum diskutiert und in Form von Rollenspielen illustriert. Die Studenten sollen in die Lage versetzt werden, unterschiedliche Charaktere von Teamarbeitern erkennen zu können und typische Konfliktsituationen zu lösen. Hier besitzt die Vorlesung einen engen Bezug zu Veranstaltungen aus dem Bereich Führung und Organisation.

Im praktischen Teil der Veranstaltung werden die Studierenden in Form von Fallstudien mit klassischen und objektorientierten Methoden vertraut gemacht und im Umgang mit CASE-Tools geübt. Auf der Fachkonzeptebene werden zur Darstellung der Geschäftsvorfälle neben ereignisorientierter Prozessketten (EPKs) auch Petri-Netz-basierte Darstellungen wie Bedingungs-Ereignis-Netze und Stelle-Transitions-Netze genutzt. Als Tools kommen hier ARIS und INCOME zum Einsatz. Bei der objektorientierten Systementwicklung werden Use Cases, Aktivitätendiagramme und erste Klassenmodelle mit Rational Rose erstellt. Die Fallstudien orientieren sich inhaltlich an betriebswirtschaftlichen Problemstellungen etwa einer Auftragsverwaltung, eines Hotelreservierungssystems oder der Reorganisation einer Entwicklungsabteilung.

Pro Veranstaltung sind ca. 100 Studierende zu betreuen, wobei sowohl Vorlesung als auch Übung im Plenum stattfinden. Die in Abhängigkeit vom Umfang wöchentlichen bzw. zweiwöchentlichen vorlesungsbegleitenden Übungen können in Kleingruppen (2-4 Personen) gelöst werden. Über die erfolgreiche Abgabe qualifizieren sich die Studierenden für eine Teilnahme an einer Scheinerwerbsklausur.

3.2 Software-Engineering II

Die 4-stündige Vorlesung Software-Engineering II behandelt im wesentlichen die Phasen Entwurf, Implementierung, Abnahme und Wartung sowie die Qualitätssicherung. Sie wird von den Studierenden in der Regel im 6. Fachsemester gehört. Neben den klassischen Ansätzen strukturierter Entwurf und modularer Entwurf wird hier insbesondere auf den objektorientierten Entwurf eingegangen. Entsprechend wird auch bei der Implementierung und beim Testen zwischen klassischem und objektorientiertem Ansatz unterschieden. Besprochen werden auch die Einbindung von Datenbanken, Expertensystemen und Benutzungsschnittstellen sowie verteilte Anwendungen und der Einsatz von Middleware. Die Veranstaltung umfasst einen praktischen Anteil, in dem mit dem CASE-Tool Rational Rose ein UML-Modell eines kleinen Beispielsystems, beim letzten Mal einer Medienhandlung, zu erstellen ist. Dies geschieht in Kleingruppen mit ca. 5 Teilnehmern. Zunächst wird ein Analysemodell für erstellt, das dann, auch im Hinblick auf eine Verteilung im Netz, die Datenbankanbindung und die Benutzerschnittstelle zu einem Entwurfsmodell verfeinert und schließlich implementiert und getestet wird. Ähnlich wie im Software-Praktikum müssen auch hier Test-Treiber und -Stümpfe sowie Testdaten systematisch generiert werden. Die erlangten Kenntnisse werden in einer 2-stündigen Klausur abgeprüft. Das Ergebnis geht in die Abschlussnote ein. Die Veranstaltung hat 80-100 Teilnehmer.

3.3 Projektseminar

Im Rahmen eines einsemestrigen Projektpraktikums werden die Studenten vor konkrete Entwicklungsaufgaben gestellt. In Gruppen von 3-20 Studierenden haben sie Projekte aus ganz unterschiedlichen Themenbereichen zu bewältigen. Diese reichen von eher betriebswirtschaftlichen Fragestellungen einer Controlling-Konzeption oder der Gestaltung einer Produktionsplanung und -steuerung, über das breite Thema der SAP-Anwendungen und Workflow-Realisierungen, bis hin zur Business-to-Business-Aspekten im Internet oder der technischen Realisierung von Middleware. Allein im Bereich SAP wurden in den vergangenen Jahren von den Studierenden zahlreiche Projekte abgewickelt, die sich inhaltlich etwa mit dem Einsatz, dem Customizing, der Internetanbindung, der Führungsunterstützung oder den Retail-Komponenten beschäftigten. Dabei wurden Aufgaben insbesondere der Fach- und DV-Ebene aber auch der Implementierungsebene gelöst. Einen weiteren Schwerpunkt bilden Themen aus dem Bereich der Interorganisationssysteme und des Wissensmanagements. Üblicherweise werden Projekte zur Verstärkung der Praxisrelevanz in Zusammenarbeit mit Unternehmen durchgeführt. So konnte beispielsweise für eine Versicherung deren Intranet-Konzept oder für ein Fertigungsunternehmen aus dem Mittelstand ein Datawarehouse entwickelt und prototypisch realisiert werden.

Projekte werden in Abhängigkeit von der Teilnehmerzahl und der Interessenlage der wissenschaftlichen Mitarbeiter im Verhältnis von 5 zu 1 betreut. Die Zuordnung der Studierenden zu den Projekten erfolgt innerhalb der verfügbaren Kapazitäten über eine freie Wahlmöglichkeit seitens der Studenten. Die Erfahrungen haben gezeigt, dass Gruppengrößen zwischen 5-10 Mitgliedern für die typischen Projekte ideal sind. Bei weniger als fünf Teilnehmern sind neben einer reduzierten Aufgabenstellung auch Einbußen im Bereich des Lernens über Teamarbeit zu akzeptieren. Die inhaltliche und organisatorische Ausgestaltung obliegt dem Projektanbieter. Üblicherweise wird auf der Grundlage einer Aufgabenstellung der Praxis oder eines hypothetischen Falls von einer eher vagen Spezifikation ausgegangen, die dann von den Studierenden inhaltlich detailliert werden muss. Geführt werden die Projekte seitens der Betreuer anhand von Ergebnissen, die zeit- und qualitätsgerecht durch die Teilnehmer abgeliefert werden müssen. Dabei wird den Studenten bewusst ein hohes Maß von Selbstkoordination zugestanden, so dass im Projektverlauf aus eher steuernden Eingriffen seitens der Projektleitung ein begleitendes Coaching wird.

Der Projektverlauf selbst folgt einer Dreiphasigkeit. Die Phase 1 oder Einarbeitungsphase beinhaltet ein themenbezogenes Literaturstudium, welches mit einer 10-15-seitigen Dokumentation und einer Präsentation abschließt. Nach der anschließenden Detaillierung des Projektplans und der damit verbunden Aufgabenzuordnung erfolgt in der zweiten Phase die eigentliche Abwicklung des Projekts. Ausführliche Abschlussdokumentationen (ca. 150 Seiten) und professionelle Ergebnispräsentationen bilden den Inhalt der abschließenden dritten Phase. Während des gesamten Verlaufs werden die Studierenden zu einem professionellen Arbeiten und praxisnahen Rahmenbedingungen angehalten. Hierdurch erklärt sich der im Vergleich zu anderen Veranstaltungen hohe Zeit und Arbeitsaufwand der Studierenden, die üblicherweise während des Seminars ihre sonstigen Aktivitäten reduzieren.

4 Zusammenfassung

Softwaretechnik wird im Rahmen der Wirtschaftsinformatik-Studiengänge an der Universität Münster in Form von Vorlesungen, Praktika und Seminaren vermittelt. Der praktische Anteil und insbesondere der Einsatz von CASE-Tools haben hierbei einen hohen Stellenwert. Eine spezifische Ausrichtung in Richtung Wirtschaftsinformatik ist insbesondere in den Veranstaltungen Software-Engineering I und den Projektseminaren festzustellen. Die übrigen Softwaretechnik-Veranstaltungen unterscheiden sich nur unwesentlich von dem Angebot an reinen Informatik-Fachbereichen.