

INSTITUT FÜR INFORMATIK

DER
LUDWIG-MAXIMILIANS-UNIVERSITÄT
MÜNCHEN



Diplomarbeit

Christian van der Leeden

Entwicklung eines intranet-basierten Systems zur
Unterstützung von Vertriebsprozessen

Aufgabensteller: Prof. Dr. Martin Wirsing

Betreuer: Reiner Deichmann, Apple Computer
Nora Koch

Abgabedatum: 30. Oktober 1998

Erklärung

Ich versichere, daß ich diese Diplomarbeit selbständig verfaßt und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 30.10.1998

Christian van der Leeden

Vielen Dank

an Herrn Professor Wirsing, für die Annahme des Themas.

meine Betreuerin Nora Koch, für die sehr gute Betreuung und Hilfe bei der Analyse-, Design- und Implementierungsphase insbesondere bei der Erstellung der OOHDM Dokumente.

an Herrn Reiner Deichmann, Leiter der Corporate Accountsabteilung bei Apple Computer Deutschland, der immer Zeit für mich fand und während der zahlreichen Treffen mir Einblick verschaffte in die Anforderungen an ein Vertriebsinformationssystem.

an die Mitarbeiter der Corporate Accounts Abteilung bei Apple Computer Deutschland, die sich offen für meine Befragung gezeigt haben und bereitwillig ihre Zeit geopfert haben.

für den kleinen Drachen und das Taschenpu

Inhaltsverzeichnis

1.	<u>Einleitung</u>	1
2.	<u>State of the Art</u>	5
2.1	Beschreibung	5
2.2	ACT!	6
2.3	Archer Enterprise for ACT!	6
2.4	Sales Logix	7
2.5	Siebel	7
2.6	Vergleich zu diesem System	8
3.	<u>Analyse</u>	9
3.1	Allgemeine Beschreibung	9
3.2	Ist Zustand	10
3.2.1	Projektdefinition	11

3.2.2	Grunddaten eines Projekts	16
3.2.3	Milestones	16
3.3	Soll-Zustand	16
3.3.1	Projektdurchführung	16
3.3.2	Aktivitäten	19
3.3.3	Weitere Punkte	20
3.3.4	Reportgenerierung	20
3.3.5	Motivation für die Benutzer	21
3.4	Weitere Bemerkungen	22
3.5	Liste von weiteren Requirements	22
3.6	Use Cases	22
3.7	Szenarios	39
3.7.1	Projektvorschlag	39
3.7.2	Projekterfassung	39
3.7.3	Projektverwaltung	39
3.7.4	Einloggen in das System	40
3.7.5	Projektreporting	40
3.7.6	Reporting	41
3.8	Glossar	41
3.9	Mögliche Attribute der wichtigsten Objekte	45
3.9.1	Projekt	45
4.	OOHDM	48
4.1	Einleitung	48
4.2	Conceptual Design	50
4.3	Navigational Design	50
4.4	Abstract Interface Design	54
5.	Design	56
5.1	Conceptual Design	56
5.2	Navigational Model	69
5.2.1	Navigational Schema	70
5.2.2	Contextual Schema	71

5.3	Abstract Interface Design	74
5.3.1	Menüs	75
5.3.2	Projektverwaltung	76
5.3.3	Accounts und Personen	80
5.3.4	Erinnerungen	82
5.3.5	Reporting	83
5.3.6	Projektvorschlag	83
6.	Implementierung	85
6.1	Einleitung	85
6.2	Implementierung des Conceptual Designs	86
6.3	Implementierung des Navigational Designs	94
6.4	Implementierung des User Interfaces	96
6.5	Die Implementierungstechnologien	97
6.5.1	WebObjects - ein Überblick	97
6.5.2	Struktur von WebObjects	98
6.5.3	Der Datenbankanschluß mit EOF	101
6.6	Einige Realisierungskonzepte	103
6.6.1	Allgemeine Anzeigeelemente	103
6.6.2	Konfiguration der Anzeige durch Attributlisten	104
6.6.3	Detailanzeigen	106
6.6.4	Konsistenz der Datenbank sicherstellen	107
7.	Benutzerhandbuch	109
7.1	Allgemeine Bedienhinweise	109
7.2	Starten der Applikation	111
7.3	Erinnerungen	112
7.4	Projektvorschlag	114
7.5	Projektverwaltung	121
7.6	Accounts	134
7.7	Personen	136
7.8	Reports	139

7.9	Produktkategorien	141
7.10	Optionen	142
8.	Bewertung und Zusammenfassung	143
A.	Anhang	147
8.1	Beispiel einer Bewertungstabelle für Projekte	148
B.	Literaturverzeichnis	149

1. Einleitung

Das World Wide Web (WWW) ist ein globales Hypertextsystem, das am CERN entwickelt wurde. Die Dokumente werden in der Beschreibungssprache HTML (Hypertext Markup Language) erstellt und mittels Hyperlinks untereinander verbunden. Der Benutzer bedient sich eines Browsers, um die Dokumente vom Webserver zu laden und anzuzeigen. Das WWW ist somit eine Client/Server Anwendung, welche plattformunabhängig ist, da für beinahe jedes Betriebssystem ein Browser existiert. Existierende Anwendungen des WWW sind z.B. Firmenpräsentationen im Internet, Kataloge und Einkaufssysteme oder auch Internetbankingsysteme. Der meistverbreitete Einsatz des WWW bedient sich statischer Dokumente, die in einer festgelegten Struktur miteinander verbunden sind. Die Einsatzmöglichkeiten des WWW erweitern sich stark, wenn die Dokumente dynamisch auf Benutzeranfrage erstellt werden. Dies ermöglicht die Erstellung von Applikationen, welche als Benutzerschnittstelle (den Browser) haben. Die Vorteile solcher Applikationen sind Plattformunabhängigkeit, geringe Anforderungen an den Client (es muß ein Browser vorhanden sein), und die intuitive Benutzung des WWW.

In dieser Diplomarbeit soll ein Vertriebsinformationssystem auf Basis des World Wide Webs entwickelt werden¹. Ein solches System dient verschiedenen Benutzergruppen. Auf der einen Seite stehen die Vertriebsmitarbeiter, welche Unterstützung in ihrem täglichen Arbeitsablauf erwarten und auf der anderen

Seite steht das Management, welches eine möglichst große Kontrolle über seine Mitarbeiter durch das System erlangen und alle Informationen jederzeit aktuell einsehen möchte.

Konkret ergibt sich folgende Problemstellung: Die Großkundenabteilung bekommt durch die Händler oder direkt vom Kunden Informationen über Projekte (einen Projektvorschlag), bei denen die Anschaffung von Vertriebsprodukten eine Rolle spielt. Um bei einem solchen Projekt erfolgreich bis zum Vertragsabschluß zu kommen, müssen verschiedene Zwischenziele (Milestones) erreicht werden, die wiederum mit Aktivitäten versehen sind. Weiterhin können den Milestones Verkäufe zugeordnet werden. Dabei spielt der Vertriebsprozeß eine Rolle. Neben der Verwaltung dieser verschiedenen Aktivitäten und Milestones, müssen die Informationen auch noch allen Parteien, die an einem Projekt mitarbeiten, zur Verfügung gestellt werden.

Weitere Anforderungen an die Applikation waren der leichte Zugriff von jedem Rechner (ergo Betriebssystem) und von Partner auf das System. Diese Einschränkung läßt keine Anwendung zu, welche eine Installation und Entwicklung einer plattformabhängiger Client-Software erfordert. Die Entscheidung viel somit auf das WWW, als plattformunabhängige Client-Server Technologie mit einer schon vorhandenen Client -Software, dem Browser.

Die allgemeinen Ziele eines Softwareprojekts sind die Erstellung eines Programms, welches die Erwartungen der Benutzer zufrieden stellt und in seiner Arbeit unterstützt, die Einhaltung von finanziellen und zeitlichen Rahmenbedingungen und die Möglichkeit, Änderungen und Anpassungen flexibel vorzunehmen[33]. Um diese Ziele zu erreichen, muß ein strukturierter Ansatz bei der Entwicklung befolgt werden. Die Entwicklung wurde in die Analysephase, die Designphase und die Implementierungs- und Testphase eingeteilt. Hierbei wurden schon erprobte Verfahren benutzt. In der Analysephase wurden die

1. In der hier vorliegenden Diplomarbeit wurde ein Vertriebssystem für die Großkundenabteilung von Apple Computer GmbH Ismaning entwickelt. Der Ansprechpartner war der Leiter der Abteilung Herr Reiner Deichmann.

Anforderungen an das System mittels Fragebögen und Gesprächen ermittelt und im Use-Case Modell, welches von der Methode OOSE stammt, festgehalten. Damit war der Grundstein für die nächste Phase, die Designphase, gelegt. Die Wahl der Designmethode mußte auf die speziellen Bedürfnisse der Webtechnologie abgestimmt werden. In diesem Bereich existieren mehrere Methoden wie RMM, WSDM (Web-Site Design Method) oder OOHDM (Object Oriented Hypermedia Design Method). Alle diese Methoden sind sehr neu und wurden bis jetzt nur im universitären Umfeld verwendet. Die Wahl fiel auf die letztere, da diese Methode im Gegensatz zu RMM objektorientiert ist und WSDM zu dieser Zeit noch nicht ausreichend dokumentiert war. Basierend auf den Resultaten der Analyse wurde das Conceptual Design entworfen. Nachfolgend wurde das Navigational Design und Abstract Interface Design erstellt, welche sich gegenseitig stark beeinflussen und deswegen zyklisch durchlaufen wurden. Nach der Designphase kam die Implementierung. Die verwendete Technologie wurde vom Kunden ausgewählt. Die Wahl fiel auf WebObjects, da diese Entwicklungsumgebung für Internetapplikationen ein objektorientierte, datenbankunabhängige und plattformunabhängige Lösung ist und ein Produkt des Kunden ist. Die Implementierung erfolgte schrittweise, wobei das Programm in mehrere Module aufgeteilt wurde, welche sukzessiv implementiert wurden. Als Grundstein für das Datenbankmodell diente das Conceptual Design. Das Abstract Interface Design beeinflusste die Erstellung der Benutzerschnittstelle und das Navigational Design wurde in der Programmlogik umgesetzt. Die drei Phasen Analyse, Design und Implementierung wurden sowohl sequentiell wie auch zyklisch durchlaufen. So gab es während der Designphase noch Änderungen in der Analyse und während der Implementierungsphase noch Änderungen im Design. Die Unterschritte der einzelnen Phasen wurden meist zyklisch durchlaufen. Beim Design gab es die Schleife zwischen Navigational Design und Abstract Interface Design.

Das erstellte Programm gliedert sich in mehrere Teilbereiche: Projektvorschlag, Projekte, Accounts, Personen, Reporting und Erinnerungen. Die Benutzerschnittstelle ist aufgeteilt in einen Menübereich mit Einträgen für jeden Bereich, einem Submenübereich mit den einzelnen Unterfunktionen und dem Hauptteil,

dem Arbeitsbereich des Benutzers. Ein Grundbaustein ist die Benutzerverwaltung, welche die Berechtigungen der einzelnen Benutzer verwaltet und die Zugriffskontrolle übernimmt. Der zentrale Bereich ist die Projektverwaltung, welche die Bearbeitung und Erstellung von Vertriebsprojekten mit ihren Milestones enthält. Für den Vertriebsleiter ist der Bereich Reporting relevant, welcher ihm einen Überblick über die laufenden Vertriebsprojekte mittels verschiedener Reports gibt. Ein weiterer Einstiegspunkt für die Projektverwaltung sind die Erinnerungen, welche jegliche Änderungen an Projekten mitdokumentieren und sie benutzerspezifisch präsentieren. Die Bereiche Personen und Accounts stellen Funktionen zur Pflege der Kontakdaten und Accountdaten zur Verfügung.

Die vorliegende Diplomarbeit gliedert sich in acht Abschnitte. In Abschnitt 2 wird der Stand der Dinge beschrieben. Im nachfolgenden Abschnitt 3 wird auf die Resultate der Analyse und im Abschnitt 5 auf das Design eingegangen. Vor dem Kapitel Design ist im Abschnitt 4 noch eine kurze Beschreibung der Methode OOHDM eingefügt. Abschnitt 6 ist der Beschreibung der Implementierung gewidmet, gefolgt von einer kurzen Anleitung für den Benutzer der Applikation in Abschnitt 7. Zum Schluß wird in der Zusammenfassung in Abschnitt 8 die Resultate beschrieben, sowie eine persönliche Bewertung und Ausblick auf Erweiterungen und andere Einsatzgebiete des Programms gegeben.

2. State of the Art

Ein Vertriebsinformationssystem per se ist keine neue Idee. Es existieren schon verschiedenartige Lösungen auf dem Markt, die mehr oder weniger den Vertriebsmitarbeiter unterstützen bzw. kontrollieren. In diesem Kapitel wird eine Beschreibung von vier Systemen verschiedener Ausprägung gegeben und das entwickelte System gegenüber diesen Systemen abgegrenzt.

2.1 Beschreibung

Das Ziel eines Vertriebsinformationssystems ist, den Mitarbeiter beim Prozeß des Verkaufens größtmöglich zu unterstützen und dem Management die Kontrolle über die Leistungen und Fortschritte von Projekten zu ermöglichen. Dies sind zwei gegenläufige Ziele. Das Management möchte am liebsten die Eingabe aller anfallenden Daten für die anschließende Auswertung und Kontrolle der Mitarbeiter. Der Vertriebsmitarbeiter ist im Gegensatz daran interessiert, nur die notwendigen Daten einzugeben und nicht durch das System und überflüssige Eingaben in seiner Art zu Verkaufen eingeschränkt zu werden. Der Verkaufsprozeß kann jedoch nicht unbeschränkt vereinheitlicht werden, da er eine nicht rationale Komponente, nämlich die Mensch-zu-Mensch Beziehung beinhaltet.

Bestehende Vertriebsinformationssysteme variieren in ihrer Unterstützung des Vertriebsmitarbeiters. Zum Einen existieren die Kontaktverwaltungssysteme, die den Vertriebsmitarbeiter nur bei der täglichen Büroarbeit unterstützen, ihm

jedoch keine Planungswerkzeug an die Hand geben und zum Anderen gibt es die Sales Force Automation Systeme, welche eine komplette Integration in alle, mit dem Verkauf verbundenen Abteilungen und eine große Planungs- und Kontrollfunktion bieten. Existierende Systeme liegen innerhalb dieser zwei Grenzen. Nachfolgend werden vier Systeme vorgestellt und abschließend mit dem in dieser Diplomarbeit entwickelten System verglichen und abgegrenzt. Zuerst wird ein reines Kontaktmanagementwerkzeug und zum Schluß ein vollintegriertes System vorgestellt.

2.2 ACT!

ACT! ist ein Kontaktmanagementwerkzeug welches von Symantec hergestellt wird. . Das System verfügt über eine zentrale Datenbank, in der alle Adressinformationen gespeichert werden. Die Grundeinheit eines Datensatzes bezieht sich auf eine Person. Zusätzlich können hier Aufgabenlisten verwaltet, Gesprächsnotizen erstellt, Briefe geschrieben und Termine verwaltet werden. Eine Einbindung in die normalen Büro- und Internetanwendung ist vorhanden. Das System übernimmt somit administrative Funktionen für den Vertriebsmitarbeiter, jedoch ist keinerlei Projektplanung oder -verfolgung integriert[19]. Das Management hat auch keinerlei direkten Nutzen aus den eingegebenen Daten in Form von Reports.

2.3 Archer Enterprise for ACT!

Das Produkt Archer Enterprise for ACT! basiert auf ACT!. Neben der kontaktbasierten Sichtweise wird eine accountbasierte Sichtweise eingeführt, da Vertriebsprojekte immer für einen Kunden (Account) durchgeführt werden[22]. Die Groupware-Aspekte von ACT! bleiben bei der Lösung vollständig erhalten. Weiterhin kommt Archer Enterprise mit einem Projektinformationssystem, welches die Planung, die Durchführung und Kontrolle von Vertriebsprojekten ermöglicht. Damit besitzt nun auch die zweite Benutzergruppe, das Management, die Möglichkeit aus den eingegebenen Informationen Reports und Forecasts zu generieren. Durch die Client-Server Architektur ist es nun auch mög-

lich, andere Abteilungen wie das Marketing oder das Accounting mit einzubeziehen[23]. Eine Anbindung an Produktdatenbanken oder direkt an existierende Accountingsoftware ist nicht vorhanden.

2.4 Sales Logix

Ein gänzlich unabhängige Lösung für ein Vertriebsinformationssystem bietet SalesLogix. Sie zielt auf die mittelständischen Unternehmen, welche mehr Unterstützung als ein Kontaktmanagementsystem wie ACT! benötigen, aber trotzdem noch keine unternehmensweite Lösung brauchen[15]. Es enthält wie die vorherige Lösung Kontaktmanagementwerkzeuge, Reporting, Projektplanung und -verfolgung sowie Einbindung in die verschiedenen Büroanwendungen inklusive eines grundlegenden Dokumentenmanagementsystems. Es können verschiedene Datenbanken verwendet werden[14]. Diese Lösung ist im Großen und Ganzen identisch zur vorherigen Lösung, jedoch baut sie auf kein existierendes Produkt auf.

2.5 Siebel

Die Firma Siebel Inc. bietet eine modular aufgebautes Sales Force Automation System an. Das Kernmodul ist das Salesmanagementmodul[18], welches Funktionalitäten wie Reporting, Charting, Projektplanung und -durchführung sowie Abrechnungsfunktionen bietet. Ergänzt wird dieses Modul durch ein Accountmanagementmodul, welches die unternehmensweite Verwaltung von Kunden und Kontakten erlaubt. Für den produktorientierten Vertrieb existiert weiterhin ein ProductConfigurator, welcher dem Vertriebsmitarbeiter bei der Zusammenstellung und Konfiguration eines Produkts hilft. Der ganze Prozess kann durch das QualityTool, einem Qualitätssicherungswerkzeug überwacht und festgelegt werden[17]. Der Zielmarkt sind hierbei die großen Unternehmen mit mehreren hundert bzw. tausend Vertriebsmitarbeitern. Die Integration in die bestehenden Produktdatenbanken und Accountingsoftware ist eng. Der Vorteil einer solchen Lösung ist das Spektrum der Anwendungsgebiete und die enge Integration in alle Abteilungen. Der daraus resultierende Nachteil ist jedoch die Komplexität

des Produkts, die Einschränkung der Arbeitsweise der Vertriebsmitarbeiter und die wahrscheinlich hohen Investitionskosten. Siebel ist bei Vertriebsinformationssystemen vergleichbar mit SAP im Finanzbuchhaltungsbereich.

2.6 Vergleich zu diesem System

Das hier entwickelte System hat den Vertriebsmitarbeiter im Mittelpunkt. Es leistet keine Kontaktmanagementaufgaben. Im Zentrum steht die Projektverwaltung und die Möglichkeit, daß viele verschiedene Personen mit diesem Werkzeug ein Projekt gemeinsam verwalten. Steht bei den traditionellen Lösungen die Kostenfrage und die Optimierung des Vertriebsprozeß im Vordergrund, wurde hier die Betonung auf die Hilfe für den Vertriebsmitarbeiter gelegt. Deshalb kann das hier zu entwickelnde System keine genaue Kostenkontrolle wie die Lösungen von Siebel, Archer Enterprise oder SalesLogix, liefern. Der Groupware-Aspekt steht hier im Vordergrund. Gleichzeitig soll aus den Daten, die gesammelt werden, noch genügend Information für das Management gewonnen werden. Die hier entwickelte Lösung ist eine Insellösung, aber sie wird jedoch in die bestehenden Systeme integriert werden. Ein limitierender Faktor für viele der hier vorgestellten Lösungen ist die Betriebssystemfrage. In dem hier entwickelten System wurde sie durch den Einsatz der Webtechnologie obsolet. Demnach ist sie für jeden erreichbar, der Zugang zum Internet bzw. zum firmeneigenen Intranet hat. Es ist also eine echte Business-to-Business Anwendung, die natürlich die Hauptkontrolle beim Vertrieb beläßt, jedoch die Einbindung von Partnern auf einfache Weise ermöglicht.

Die Vorteile gegenüber traditionellen Lösungen liegen demnach in der Verwendung vom World-Wide-Web als nächstes "Betriebssystem" und der Fokussierung auf den Vertriebsmitarbeiter. Eine quasi-freiwillige Nutzung des Systems ist die beste Grundlage für eine solide Datenbasis, auf der das Management seine Reports generieren kann. Es handelt sich hier auch um eine maßgeschneiderte Lösung, welche sich an den Bedürfnissen des Auftraggebers orientiert. Ein Nachteil kann die schwierige Erweiterbarkeit sein, welcher jedoch durch ein offenes Design vorgebeugt ist.

3. Analyse

Die Analyse stützt sich auf die Informationen, welche durch Befragungen der Vertriebsmitarbeiter und Manager gewonnen wurden. Der erste Teil der Analyse beschäftigt sich mit der Ist- und Soll-Analyse sowie der allgemeinen Definition eines Vertriebsinformationssystems. Im zweiten Teil werden mit Hilfe von Use-Cases die möglichen Benutzungsfälle aufgezeigt, welche als Basis für das Design dienen. Abschließend wird ein Glossar mit den verwendeten Begriffen und ihrer Definitionen erstellt.

3.1 Allgemeine Beschreibung

Ein Vertriebsinformationssystem hat zwei Benutzertypen. Auf der einen Seite steht der Vertriebsleiter, der an möglichst umfassenden Informationen über alle Projekte und Leistungen seiner Vertriebsmitarbeiter interessiert ist, auf der anderen Seite steht der Vertriebsmitarbeiter, der in seiner Arbeit unterstützt werden möchte und nicht durch die Eingabe für ihn überflüssiger Informationen daran gehindert werden soll. Damit die Güte der eingegebenen Daten hoch ist, muß der Vertriebsmitarbeiter einen Mehrwert aus der Benutzung des Systems im Sinne von Arbeitserleichterung oder höherer Effizienz bekommen. In dem zu entwickelnden Vertriebsinformationssystem steht deswegen der Vertriebsmitarbeiter im Mittelpunkt der Entwicklung. Aus den so gesammelten Daten können Auswertungen für das Management erstellt werden. [42]

Das System gliedert sich in zwei Bereiche: dem Projektinformationssystem und dem Reportingssystem. Das Projektinformationssystem wird hauptsächlich von den Vertriebsmitarbeitern benutzt, um die Vertriebsprojekte zu planen, durchzuführen und mit externen Partnern zu koordinieren. Der Bereich Reporting nutzt die im Projektinformationssystem eingegebenen Daten, um daraus für die Vertriebsleiter Umsatzvorhersagen zu generieren und den getätigten Umsatz pro Mitarbeiter zu errechnen.

3.2 Ist Zustand

Die ist aufgeteilt in drei Markt- und Kundensegmente: Design und Publishing, Education (Lehr- und Forschungseinrichtungen) sowie K12 (Schulen und Kindergärten). Design und Publishing kümmern sich unter anderem um Verlage, Digital Video oder auch Internet/Intranet/Extranet.

In der Corporate Accounts (CA) Abteilung ist noch keine strukturierte Form der Projektdurchführung vorhanden. Der Hauptfokus der CA Abteilung liegt im Verkauf von Hardware. Es existiert ein sehr rudimentäres Tool basierend auf FileMaker, welches die Eingabe von Kunden, ihren zu erwartenden Bestellungen (Anzahl der Rechner) und der Wahrscheinlichkeit der Auftragserteilung ermöglicht. Keinerlei Projektmanagementfunktionalität, wie Einträge von Aktivitäten für das Projekt, Definition und Überprüfung von Zielen etc., ist vorhanden. Forecasts beschränken sich auf den Ausdruck von den für diesen Monat zu erwartenden Verkauf von Rechnern und der geschätzten Wahrscheinlichkeit.

Die Vertriebskanäle sind indirekt, d.h. es werden keine Produkte direkt an den Kunden verkauft. Alle Verkäufe laufen über die Händler. Die Corporate Accounts Abteilung fällt in die Kategorie der kundenorientierten Verkaufsorganisation, da sie ihre Bereiche nach verschiedenen Kundenklassen aufschlüsselt.[43] Aus dem SAP System steht die Information von den verkauften Rechnern pro Händler zur Verfügung. Diese läßt aber keine Rückschlüsse zu, an welchen Kunden (Account) der Händler die Rechner verkauft hat. Da ein Account von einem Mitarbeiter betreut wird, müßte die Information verkaufte Rechner

pro Account vorhanden sein, um festzustellen, welche Rechner der Vertriebsmitarbeiter verkauft hat. Damit ist auch keine Information über die Leistung (Umsatz) der ganzen Abteilung möglich. Auch dieses Problem wird mittels des entwickelten Vertriebsinformationssystems gelöst.

Nachfolgend soll ein Überblick über die Begriffe *Projekt* sowie einen Projekt- ablauf gegeben werden.

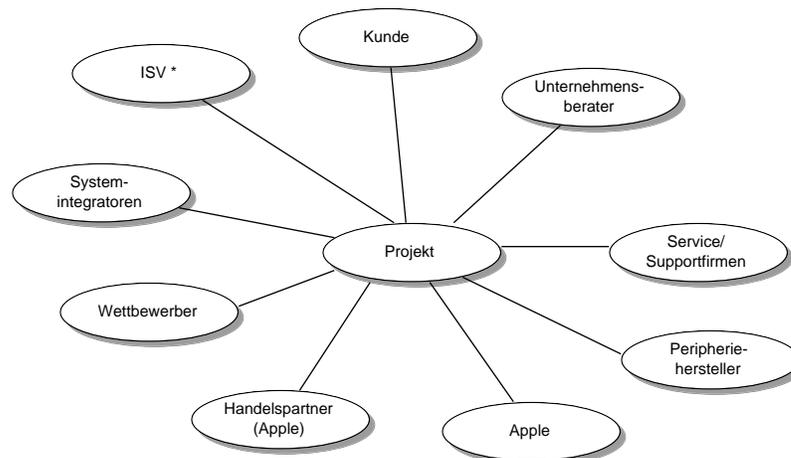
3.2.1 Projektdefinition

Ein Projekt ist eine Zusammenfassung von Leistungen/Produkten, um ein Problem des Kunden zu lösen. Es hat drei verschiedenen Dimensionen: eine zeitliche, eine inhaltliche und eine Budgetdimension.

Ein Projekttemplate ist ein vordefiniertes Projekt mit Aktivitäten und Milestones, das als Grundlage für neue Projekte dient. Mit Hilfe eines Projekttemplates kann ein Verkaufsprozeß im System abgebildet werden.

3.2.1.1 Inhaltliche Dimension

Abbildung 3-1 Mitwirkende bei einem Projekt der Abteilung Design und Publishing



*ISV=Independant Software Vendor

Bei einem Projekt können die in Abbildung 3–1 dargestellten Partner mitwirken. Diese Aufstellung ist typisch für die Design und Publishing Abteilung. Es müssen nicht alle Mitwirkenden involviert sein. Dies hängt v.a. von der Größe des Projekts ab. Für andere Abteilungen ist Abbildung 3–1 leicht zu modifizieren. So ist bei vielen Design und Publishingprojekten ein separater Peripheriehersteller dabei, der sehr spezielle Geräte liefert. Bei Projekten der Abteilungen Education und K12 ist dies nicht der Fall.

Der wichtigste Partner in einem Projekt ist der Kunde, deswegen ist es wichtig, über ihn mehr Informationen zu kennen. Hierzu gehört die innere Struktur des Kunden. Die in Abbildung 3–2 gezeigte Struktur ist typisch für die meisten Kunden. Da bei einem Verkauf nicht die Firma im Vordergrund steht, sondern immer über Personen verkauft wird, ist es wichtig zu wissen, welche Personen in welchen Abteilungen für was verantwortlich sind. Es ist auch notwendig, die internen politischen Strukturen abzubilden, d.h. wer beeinflusst wen, wer gehört dem Inner Circle der Firma an, wer ist Entscheidungsträger, wer hat nur beratende Funktion etc.

Die Schlüsselperson beim Kunden ist der Projektleiter. Alle Aktivitäten sollten über ihn laufen oder mit ihm abgesprochen sein. Weiterhin ist es nützlich noch den Projektleiter beim Key-Partner (Partner mit dem größten Anteil am Projekt) zu kennen. Der Partnerprojektleiter hat den häufigsten Kontakt zu dem Projektleiter beim Kunden. Damit ist er eine wichtige Informationsquelle.

Abbildung 3–2

Kundenstruktur

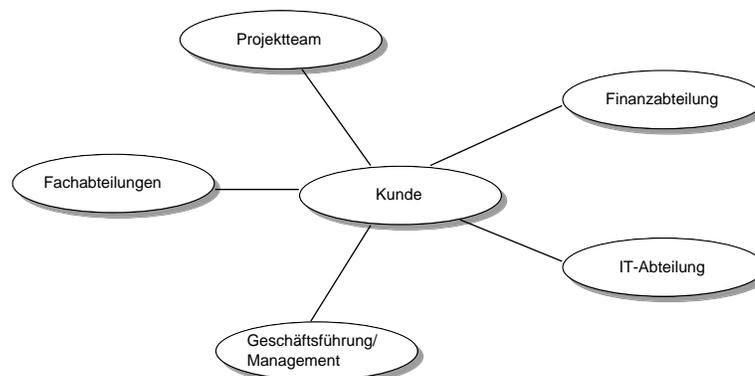


Tabelle 3–1 Beschreibung der Teilnehmer an einem Projekt

Bezeichnung	Beschreibung
Kunde	Der Kunde hat ein Problem, welches es in einem Projekt zu lösen gilt. Durch das Problem und ein Konzept wird der Bedarf des Kunden ermittelt.
Unternehmensberater	Ein Unternehmensberater hilft dem Kunden zu einem vorliegenden Problem ein Lösung zu finden, indem er ein Konzept erarbeitet. Aus dem Konzept geht der Bedarf hervor.
Independant Software Vendor (ISV)	Ein ISV liefert spezielle Software für den Problembereich.
Systemintegrator	Ein Systemintegrator liefert Dienstleistungen (z.B. in Form von Konfiguration, Installation, Anpassung von Software), Software und in begrenzten Maße Hardware
Peripheriehersteller	Ein Peripheriehersteller liefert z.B. Scanner, Belichter, Drucker und andere Peripheriegeräte.
Support/Servicefirmen	Support und Servicefirmen bieten Dienstleistungen wie z.B. Wartung der Installation, Netzwerkadministration oder auch Schulungen an.
Handelspartner	Handelspartner verkaufen Produkte von Apple an den Kunden
Apple	Der Accountmanager der Firma Apple betreut das Projekt mit und unterstützt den Handelspartner.
Wettbewerber	Wettbewerber liefern zu Apples Produktpalette konkurrierende Produkte. Hierzu zählen z.B. PC Hersteller.

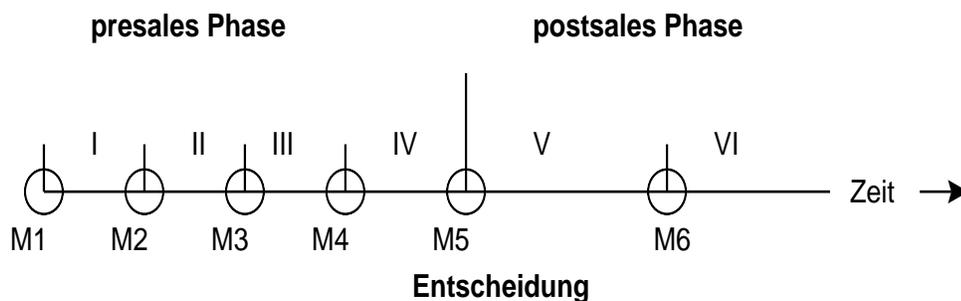
3.2.1.2 Zeitliche Dimension

Ein Projekt hat außer einer inhaltlichen Dimension auch noch eine zeitliche Dimension wie in Abbildung 3–3 dargestellt. Die erste Unterteilung auf der Zeitachse ist vor der Kaufentscheidung des Kunden (Presales Phase) und nach der Kaufentscheidung des Kunden (Postsales Phase). Diese Phasen lassen sich so wie abgebildet weiter unterteilen. Die dargestellten verschiedenen Phasen

sind typisch für ein Projekt. Sie definieren einen Verkaufszyklus, der im Rahmen eines Projekts mehrfach durchlaufen werden kann (mehrere Verkäufe im Rahmen eines Projekts).

Jede Phase ist von Milestones beschränkt. Zwischen diesen Milestones liegen für die Phase typische Aktivitäten und eventuell weitere Milestones. Weiterhin können eigene Aktivitäten definiert werden. Aktivitäten sind nicht unbedingt an einen Milestone gebunden sondern können auch an das Projekt an sich gebunden sein.

Abbildung 3-3 Zeitphasen eines Projekts



Milestone: M1-M6

Phase: I-VI

Für den Vertriebsleiter ist es wichtig zu wissen in welcher Phase sich das Projekt gerade befindet. Neben den Phasen gibt es noch drei Projektkategorien A, B und C. Die Einstufung der Projekte geschieht nicht automatisch nach der Phase des Projekts sondern wird manuell vorgenommen. Die Kriterien nach denen ein Projekt eingestuft wird, müssen vom Vertriebsteam festgelegt werden. Die Einstufung in die Projektkategorien ist für die Beurteilung der Gesamtsituation wichtig. Siehe auch Abschnitt 3.3.1.2.

Um den oben dargestellten Vertriebsprozeß mit seinen Phasen abzubilden können Projekttemplates definiert werden. Sie enthalten Milestones und zu den Milestones typische Aktivitäten. Bei der Neueingabe eines Projekts wird dann als Grundlage des Projekts das Projekttemplate genommen. Es ist mögliche

mehrere Projekttemplates zu definieren, um verschiedene Projektabläufe abzubilden. Z.B. wird ein größeres Projekt sicher mehr Aktivitäten enthalten als ein kleineres Projekt.

Tabelle 3-2 Phasen eines Projekts

Phasen	Beschreibung
I	Problemanalyse und Bedarfsfindung: Mit dem Kunden wird das Problem analysiert, ein Konzept erarbeitet und der Bedarf festgestellt.
II	Lösungsfindung: Mögliche Lösungen für das Problem werden erarbeitet. Es fällt schon eine Vorentscheidung für die einzusetzende Technologie.
III	Angebotsphase: Von allen Konkurrenten wird ein Angebot vorgelegt.
IV	Entscheidungsphase: Der Kunde entscheidet sich für ein Angebot.
V	Implementierung: Die Lösung wird implementiert.
VI	Wartung: Die Installation wird während des Einsatzes betreut und gewartet.

3.2.1.3 Budgetdimension

Weiterhin haben Projekte noch eine Budgetdimension. Hierbei kommt zum Tragen, wie groß das Projekt ist. Das Gesamtbudget ist unterteilt in verschiedenen Kategorien. (Hardware, Software, Dienstleistungen, Peripherie). Auch kann das Gesamtbudget nach Partnern aufgeteilt werden. Entscheidend für die Größe des

Projekts ist nicht das Gesamtbudget sondern der Anteil von Apple. Es kann Projekte geben, die ein Volumen von mehreren Millionen DM haben, wobei der Anteil von Apple jedoch nur einige Zehntausend DM beträgt.

Apples Anteil an einem Projekt wird weiter aufgeschlüsselt in verschiedene Produktkategorien. Mögliche Produktkategorien sind *lowend*, *midrange*, *highend* Systeme, Powerbooks, Monitore oder Peripheriegeräte. Für die Berechnung der Gesamtsumme wird von einem Durchschnittspreis pro Kategorie ausgegangen.

3.2.2 Grunddaten eines Projekts

Aus der inhaltlichen Dimension ergeben sich die Grunddaten eines Projekts. Es besteht aus einer Projektnummer, einem Projektnamen, einer Projektbeschreibung, einem Kunden, einem Projektleiter beim Kunden, einem Projektleiter beim Keypartner, den Entscheidern beim Kunden und der inneren Struktur des Kundens.

3.2.3 Milestones

Ein Projekt besteht aus mehreren Milestones. Sie werden durch ihren Milestonetyp eindeutig einer Phase zugeordnet. Es existiert ein ausgezeichneter Milestonetyp für das Ende der Phase IV (Entscheidungsphase). Er enthält die mit dieser Entscheidung verbundenen geplanten und getätigten Verkäufe aufgeschlüsselt in Produktkategorien. Da ein Verkaufszyklus im Rahmen eines Projektes öfters durchlaufen werden kann, können auch mehrere Entscheidungsmilestones in einem Projekt vorhanden sein.

3.3 Soll-Zustand

3.3.1 Projektdurchführung

Projekte werden von Partnern, Händlern und der Firma zusammen durchgeführt. Sie können von allen in Abbildung 3–1 gezeigten Partnern ausgehen. Jedes Projekt muß zuerst vorgeschlagen werden. Dies gilt sowohl für Partner als

auch für interne Mitarbeiter. Es wird ein Projektantrag ausgefüllt, um sicherzustellen, daß die Grunddaten des Projekts schon gesammelt wurden. Die Projektvorschläge werden nun von einem Vertriebsmitarbeiter gesichtet und verglichen mit schon vorhandenen Projekte, falls es sich um ein neues Projekt handelt, wird es in die Projektdatenbank übernommen. Nun wird es in eine Projektkategorie A, B oder C eingestuft. (siehe auch 3.2.1.2). Diese Einstufung geschieht manuell und kann sich im Verlauf des Projekts ändern. Es wird nun je nach Größe und Art des Projekts ein Projekttemplate gewählt. Dadurch werden mehrere Milestones vorgegeben (siehe 3.2.1.2). Milestones bestehen aus dem Eintrittsdatum, einer textuellen Beschreibung und einem Milestonetyp, der die Phase definiert. Der Entscheidungsmilestone muß die zu erwartenden Verkäufe enthalten. Für jeden Milestone müssen weiterhin Aktivitäten geplant werden, die für die Erreichung des Ziels notwendig sind. Typische Aktivitäten werden automatisch vorgeschlagen und können übernommen werden. Aktivitäten bestehen aus einer textuellen Beschreibung, einem Datum, der dafür verantwortlichen Person und einem Status. Sie können geplant sein oder schon stattgefunden haben. Weiterhin muß der Mitarbeiter eine Checkliste ausfüllen, die nach einer Entscheidungstabelle die objektive Wahrscheinlichkeit des Projekts beurteilt. Hierzu würden Fragen gehören wie “Kennen wir die Ansprechpartner?” oder “Hat der Kunde einen zwingenden Handlungsbedarf?” (Eine mögliche Beispieltabelle ist in Anhang A). Damit ist das Projekt angelegt.

Im Verlauf des Projekts kommen Aktivitäten hinzu, andere werden stattfinden oder verschoben und es werden Milestones erreicht. Die Entscheidungstabelle wird sich auch ändern. Somit dokumentiert das System immer den aktuellen Status des Projekts mit. Es wird für jede Änderung eine kurze textuelle Beschreibung gefordert, die in die Historie eingeht. Damit ist ersichtlich, wer wann was geändert hat und welchen Verlauf das Projekt genommen hat.

3.3.1.1 Sicht des Vertriebsmitarbeiters

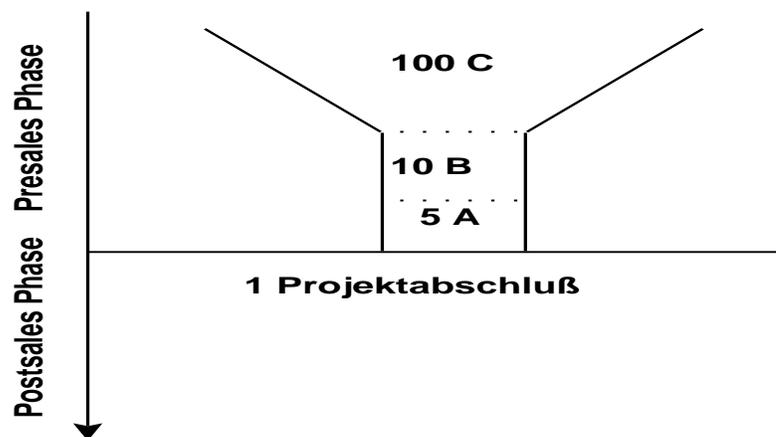
Der Mitarbeiter sieht mit Hilfe des Projektinformationssystems sofort in welcher Phase sich ein Projekt befindet und den aktuellen Projektstatus. Außerdem werden nicht so erfahrenen Vertriebsmitarbeitern Hilfestellungen angeboten in

Form von vorgeschlagenen Aktivitäten für jede Phase. Außerdem hilft ihm die Checkliste mit Fragen, die Gewinnwahrscheinlichkeit selber einzuschätzen und eventuell Aktivitäten einzuleiten, die durch die Antwort auf die Fragen impliziert werden. So kann z.B. eine negative Antwort auf die Fragen “Kennen wir die Entscheider?” den Mitarbeiter implizit darauf hinweisen, daß es notwendig ist, herauszufinden wer die Entscheider sind.

3.3.1.2 Sicht des Vertriebsleiters

Für den Vertriebsleiter ist eine andere Fragestellung interessant. In Abbildung 3-4 ist das Trichtermodell dargestellt. Die eingetragenen Werte sind fiktiv. Es beschreibt den Verlauf von Projekten. Damit ein Projektabschluß zustande kommt, müssen 100 C Projekte angefangen werden. Daraus werden nur 10 Projekte zu B Projekten, wiederum kommen davon nur 5 als A Projekte heraus. Von diesen 5 A Projekten kommt nur eins zum Abschluß. Der Vertriebsleiter möchte nun wissen, wo er steht. Einerseits muß das Verhältnis C:B:A:Abschluß ermittelt werden, andererseits möchte er wissen, ob genügend C Projekte generiert wurden, damit später ein Projektabschluß zustande kommt. [41]

Abbildung 3-4 Trichtermodell



Weiterhin möchte der Vertriebsleiter die eingegebenen Daten seiner Mitarbeiter auswerten. Dazu gehört das Generieren von Forecast und das Tracking.

3.3.1.3 Vertriebsprozeß und Projekttemplates

Da sich nach einer gewissen Zeit der Prozeß, wie man am besten ein Projekt durchführt, etablieren wird, können Projekttemplates definiert werden. Sie enthalten typische Milestones und dazugehörige Aktivitäten. Bei der Erstellung eines Projekts kann nun als Grundlage ein Projekttemplate genommen werden und die typischen Aktivitäten und Milestones als "Skelett" übernommen werden. Die Eingabe von Projekttemplates ist identisch mit der Eingabe von normalen Projekten. Damit ist es möglich, einen Vertriebsprozeß in dem System abzubilden.

Es kann mehr als ein Projekttemplate geben. Damit können verschiedene Projektabläufe modelliert werden. Bei der Neueingabe eines Projekts kann entweder ein Template als Grundlage gewählt oder auch ein leeres Projekt kreiert werden. Falls ein Projekttemplate die Grundlage des neuen Projekts ist, werden alle Milestones des Templates in das neue Projekt übernommen. Die im Projekttemplate vorhandenen Aktivitäten können in das neue Projekt übernommen werden, es besteht jedoch kein Zwang dazu. Das Datum der Aktivitäten und Milestones muß natürlich dem Projektplan für das neue Projekt angeglichen werden.

3.3.2 Aktivitäten

Aktivitäten sind Dinge, die erledigt werden müssen, um einen Milestone zu erreichen. Hierzu können Meetings, Workshops oder Demos zählen. Eine Aktivität kann aber z.B. auch sein "Aufbau einer Beziehung mit dem Geschäftsführer". Somit deckt man mit Aktivitäten nicht nur organisatorische Aktivitäten des Projekts ab sondern auch politische Aktivitäten ab. Alle möglichen Aktivitätstypen sind noch zu definieren.

Eine Aktivität besteht aus einer Beschreibung, einem Aktivitätstyp, einer Person die dafür zuständig ist, einen festen Zeitpunkt (Deadline) bis wann die Aktivität stattgefunden haben muß, einem Breakoff Datum, ab dem die Aktivität keinen Sinn mehr macht, einem Kommentar der bei Erledigung der Aktivität ausgefüllt wird und einem Statusfeld (geplant, erledigt, abgesagt).

Aktivitäten können auch mit Erinnerungen versehen werden als Gedächtnisstütze für den Mitarbeiter.

3.3.3 Weitere Punkte

Da der Händler keine herausragende Funktion in dem Projekt hat, wird ab hier nur noch von Partnern gesprochen. Nur in Fällen, die nur auf den Händler zutreffen, wird noch der Begriff Händler benutzt.

Das System dient auch der Kommunikation zwischen Apple und seinen Partnern. Wichtig ist, daß Partner und Apple zusammen Zugriff auf eine Projektdatenbank haben. Natürlich sind diese Daten mit Zugriffsrechten versehen, damit die Partner nicht an vertrauliche Informationen gelangen können. Der Mehrwert einer solchen Projektdatenbank liegt auf der Hand. Jeder Beteiligte kann sich sofort über den aktuellen Stand eines Projekt informieren. Die Koordination zwischen den Partnern und Apple wird einfacher.

3.3.4 Reportgenerierung

Tracking und Forecast basieren auf den eingegebenen Entscheidungsmilestones. Hierbei werden beim Forecast alle geplanten Verkäufe von Projekten in einem gewissem Zeitraum addiert und nach Produktkategorien aufgeschlüsselt. Außerdem werden der gewichtete und der ungewichtete zu erwartende Umsatz errechnet. Der gewichtete Umsatz ergibt sich durch die Einbeziehung der Eintrittswahrscheinlichkeit (subjektiv und objektiv) aus dem ungewichteten Umsatz. Für weitere Methoden der Absatzprognose siehe [38] oder [39].

Beim Tracking werden die eingegebenen Verkäufe der Projekte eines Mitarbeiters über einen Zeitraum in der Vergangenheit, für den Report herangezogen.

Reportingfunktionen wie die Übersicht der geplanten oder getätigten Verkäufe, aufgeschlüsselt nach Projekt und Monat, sollen auch Bestandteil des Systems sein (Abbildung 3–5). In Abbildung 3–6 ist ein weiterer möglicher Projektreport gezeigt.

Abbildung 3-5 Projektreport nach Produktkategorie

Produkt	Planung					Realisierung				
	Jan	Feb	März	April	...	Jan	Feb	März	April	...
highend	1	2	1	2		0	1	3	4	
midrange	4	5	4	5		3	4	6	7	
lowend	5	6	8	9		7	8	7	8	
Powerbooks	3	4	6	7		5	6	5	6	
Monitore	4	5	5	6		4	5	6	7	

Abbildung 3-6 Projektreport mit allen Grunddaten

Nr.	Name	Kunde	Gesamt -budget	Apple Anteil	Entschei- -dung	Wahr- scheinlich keit	Phase
1	Anzeigen -system 2000	G&J	3 M DM	100kD M	1.3.98	60 %	III

3.3.5 Motivation für die Benutzer

- **Händler** müssen Projekte anmelden, um Unterstützung der Großkundenabteilung von Apple zu bekommen. Dies ist eventuell auch mit Rabatten für das spezielle Projekt verbunden. Für weitere Informationen siehe den Punkt *Partner*.
- **Partner** profitieren durch die Eingabe ihrer Aktivitäten indirekt. Da das System als Kommunikation zwischen Apple und Partner dient, liegt es in ihrem Interesse, Apple durch Eingabe der Aktivitäten zu informieren. Weiterhin haben sie auch Zugriff auf die Aktivitäten, die Apple für dieses Projekt geplant hat. Es liegt also in ihrem Interesse, das System zu benutzen. Auch sie profitieren von der Erinnerungsfunktion, die sie sofort auf irgendwelche Änderungen in den Projekten hinweist oder an fällige Aktivitäten erinnert.

- **Mitarbeiter** haben durch das System eine Hilfestellung für die Projektdurchführung (z.B. Projekttemplates, siehe Abschnitt 3.3.1.3 auf Seite 19). Weiterhin können sie sich sehr schnell einen Überblick über die laufenden Projekte machen. Die automatischen Erinnerungen an bestimmte Aktivitäten helfen dem Mitarbeiter, die geplanten Aktivitäten auch rechtzeitig zu organisieren und durchzuführen. Außerdem orientiert sich ein Teil ihres Verdienstes an ihrem geleisteten Umsatz, der durch das System bestimmt wird.

3.4 Weitere Bemerkungen

Da die CA Abteilung verschiedene Markt- und Kundensegmente bearbeitet, sind auch die Anforderungen an ein Vertriebsinformationssystem unterschiedlich. So differiert das Einkaufsverhalten von Lehr- und Forschungseinrichtungen von dem der Industrie. Auch sind die Entscheidungswege bei Universitäten länger.

3.5 Liste von weiteren Requirements

- Erinnerungsfunktion für alle Aktivitäten
- Da nicht alle Daten für alle Mitarbeiter bzw. Partner sichtbar sein sollen, muß eine Benutzerverwaltung mit Authentisierungsschema existieren.
- Für jedes Eingabefeld soll es eine ausführliche Erklärung geben, die dem Mitarbeiter bei der Eingabe hilft
- Für jede Seite soll es eine ausführliche Hilfe geben.
- An jedes Projekt kann man außer Notizen auch noch Dokumente anhängen. Das gleiche gilt für Accounts.

3.6 Use Cases

Um den Anwendungsbereich zu beschreiben wird die Methode der Use-Cases angewandt. Ein Use-Case entspricht einem Anwendungsfall, wie z.B. der Eingabe eines Projekts. Bei jedem Anwendungsfall wirken ein oder mehrere Akto-

ren mit. Aktoren sind alles, was sich außerhalb des Systems befindet und mit ihm interagiert. Sie beschränken sich nicht auf Menschen sondern können auch andere Systeme oder Anwendungen sein. Aktoren sind Initiator des Anwendungsfalls oder werden während der Abarbeitung mit einbezogen. Ein großer Vorteil dieser Analysemethode ist die gute Verständlichkeit für den Auftraggeber, der sich leicht in die verschiedenen Anwendungsfälle hineindenken kann, ohne tief in die objektorientierten Techniken Einblick zu haben. Jeder Anwendungsfall wird in einem Diagramm modelliert und zusätzlich textuell beschrieben. Für die Requirementsanalyse wird das Use-Case Modell der Spezifikation der Unified Modeling Language erstellt ([34],[35], [36] und [37]). Das Ziel ist, den Anwendungsbereich des Systems klar abzustecken und eine strukturiert Beschreibung der Applikation zu erarbeiten, welche dann in das Design eingeht.

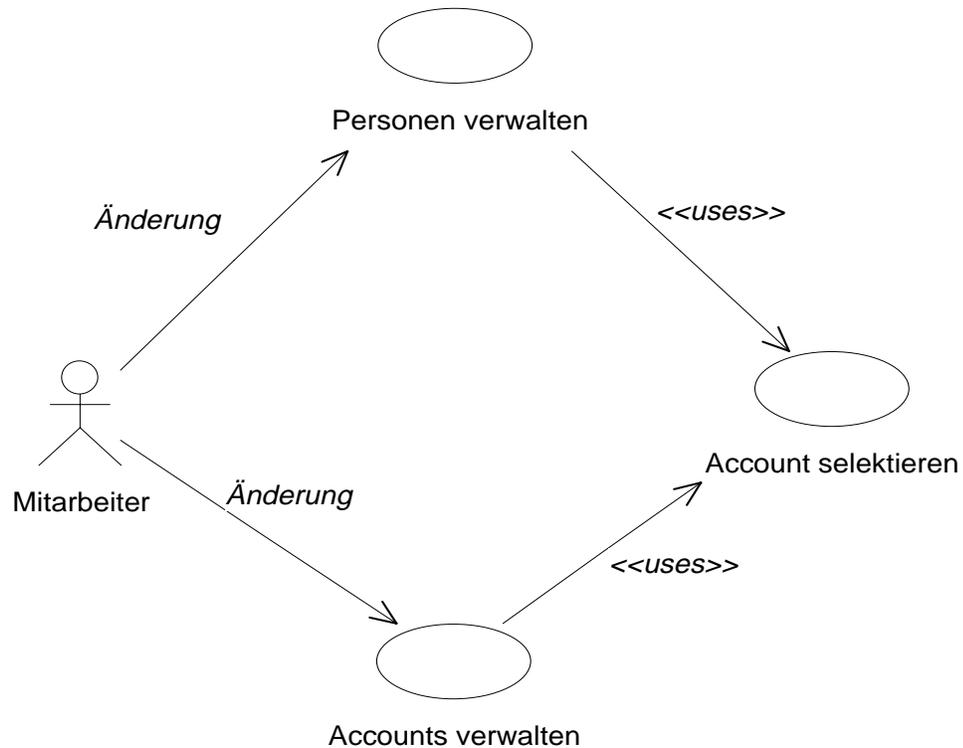
In den Use Cases wird der Begriff *Person* verwendet. Eine Person steht für Mitarbeiter von Apple, Mitarbeiter der Partnern bzw. Mitarbeiter beim Kunden. Eine *Person* ist kann Benutzerrechte haben und somit zum Benutzer werden. Die Benutzerverwaltung ist aber in die Personenverwaltung mit integriert.

Es wurde kein Unterschied gemacht zwischen Eingabe/Löschen/Änderung. Deshalb ist der Use Case “....verwaltung” gleichbedeutend mit Eingabe/Löschen/Ändern. Das gilt nicht für die Projektverwaltung, Milestoneverwaltung und Aktivitätenverwaltung.

Abbildung 3-7

Use Case Diagram Accounts

Use Case Diagram



Use Case 1. Personen verwalten

• Use Case	Personen verwalten
• Vorbedingung	Bei Änderungen muß die Person schon existieren.
• Beschreibung	1a) eine Person wird gesucht. 1b) ein Account wird selektiert. 2) die Personendaten werden geändert oder neu eingegeben (inkl. Benutzerdaten).
• Nachbedingung	Die Personendaten sind in der Datenbank gespeichert.
• Daten	Personendaten
• Schnittstellen	Personenverwaltungsmaske, Personensuchmaske, Accountselektionsmaske

Use Case 2. Accounts verwalten

• Use Case	Accounts verwalten
• Vorbedingung	Bei Änderungen muß der Account schon existieren.
• Beschreibung	1a). Ein Account wird selektiert. 2) Accountdaten werden geändert oder neu eingegeben.
• Nachbedingung	Accountdaten sind in der Datenbank gespeichert.
• Daten	Accountdaten
• Schnittstellen	Accountselektion, Accountverwaltungsmaske

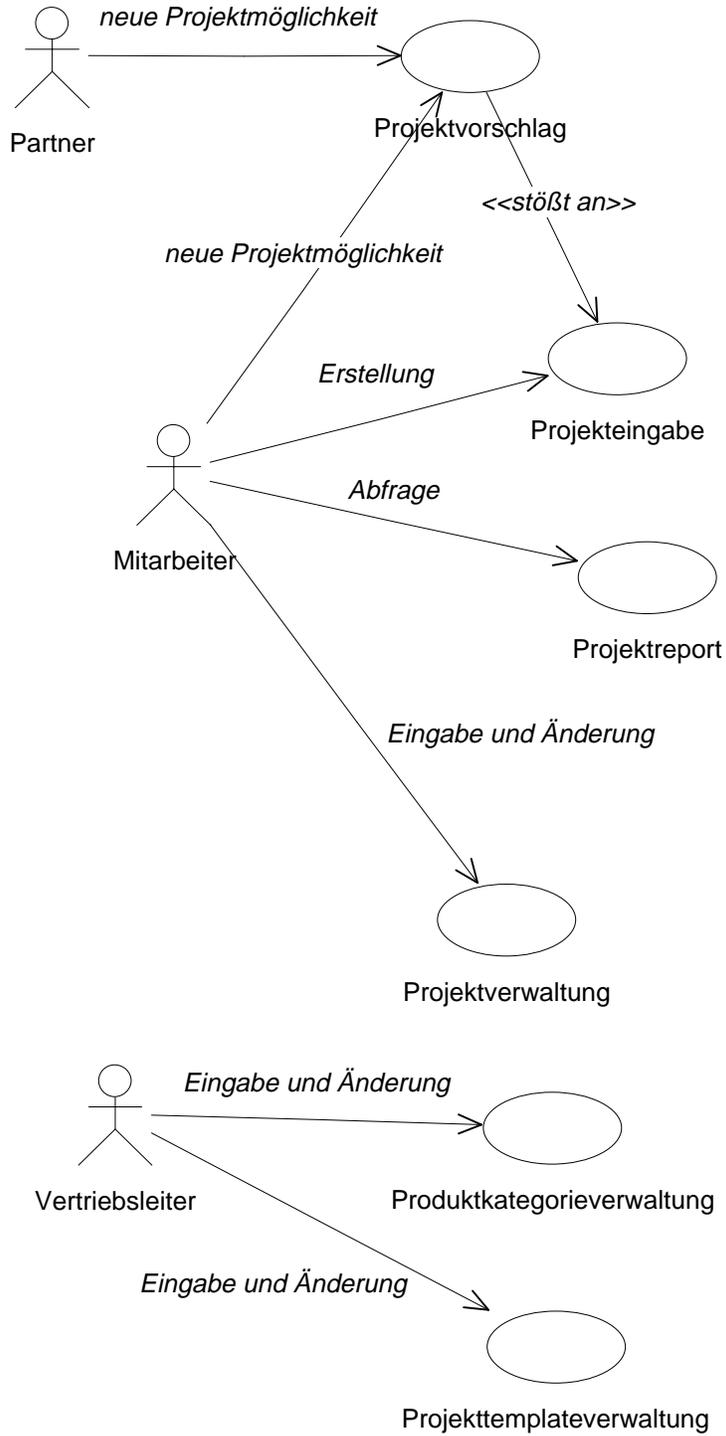
Use Case 3. Accounts selektieren

• Use Case	Accounts selektieren
• Vorbedingung	Account ist vorhanden
• Beschreibung	1. Mittels Accountdaten wird eine Selektion vorgenommen. 2. Die Selektion kann noch rekursiv verfeinert werden. 3. Die Selektion wird die aktuelle Accountselektion oder kann für späteren Gebrauch gespeichert werden.
• Nachbedingung	Eine Accountselektion wurde festgehalten
• Daten	Accountdaten
• Schnittstellen	Accountselektion.

Abbildung 3-8

Use Case Diagram Projektverwaltung

Use Case Diagram Projektverwaltung



Use Case 4. Projektvorschlag

• Use Case	Projektvorschlag
• Vorbedingung	keine
• Beschreibung	<ol style="list-style-type: none"> 1. Ein Partner (z.B. Händler) oder ein Mitarbeiter hat eine Projektmöglichkeit entdeckt und trägt die Rahmendaten des Projekts wie z.B. Zeitpunkt, Vorschläge von weiteren Partnern, Volumen, Kunde etc. ein. 2. Der Projektvorschlag wird in die Datenbank gespeichert.
• Nachbedingung	Ein Projektvorschlag wurde angenommen und einem Mitarbeiter zugewiesen.
• Daten	Projektrahmendaten, Name des Mitarbeiters von Apple
• Schnittstellen	

Use Case 5. Projekttemplateverwaltung

• Use Case	Projekttemplateverwaltung
• Vorbedingung	Das Projekt, welches Projekttemplate werden soll, existiert bereits.
• Beschreibung	<ol style="list-style-type: none"> 1. das Projekt, auf dem das Projekttemplate basiert wird ausgewählt. 2. ein Name wird festgelegt. 3. das Projekttemplate wird in der Datenbank gespeichert.
• Nachbedingung	Ein Projekttemplate wurde erstellt.
• Daten	Projektdaten
• Schnittstellen	Projektauswahlmaske, Projekttemplateeingabe

Use Case 6. Projekteingabe

• Use Case	Projekteingabe
• Vorbedingung	Projektvorschlag ist eingegangen. (Siehe auch Use Case Diagram Projekteingabe verfeinert)
• Beschreibung	<ol style="list-style-type: none">1. Alle Projektvorschläge für das gleiche Projekt werden gesichtet und einem Partner zugewiesen.2. Projektdaten werden eingegeben und die Verknüpfung zu den Accounts und den Personen erstellt. Es muß eine Suche erfolgen über Accounts und über die darin enthaltenen Personen.3. Falls schon weitere Partner bekannt sind, werden sie mit in die Liste der Partner für das Projekt aufgenommen.4. Weiterhin muß die Checkliste mit Fragen beantwortet werden.5. Projektdaten werden in der Datenbank gespeichert.6. Milestoneeingabe und Aktivitätseingabe kann nun beginnen. Je nach Größe des Projekts kann ein Projekttemplate mit vordefinierten Milestones und Aktivitäten aus der Datenbank als Grundlage verwendet werden.
• Nachbedingung	Projekt ist aufgesetzt und als offen markiert.
• Daten	Projektdaten, Partner, Projektvorschläge
• Schnittstellen	Projekteingabe, Accountselektion

Use Case 7. Projektreport

• Use Case	Projektreport
• Vorbedingung	Projekt ist vorhanden
• Beschreibung	<ol style="list-style-type: none"> 1. Ein Projekt wird selektiert 2. Reporttyp auswählen 3. Alle Aktivitäten, die der Benutzer sehen darf, werden in den Report aufgenommen. <p>Spezifische Reports sind v.a. die Auflistung aller geplanten und getätigten Verkäufe in einer Matrix für einen gewissen Zeitrahmen. Zusätzlich ist noch ein Projektreport über alle Grunddaten aller Projekte möglich. Eine Selektion der Projekte über jegliche Grunddaten kann geschehen.</p>
• Nachbedingung	Der Projektreport wurde erstellt
• Daten	Projektdateien, Reporttyp
• Schnittstellen	Projektreportgenerierung, Selektion von Projekten

Use Case 8. Projektverwaltung

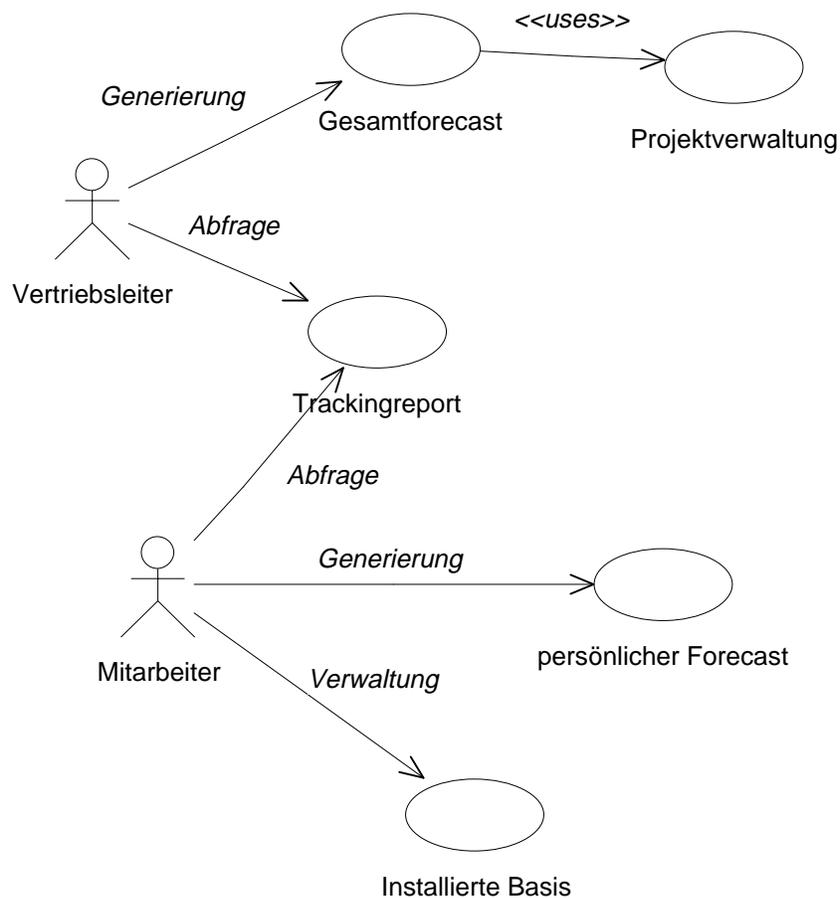
• Use Case	Projektverwaltung (siehe auch Use Case Diagram Projektverwaltung verfeinert)
• Vorbedingung	Projekt existiert
• Beschreibung	<ol style="list-style-type: none"> 1a) Neue Milestones und Aktivitäten werden eingegeben. 1b) Existierende Milestones und Aktivitäten werden verändert.
• Nachbedingung	Ein Milestone oder eine Aktivität wurden erstellt oder verändert.
• Daten	Projektdateien, Milestonedaten, Aktivitätsdaten.
• Schnittstellen	

Use Case 9. Produktkategorieverwaltung

• Use Case	Produktkategorieverwaltung.
• Vorbedingung	
• Beschreibung	1a) Neue Produktkategorien werden hinzugefügt. 1b) Existierende Produktkategorien werden verändert (z.b. Durchschnittspreis).
• Nachbedingung	Eine Produktkategorie wurde erstellt oder verändert.
• Daten	Produktkategoriedaten
• Schnittstellen	

Abbildung 3-9 Use Case Diagram Reporting

Use Case Diagram Forecast/Tracking



Use Case 10. Gesamtforecast

• Use Case	Gesamtforecast
• Vorbedingung	Alle Verkäufe für das Projekt wurden eingegeben und den Entscheidungsmilestones zugeordnet. Außerdem wurde für alle Verkäufe die subjektive Wahrscheinlichkeit eingegeben sowie für das Projekt die Checkliste mit Fragen ausgefüllt (objektive Wahrscheinlichkeit).
• Beschreibung	<ol style="list-style-type: none"> 1. Der Vertriebsleiter wählt alle Vertriebsmitarbeiter aus, die in dem Gesamtforecast enthalten sein sollen. 2. Das System erstellt aus allen Verkäufen für einen spezifizierten Zeitraum einen Forecast.
• Nachbedingung	Forecast für die Abteilung ist erstellt.
• Daten	Verkäufe aller Projekte
• Schnittstellen	

Use Case 11. persönlicher Forecast

• Use Case	persönlicher Forecast
• Vorbedingung	<p>Projektdate mit Verkäufen liegen vor.</p> <p>Außerdem wurde für alle Verkäufe die subjektive Wahrscheinlichkeit eingegeben sowie für das Projekt die Checkliste mit Fragen ausgefüllt (objektive Wahrscheinlichkeit).</p>
• Beschreibung	<ol style="list-style-type: none"> 1. Alle Verkäufe, der vom Mitarbeiter betreuten Projekte, werden aufgelistet (für eine vorgegebenen Zeitraum in der Zukunft) und nach Produktkategorien angezeigt. 2. Der Forecast des Mitarbeiters wird generiert.
• Nachbedingung	Persönlicher Forecast wurde erstellt.
• Daten	Projektdate und Verkäufe
• Schnittstellen	

Use Case 12. Trackingreportgenerierung

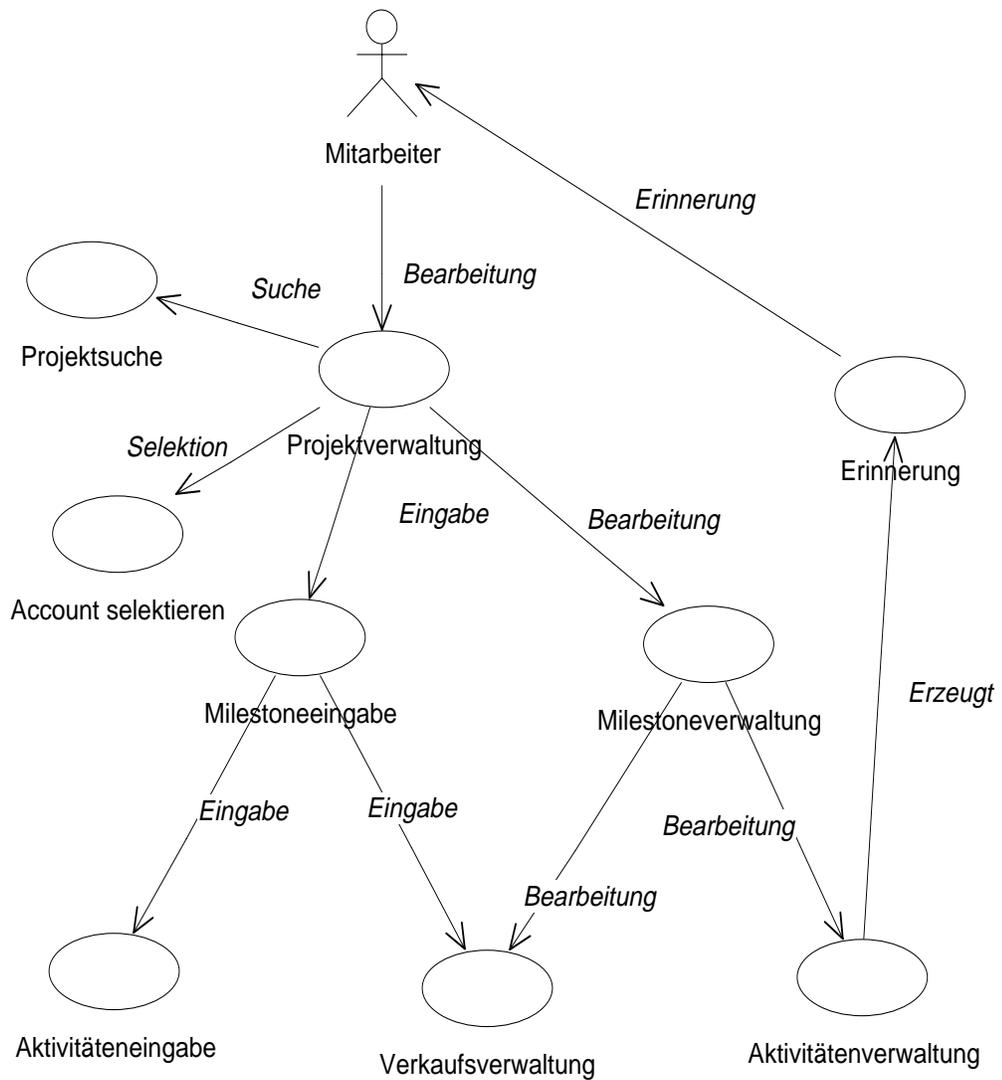
• Use Case	Trackingreportgenerierung
• Vorbedingung	Projektdate mit bestätigten Verkäufen.
• Beschreibung	1. Alle getätigten Verkäufe für die vom Mitarbeiter betreuten Projekte werden für einen gewissen Zeitraum aufsummiert.
• Nachbedingung	Trackingreport wurde generiert.
• Daten	Projektdate und Verkäufe
• Schnittstellen	

Use Case 13. Installierte Basis

• Use Case	Installierte Basis
• Vorbedingung	keine
• Beschreibung	1. die installierte Basis für einen Account wird anhand der getätigten Verkäufe und von schon vorher eingegebenen Werten ermittelt.
• Nachbedingung	Installierte Basis wurde aktualisiert.
• Daten	Projektdate und Verkäufe
• Schnittstellen	

Abbildung 3-10 Use Case Diagram Projektverwaltung verfeinert

Use Case Diagram Projektverwaltung verfeinert



Use Case 14. Projektsuche

• Use Case	Projektsuche
• Vorbedingung	Projekt ist vorhanden
• Beschreibung	1. Anhand von unvollständigen Daten wird eine Menge von Projekten ausgewählt. 2. Ein Projekt wird daraus selektiert.
• Nachbedingung	Projekt wurde selektiert.
• Daten	Projektdateien
• Schnittstellen	

Use Case 15. Erinnerungen

• Use Case	Erinnerung
• Vorbedingung	Aktivitätenverwaltung, Milestoneverwaltung oder Projektverwaltung hat Erinnerungen generiert.
• Beschreibung	1a. Beim Einloggen oder auf Benutzerwunsch werden die aktuellen Erinnerungen an Aktivitäten angezeigt. 1b. Es werden auch noch alle Änderungen an Projekten angezeigt, bei denen der Benutzer eine aktive Rolle spielt oder Beobachter ist. 2. Der Benutzer hat die Möglichkeit die Erinnerungen zu verwerfen oder in die Projektverwaltung für die Bearbeitung der Aktivitäten zu gehen.
• Nachbedingung	Erinnerungen wurden präsentiert und abgearbeitet.
• Daten	Aktivitätsdaten, Projektdateien
• Schnittstellen	

Use Case 16. Milestoneeingabe

• Use Case	Milestoneeingabe
• Vorbedingung	Projekt existiert
• Beschreibung	<ol style="list-style-type: none">1. Projekt wurde selektiert oder neu eingegeben.2. Milestonedaten werden eingegeben.3. Nun müssen die Aktivitäten für diesen Milestone eingegeben werden.4. Falls dieser Milestone ein Entscheidungsmilestone ist, müssen zusätzlich die geplanten Verkäufe eingegeben werden.
• Nachbedingung	Milestone wurde eingegeben.
• Daten	Projektdatei und Milestonedaten
• Schnittstellen	

Use Case 17. Aktivitäteneingabe

• Use Case	Aktivitäteneingabe
• Vorbedingung	Milestone ist vorhanden.
• Beschreibung	<ol style="list-style-type: none">1. die Aktivitätsdaten (wer, wann, wo, was) werden eingegeben.2. Alle Daten sind in der Datenbank gespeichert.
• Nachbedingung	Die Aktivität wurde eingegeben.
• Daten	Aktivitätsdaten
• Schnittstellen	Aktivitätseingabe

Use Case 18. Verkaufsverwaltung

• Use Case	Verkaufsverwaltung
• Vorbedingung	Milestone vom Typ Entscheidungsmilestone ist vorhanden.
• Beschreibung	1. Verkaufsdaten werden eingegeben oder geändert, aufgegliedert nach Produktkategorien (geplante oder getätigte Verkäufe). 2. Planungsdatum wird eingegeben.
• Nachbedingung	Der Verkauf wurde eingegeben.
• Daten	Verkaufsdaten
• Schnittstellen	Verkaufseingabemaske

Use Case 19. Milestoneverwaltung

• Use Case	Milestoneverwaltung
• Vorbedingung	Milestone existiert
• Beschreibung	1. Milestonedaten werden geändert. 2. für jegliche Änderung muß eine textuelle Begründung hinzugefügt werden für die Historie der Änderungen. 3. Der geänderte Milestone und die Historie werden gespeichert.
• Nachbedingung	Milestonedaten wurden verändert gespeichert mit der Historie der Änderung.
• Daten	Projektdaten und Milestonedaten
• Schnittstellen	

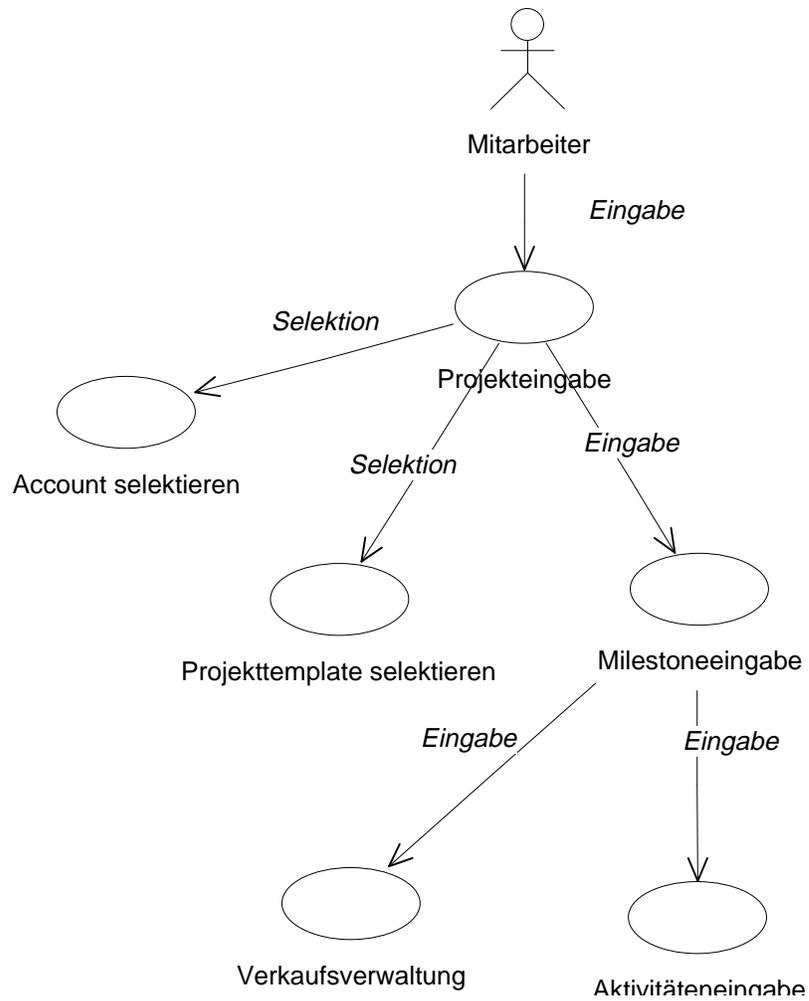
Use Case 20. Aktivitätenverwaltung

• Use Case	Aktivitätenverwaltung
• Vorbedingung	Aktivität ist vorhanden.
• Beschreibung	<ol style="list-style-type: none">1. Die Aktivitätsdaten werden verändert, z.B. der Status (von geplant nach stattgefunden) oder der Termin etc.2. für jegliche Änderung muß eine textuelle Begründung eingegeben werden für die Historie der Änderungen.3. Die geänderte Aktivität und die Historie werden gespeichert.
• Nachbedingung	Veränderte Aktivität und die Historie der Änderungen wurden in der Datenbank gespeichert.
• Daten	Aktivitätsdaten
• Schnittstellen	

Abbildung 3-11

Use Case Diagram Projekteingabe verfeinert

Use Case Diagram Projekteingabe verfeinert



3.7 Szenarios

3.7.1 Projektvorschlag

Ein Partner oder Mitarbeiter schlägt ein Projekt vor und gibt dabei informell die Daten des Projekts an wie z.B. Name der Firma, Projektvolumen und wann der anvisierte Projektabschluß ist. Das System leitet den Vorschlag an Accountmanager von Apple weiter.

3.7.2 Projekterfassung

Basierend auf einen Projektvorschlag wird ein Projekt eingegeben. Vorher ist noch zu prüfen, ob alle Daten korrekt sind und ob das Projekt nicht schon in der Datenbank vorhanden ist. Zu den einzugebenden Projektdaten gehören der Account, die Entscheidungsträger und andere wichtigen Kontakte bei diesem Account, das Projektvolumen in verschiedene Kategorien wie Hardware/Software/Consulting unterteilt, mögliche Partner, die gebraucht werden und natürlich der Zeitpunkt der Kaufentscheidung des Kundens. Eine kurze Beschreibung und eine Einstufung des Projekts in die Kategorien A, B und C soll erfolgen. Weiterhin muß der Mitarbeiter noch Milestones festlegen und dafür Aktivitäten definieren, um die Milestones zu erreichen. Er kann aber auch aus schon existierenden Projekttemplates auswählen und somit schon vordefinierte Milestones und Aktivitäten für das anzulegenden Projekt erhalten. Damit ist die Projekteingabe abgeschlossen.

3.7.3 Projektverwaltung

Der Mitarbeiter möchte eine Aktivität (Workshop beim Kunden) verschieben und einen weiteren Workshop planen. Er sucht das Projekt nach seinem Namen, Kunden oder anderen Projektattributen (Projektleiter, Partner, etc.) und geht in die Projektverwaltung. Die Aktivität Workshop wird verschoben in dem ein neues Datum eingegeben wird. Der Mitarbeiter muß diese Terminänderung mit einem kurzen Kommentar begründen. Danach plant er den zweiten Workshop. Da noch kein Termin feststeht, kann er die Aktivität ungenau planen. D.h anstatt

einem festen Datum wird ein Zeitraum eingegeben wie 10. Kalenderwoche oder Monat Mai. Er trägt außerdem für sich eine Erinnerung ein, die ihn zwei Wochen vor dem geplanten Workshop noch einmal daran erinnern soll.

Alle an dem Projekt Beteiligten erhalten beim nächsten Einloggen eine Notiz, in der diese Änderungen notiert sind. Damit sind sie sofort auf dem aktuellen Stand.

3.7.4 Einloggen in das System

Ein Mitarbeiter oder Partner loggt sich in das System ein. Er wird automatisch mit noch nicht abgearbeiteten Erinnerungen konfrontiert. Dies können Erinnerung an Aktivitäten sein, die noch genauer geplant werden müssen oder Erinnerungen an Aktivitäten bei von ihm betreuten Projekten, die von Dritten (Partnern) eingegeben oder geändert worden sind. Damit ist er sofort auf dem neuesten Stand.

3.7.5 Projektreporting

Ein Mitarbeiter oder ein Partner möchten sich über den Stand eines Projekts genauer informieren. Es wird anhand der Daten ein Projektreport erstellt.

Der Vertriebsleiter möchte wissen, ob er genügend Projekte “in der Pipeline” hat, damit auch in einem Jahr genügend Projekte zum Abschluß kommen. Er läßt sich einen Projektreport generieren, der die Anzahl der Projekte der Kategorie A, B und C aufzeigt. Mit Hilfe seiner Erfahrungsdaten kann er die Situation sofort beurteilen und Entscheidungen treffen.

Der Vertriebsleiter möchte sich über alle seine laufenden Projekte informieren. Dafür läßt er sich einen Projektreport mit allen Grunddaten aller Projekte anzeigen.

3.7.6 Reporting

Der Vertriebsleiter möchte den voraussichtlichen Umsatz im nächsten Quartal schätzen. Er lässt sich vom System für alle seine Mitarbeiter für das nächste Quartal einen Forecast generieren.

Der Vertriebsmitarbeiter möchte wissen, ob er seine Quote im letzten Quartal erfüllt hat. Das System generiert ihm einen Trackingreport, der ihm seine Umsätze aufzeigt.

3.8 Glossar

Abbildung 3–12 enthält das Glossar. Es enthält Begriffsdefinitionen.

Abbildung 3–12 **Glossar**

Begriff	Beschreibung
Account	Ein Account ist ein Bereich bei einer Firma, mit dem Geld verdient wird. Er wird von einem Account manager verwaltet (Vertriebsmitarbeiter). Zu einem Account gehören ein oder mehrere Kontakte. Er kann in einem Organisational Chart einem Teil einer Firma zugeordnet werden.
Aktivität	Um einen Milestone zu erreichen, müssen Aktivitäten durchgeführt werden. Eine Aktivität kann eine Vorführung, ein Meeting oder auch ein Workshop sein. Eine Aktivität hat ein Datum oder auch einen Zeitraum (Kalenderwoche, Monat) in dem sie geplant sein kann.
Bedarf	Der Bedarf eines Kunden wird nach dem erarbeiteten Konzept ermittelt. Er enthält die notwendigen Leistungen (Software/ Hardware/ Dienstleistungen) für die Lösung des Problems.
Entscheidung	Die Entscheidung (Kaufentscheidung) ist der voraussichtliche Zeitpunkt, zu dem sich der Kunde für ein Angebot entscheiden wird.

Abbildung 3–12 Glossar

Begriff	Beschreibung
Milestone	Ein Milestone ist ein Zwischenziel, das zu erreichen ist. Um es zu erreichen werden eine Anzahl von Aktivitäten definiert.
Entscheidungsmilestone	Ein Entscheidungsmilestone ist ein Milestone, der die Kaufentscheidung des Kunden darstellt. Er enthält zusätzlich zu den Milestonedaten noch die Verkaufsdaten (geplante, getätigte) aufgeschlüsselt nach Produktkategorien.
Forecast	Ein Forecast ist der zu erwartende Umsatz in einem gewissen Zeitraum in der Zukunft.
Händler	Ein Händler ist ein Partner der Produkte von Apple an den Kunden verkauft. Apple verkauft keine Produkte direkt an den Kunden.
Historie	Für jegliche Änderungen bei Aktivitäten und Milestones existiert eine Historie der Änderungen.
Keypartner	Ein Keypartner ist ein Partner, der in einem Projekt den größten Anteil (Budget und Zeit) hat.
Konzept	Das Konzept wurde nach der Problemstellung beim Kunden erarbeitet. Es spezifiziert eine Lösungsansatz für das Problem. Aus ihm geht der Bedarf des Kunden hervor.
Kunde	Ein Kunde ist das Synonym für ein Account. Es macht im Text jedoch mehr Sinn, von <i>Kunde</i> zu sprechen.
Organisational Chart	Eine Firma besteht aus mehreren Unterfirmen. Diese Baumstruktur von Holding zu Tochterfirmen u.s.w. graphisch dargestellt sind Orgcharts.
Partner	Ein Partner ist eine Firma, die mit Apple kooperiert, um ein Projekt beim Kunden zum Erfolg zu führen.
Person	Eine Person kann ein Mitarbeiter bei Apple, ein Mitarbeiter beim Kunden oder ein Mitarbeiter bei einem Partner sein.

Abbildung 3–12 Glossar

Begriff	Beschreibung
Phase	Eine Phase ist der Zeitraum zwischen zwei Milestones. Die Milestones müssen nicht direkt aufeinanderfolgen.
Projekt	Ein Projekt ist eine Zusammenfassung von Leistungen und Produkten unter Mitarbeit von Partnern und Apple, um ein Problem des Kunden zu lösen. Es erstreckt sich über eine Zeitachse auf der wiederum mehrere Milestones mit dazugehörigen Aktivitäten verteilt sind. An einem Projekt arbeiten ein oder mehrere Partner.
Projektleiter	Verantwortliche Person für ein Projekt. Der Projektleiter beim Kunden ist der erste Ansprechpartner für alle Beteiligten an einem Projekt. Der Projektleiter beim Keypartner hat den meisten Kontakt zum Projektleiter beim Kunden und ist somit auch eine wichtige Informationsquelle. Bei Apple ist der Projektleiter immer der Accountmanager.
Projektkategorie	Mehrere Phasen können einer Projektkategorie zugewiesen werden. Es gibt drei Kategorien: A, B und C. Die Zuweisung zwischen Phase und Kategorie kann geändert werden.
Produktkategorie	Produkte werden zusammengefaßt in Produktkategorien. Mögliche Produktkategorien bei Apple sind lowend, midrange, highend Systeme, Printer, Powerbooks, Monitore.
Projekttemplate	Ein Projekttemplate ist ein generisches Projekt, welches Milestones und Aktivitäten enthält. Bei der Neueingabe eines Projekts kann ein Projekttemplate als Grundlage genommen werden.
Tracking	Tracking meint das Messen der Leistung der Vertriebsmitarbeiter an der Anzahl in einem Zeitraum in der Vergangenheit verkauften Produkten.

Abbildung 3-12 Glossar

Begriff	Beschreibung
Verkauf	Ein Verkauf besteht aus einer geplanten und einer wirklich getätigten Anzahl an zu verkaufenden Produkten, aufgeschlüsselt nach Produktkategorien. Ein Verkauf wird einem Entscheidungsmilestone zugeordnet.
Vertriebsmitarbeiter	Ein Vertriebsmitarbeiter ist eine Person, die direkt mit dem Kunden Kontakt hat, der Verkäufer also.
Vertriebsleiter	Leiter von mehreren Vertriebsmitarbeitern.
Wahrscheinlichkeit, objektiv	Die objektive Wahrscheinlichkeit für das Gewinnen eines Projekts ergibt sich aus den beantworteten Fragen der Entscheidungstabelle über das Projekt.
Wahrscheinlichkeit, subjektiv	Die subjektive Wahrscheinlichkeit für das Gewinnen eines Projekts setzt der Mitarbeiter aus dem eigenen Gefühl fest.

3.9 Mögliche Attribute der wichtigsten Objekte

3.9.1 Projekt

Tabelle 3-3 Attribute eines Projekts

Name	Beschreibung
Nummer	Eindeutige Nummer, die das Projekt identifiziert
Name	maximal drei Wörter langer Kurzname des Projekts
Beschreibung	Beschreibung des Projekts
Kunde	zu dem Namen des Kunden gehört auch noch seine innere Struktur
Partner	Liste aller Partner für das Projekt
Projektleiter beim Kunden	Daten des Projektleiters beim Kunden
Projektleiter beim Keypartner	Daten des Projektleiters beim Keypartner
weitere Entscheidungsträger und andere Personen	alle anderen Personen beim Kunden, die mit dem Projekt zu tun haben.
Budgetdimension	Budgetdimension des Projekts mit Aufschlüsselung der Anteile (HW, SW, etc).
Notizen	Notizen, die jeder an das Projekt anhängen kann
Attachments	Jegliche Dokumente oder sonstige Dateien, die mit dem Projekt verknüpft sind.

Tabelle 3-4 Attribute einer Aktivität

Name	Beschreibung
Beschreibung	Beschreibung der Aktivität
Aktivitätstyp	Typ der Aktivität wie Workshop, Meeting oder Demo.
Status	geplant oder stattgefunden
Verantwortlicher	Die Person, die für die Durchführung der Aktivität verantwortlich ist.
Datum	Bis wann muß die Aktivität durchgeführt werden.
Breakoff Datum	Ab wann macht die Aktivität keinen Sinn mehr
Notizen	Weitere Notizen, die von jedem an diese Aktivität angehängt werden können.
Erinnerungsinterval	Zeitpunkt, an dem der Benutzer an diese Aktivität erinnert werden möchte.

Tabelle 3-5 Attribute eines Milestones

Name	Beschreibung
Beschreibung	Beschreibung des Milestones
Phase	Gibt die Phase des Milestones an (Abbildung 3-3).
Status	geplant oder stattgefunden
Verantwortlicher	Die Person, die für die Durchführung der Aktivität verantwortlich ist.
Datum	Bis wann muß der Milestone erreicht sein.
Notizen	Weitere Notizen, die von jedem an diese Aktivität angehängt werden können.
Aktivitäten	Aktivitäten, welche zum Erreichen des Milestones notwendig sind.

4. OOHDM

In diesem Kapitel wird ein Überblick über die verwendete Designmethode OOHDM gegeben. Es werden die einzelnen Designphasen besprochen und die Notation und deren Semantik erläutert.

4.1 Einleitung

Die Methode OOHDM (Object Oriented Hypermedia Design Method) ist ein Modell-basierter Ansatz, um große Hypermediaanwendungen zu entwerfen. OOHDM wurde von Gustavo Rossi und Daniel Schwabe entwickelt [1],[2],[3]. Ziel ist, eine Softwareentwicklungsmethode für Hypermediaanwendungen bereit zu stellen. Herausragend ist die weitreichende Form der Navigation, welche das Paradigma Hypertext zur Verfügung stellt. Sie ermöglicht dem Benutzer über eine bestimmte Informationsbasis zu navigieren und dabei komplexe Aufgaben zu erledigen. Die Erstellung von derartigen Applikationen ist ein schwieriges Unterfangen. Das erste Ziel einer solchen Anwendung ist eine gut funktionierende und durchdachte Navigation zu erstellen. Weiterhin unterscheiden sich das User-Interface einer normalen Applikation stark von dem einer Hypermediaapplikation. Es muß z.B. festgelegt werden, welche Events vom Benutzer in den Anzeigenelementen nur lokale Effekte in der Benutzerschnittstelle bewirken und welche Events eine Navigation auslösen. Der Blick auf die Entwicklung von herkömmlichen Applikationen zeigt, daß sich das objektorien-

tierte Softwareengineering bewährt hat. Demnach sollte die Methode für die Erstellung von Hypermediaapplikationen objektorientiert sein und die oben genannten Problembereiche berücksichtigen.

OOHDM gliedert sich in die vier nachfolgend beschriebenen Phasen:

- **Conceptual Design:** Im Conceptual Model wird ein Domänenmodell des Problembereichs erstellt, in dem alle Datenobjekte mit ihren unterschiedlichen Beziehungen modelliert werden. OOHDM benutzt dazu die Syntax von den gängigen objektorientierten Designmethoden wie UML oder OMT.
- **Navigational Design:** Hier werden verschiedene Ansichten auf die Applikation, wie sie im Conceptual Design dargestellt wurden, modelliert. Zusätzlich werden die Navigationsmöglichkeiten, die der Benutzer hat, graphisch modelliert. OOHDM definiert zwei verschiedene Diagramme für das Navigational Model: das Navigational Schema und das Contextual Schema.
- **Abstract Interface Design:** Hier wird spezifiziert welche Benutzerschnittstellenelemente dem Benutzer angezeigt werden. Zusätzlich werden die Auswirkungen von Ereignissen (Benutzerinteraktionen), die nicht der Navigation dienen, festgelegt. Die Datenobjekte aus dem Conceptual Design werden mit den Abstract Data Views verknüpft, um die Relation zu der Datenbasis herzustellen.
- **Implementierung:** Die Implementierung ist in der Methode direkt als letzter Schritt spezifiziert.

Demnach bietet OOHDM für die obengenannten Problembereiche einer Hypermediaapplikation genügend Unterstützung. OOHDM unterstützt die Navigation in zwei Schemata und die Bildung von der Benutzerschnittstelle im Abstract Interface Design. Allem voran wird eine Phase der Definition des Anwendungsbereich gestellt. Die Grundsteine von OOHDM sind nachfolgend beschrieben.[7]

- Navigationsobjekte sind spezielle Ansichten von den Objekten des Conceptual Designs
- Navigationskontexte beschreiben und strukturieren auf einem angemessenen Abstraktionsniveau den Navigationsraum der Anwendung.

- Das Design der Navigation wird vom Design der Benutzerschnittstelle getrennt.
- Es gibt spezifische Designentscheidungen, die erst während der Implementierungsphase gemacht werden müssen und deswegen nicht in der Abstract Interface Designphase realisiert werden.

Die weiteren Unterkapitel gehen näher auf die einzelnen Phasen ein.

4.2 Conceptual Design

Während der Conceptual Design Phase wird ein statisches Objektmodell des Anwendungsgebiets aufgebaut. Zur Erstellung werden die geläufigen Schritte angewandt, wie sie beispielsweise in OMT vorgeschlagen werden. [44] OOHDM legt sich auf keine Beschreibungssprache für das Conceptual Design fest. Es kommen alle gängigen Methoden, wie UML oder OMT, für die Objektmodellierung in Betracht[6]. Für das Objektmodell wird das statische Klassendiagramm benutzt. Als einzige Erweiterung sieht OOHDM die Überführung von nicht-gerichteten Relationen in gerichtete Relationen vor. Diese werden mit Pfeilen gekennzeichnet und entsprechen der Berücksichtigung der Navigationsrichtung zwischen den Objekten beim Navigational Design.

Es soll hier nicht weiter auf das Conceptual Design eingegangen werden, da es den klassischen Objektmodellen entspricht. Es wird hier auf die Dokumentation der Methode OMT [45] oder der Beschreibungssprache UML ([28],[29], [30] und [31]) verwiesen.

Im Conceptual Design wurde festgehalten, aus welchen Objekten die Applikationsgebiet besteht und in welcher statischen Beziehung sie zueinander stehen.

4.3 Navigational Design

Während das Conceptual Design noch mit traditionellen Methoden erstellt wird, geht das Navigational Design auf die spezielle Thematik "Navigation" in einer Hypermediaapplikation ein. Der Schritt beschreibt losgelöst von der Beschrei-

bung der Oberfläche die möglichen Navigationsobjekte, basierend auf den Klassen des Conceptual Designs. Das Navigational Design repräsentiert demnach die verschiedenen Sichten auf die Klassen des Conceptual Design.

Eine spezielle Klasse, der Kontext, beschreibt eine Menge von Objekten, welche durch eine Einschränkung (Query) über die gesamten Objekte spezifiziert wird. OOHDM führt dazu eine eigene Anfragesprache ein. Kontexte können auch ineinander geschachtelt werden. Die verschiedenen Kontextarten, die OOHDM spezifiziert, sind weiter unten beschrieben.

Während der Entwicklung des Navigational Designs werden folgende Aspekte untersucht:

- Welche Objekte spielen bei der Navigation eine Rolle, welche Attribute haben sie und wie sind die Relationen zwischen den Objekten?
- In welchem Kontext wird der Benutzer navigieren?
- Hat das Objekt unterschiedliche Attribute je nach dem in welchem Kontext der Benutzer sich befindet?
- Von welchem Typ sind die Verbindungen zwischen den Objekten und über welche Zugriffsstruktur werden sie realisiert? (Links, Indizes etc.)
- Was passiert, wenn der Benutzer zwischen Objekten hin- und herspringt, d.h. den Kontext wechselt?

Wie schon weiter oben in den Grundbausteinen von OOHDM beschrieben, werden im Navigational Design bestimmte Ansichten auf Klassen des Conceptual Designs erstellt. Dies geschieht im Navigational Schema. Ein weiterer Grundstein, die Navigational Kontexte, werden im Contextual Schema definiert. Das Navigational Design wird demnach in zwei Schemata spezifiziert:

- Das **Navigational Schema** stellt eine Sicht vom Standpunkt der Navigation auf die Klassen des Conceptual Schemas dar. Es bedient sich der gleichen Notation und der gleichen Konzepte. Zusätzlich werden aber spezielle Navigationsklassen eingeführt wie Indizes, Links und Knoten. Weitere Änderungen können das Wegfallen von Klassen bzw. das Anreichern von Klassen durch fremde Attribute (welche durch z.B. 1-1 Relationen zugreifbar sind) sein. Navigationsklassen können ein Subset der

Attribute ihrer entsprechenden Klassen im Conceptual Schema enthalten. Sie repräsentieren somit eine spezielle Sichtweise auf die Klasse des Conceptual Schemas, welche sich in Abhängigkeit von einem Kontext ändern kann (siehe dazu den nächsten Punkt).

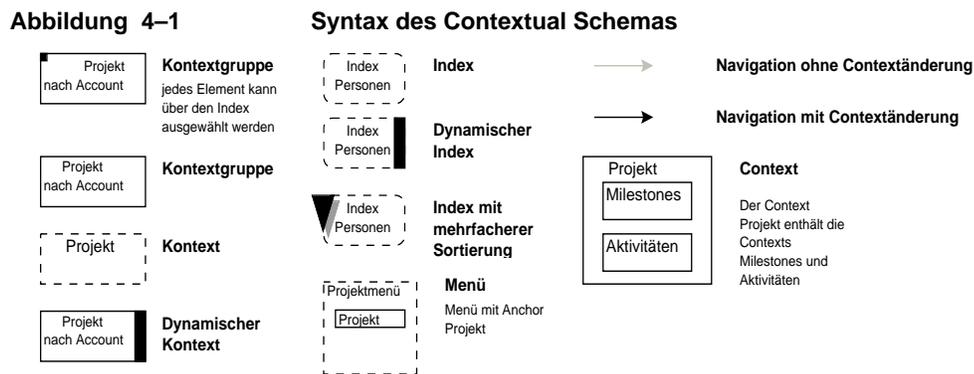
- Das **Contextual Schema** erläutert und erklärt den Navigationsraum mit Hilfe von Kontextklassen. Ein Kontext ähnelt einer Sammlung von Elementen. Weiterhin kann ein Kontext geschachtelt werden.

Es gibt sechs verschiedene Kontextarten:

- **Klassenbasierter Kontext:**
Alle Objekte gehören der selben Klasse C an und können durch ein gemeinsames Attribut ausgewählt werden. Ein Beispiel wären alle Partner des Accounttyps "Händler". Der Accounttyp ist Attribut der Klasse Partner.
- **Klassenbasierte Kontextgruppe:**
Dies ist eine Menge von klassenbasierten Kontexten, welche dadurch gebildet wird, daß ein Parameter der Klasse alle möglichen Werte annimmt. Ein Beispiel wäre Partner nach Accounttyp. Der Accounttyp ist Attribut der Klasse Partner.
- **Link-basierter Kontext:**
Auch hier sind die Objekte von einem Klassentyp C und werden über eine 1-n Relation selektiert. Z.B. alle Kunden von Accountmanager Herrn Schmidt. Kunde zu Accountmanager ist eine 1-n Relation.
- **Link-basierte Kontextgruppe:**
Dies ist eine Menge von Link-basierten Kontexten, die aus einer 1-n Relation entsteht, indem nicht ein Quellobjekt festgelegt wird, sondern diese Relation für alle Quellobjekte aufgelöst wird. Z.B. Account nach Accountmanager. Kunde zu Accountmanager ist eine 1-n Relation.
- **Enumerierte Kontextklasse:**
Diese Kontextklasse beschränkt sich nicht nur auf Objekte einer Klasse, sondern enthält Objekte aus verschiedenen Klassen. Ein Beispiel hierfür wären alle Notizen und Erinnerungen.
- **Dynamische Kontextklasse:**
In die dynamische Kontextklasse können während der Navigation die in ihr enthaltenen Objekte gelöscht oder neue hinzugefügt werden.

In Abbildung 4–1 ist die Syntax des Contextual Schemas angegeben. Das Schema besteht aus den Komponenten:

- **Zugriffsstrukturen:** Ein Zugriffsstruktur wie z.B. ein Index oder eine Guided Tour erlaubt es dem Benutzer über eine Auswahl von mehreren Objekten zu navigieren, sie auszuwählen und damit in den Context zu übernehmen, welcher der Zugriffsstruktur nachfolgt. Der Benutzer muß immer wieder zur Zugriffsstruktur zurückkehren, um ein anderes Objekt zu selektieren.
- **Context:** die verschiedene Kontextarten wurden oben beschrieben.
- **Link:** die Verknüpfung von verschiedenen Kontexten.



Zusätzlich zu Abbildung 4–1 muß angemerkt werden, daß alle Kontexte und Kontextgruppen, die über einen Index gesteuert werden ein kleines schwarzes Rechteck links oben in der Ecke haben. Auf dem Diagramm ist eine Kontextgruppe mit dieser Notation dargestellt.

Die sechs Kontextarten werden zusammengefaßt in drei verschiedene Notationen. Hierbei wird zwischen Kontext und Kontextgruppe differenziert. Die Unterscheidung der Link-basierten Kontexte und der einfachen Kontexte ist implizit. Liegt dem Kontext eine Relation zugrunde so ist er Link-basiert, ansonsten ist er Klassen-basiert.

Für jeden Navigational Kontext muß außerdem noch eine Spezifikation geschrieben werden, in der die Abfragen, welche den Kontext einschränken, spezifiziert sind. Die genaue Syntax ist in [7] beschrieben.

4.4 Abstract Interface Design

Dem Navigational Design folgt der Entwurf der Benutzerschnittstelle. Hier wird definiert wie die Navigationsobjekte dargestellt werden. Es muß unterschieden werden zwischen Ereignissen vom Benutzer, die navigationsspezifisch sind, und Ereignissen, die eine Veränderung der Darstellung von Navigationsobjekten bewirken. Die Trennung von Navigation und Benutzerschnittstelle erlaubt es, für ein Navigationsdesign mehrere Benutzerschnittstellen zu entwerfen. Es wird eine klare Trennung von Darstellung und Applikationslogik forciert.

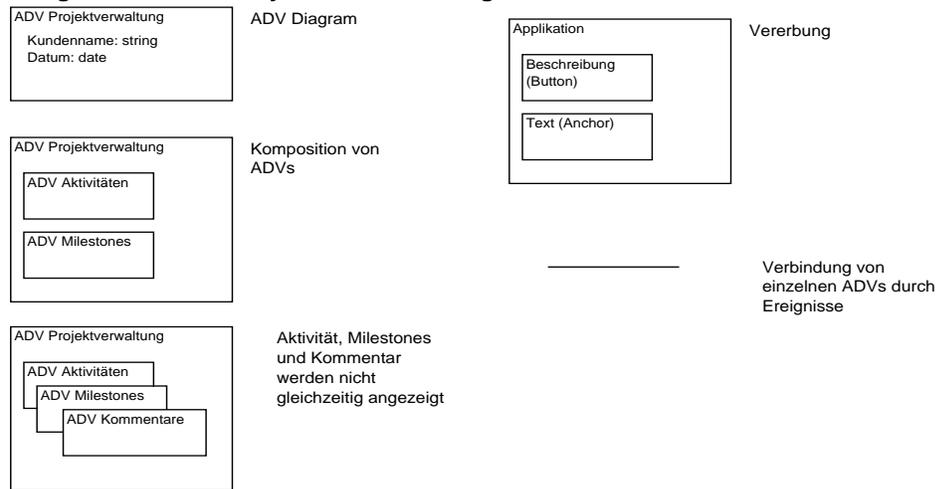
OOHDM benutzt die Notation der Abstract Data Views (ADV) für das Design des User-Interfaces. ADVs haben einen internen Zustand und eine Schnittstelle. Sie können auf externe (vom Benutzer erzeugte) Ereignisse reagieren. Benutzerereignisse lösen entweder einen "Navigationsschritt" aus oder beeinflussen die Anzeige. Beispielsweise könnte ein Objekt in einer Grobansicht und einer Detailansicht angezeigt werden. Der Wechsel zwischen den Ansichtsarten erfordert ein Event vom Benutzer. Dieser Event löst aber keine Navigation aus, sondern beschränkt sich auf die Darstellung. Falls aber nun aus der Detailansicht ein Link zu einer anderen Klasse verfolgt wird, handelt es sich hierbei um eine "Navigationsschritt".

Die Ereignisse, auf die ein ADV reagieren kann, sind beschränkt auf die vom Browser zur Verfügung gestellten Ereignisse. Hierzu gehören auf jeden Fall `MouseClicked`, `MouseOver`, `MouseUp`, etc.

ADV's können durch Contains-Beziehungen oder durch Vererbung in Relation gebracht werden. Abbildung 4-2 zeigt die Notation eines ADV Diagramms.

Abbildung 4–2

Syntax von ADV Diagrammen



Die Dynamik der Benutzerschnittstelle wird mittels ADVCharts beschrieben. Sie zeigen den Einfluß den Ereignisse des Benutzers (MouseClicked, etc.) auf die Darstellung der Interfaceobjekte haben. Ereignisse, welche die Navigation beeinflussen, werden vom Navigational Design abgedeckt.

Weitere Literatur bezüglich OOHDM sind [1], [2], [3], [4], [5], [6], [7], [8], [9] und [10].

5. Design

In diesem Kapitel ist die Designphase des Projekts beschrieben. Sie gliedert sich in die drei Phasen Conceptual Design, Navigational Design und Abstract Interface Design. Die verwendete Methode (OOHDM) ist in Abschnitt 4 genauer erklärt.

5.1 Conceptual Design

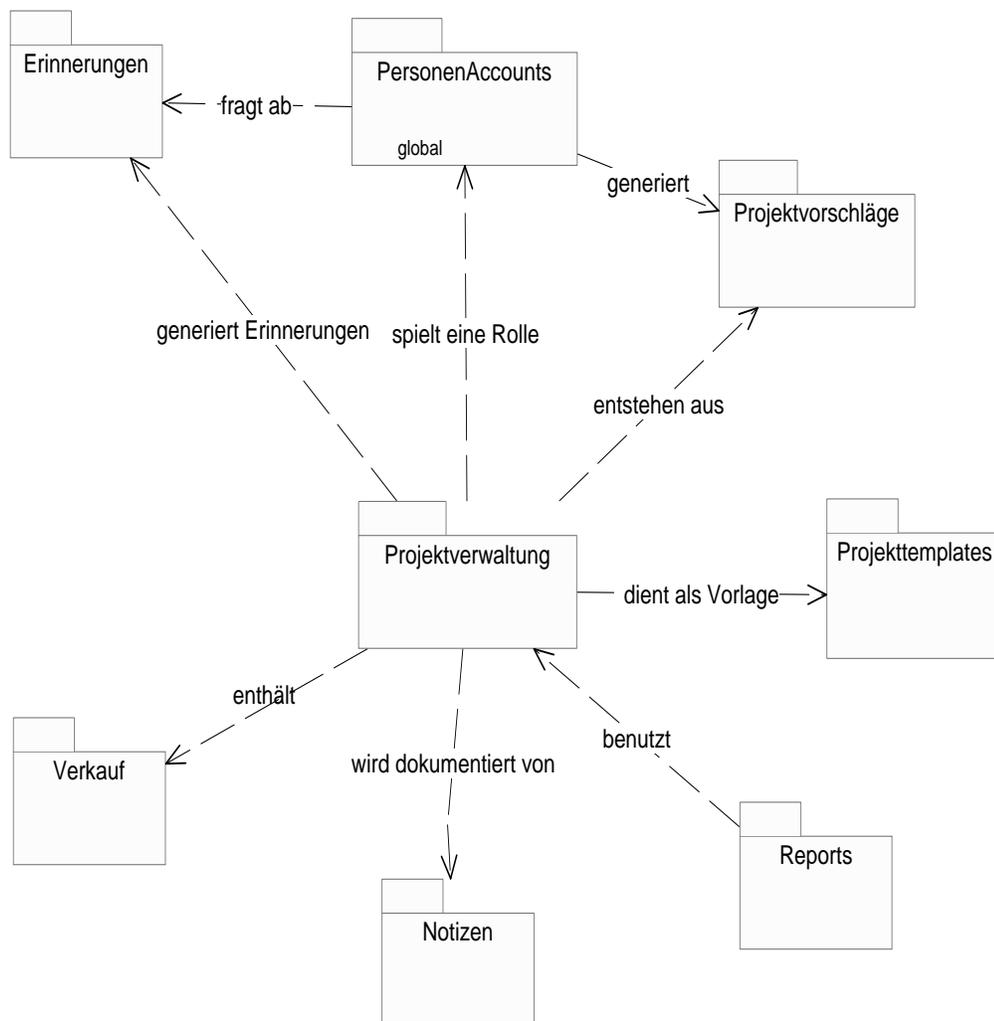
Im Conceptual Design wird der statische Aspekt des Anwendungsbereichs beschrieben. Die verwendete Notation für das Conceptual Design ist UML, da UML ein Standard ist. Sowohl OMT als auch UML werden als geeignete Notation empfohlen. Das Conceptual Design verwendet objektorientierte Konzepte zur Modellierung, wie Vererbung, Aggregation und Assoziationen.

Die Applikation besteht aus acht Subsystemen. Im Zentrum der Applikation steht, wie die Abbildung 5–1 verdeutlicht, die Projektverwaltung. Sie enthält die Milestones und Aktivitäten (hier nicht dargestellt). Alle weiteren Module erweitern die Funktionalität der Projektverwaltung. Ein Projekt generiert Erinnerungen, die von Personen abgefragt werden. Jede Person, die in einem Projekt mitarbeitet, nimmt eine bestimmte Projektrolle ein. Die Rechte, welche damit verbunden sind, addieren sich zu den Rechten, welche die Person durch ihre Personenrolle erlangt. Jedem Projekt geht ein Projektvorschlag voraus, der von einer Person eingegeben wird. Einem Projekt kann noch optional ein Projekt-

template zugrunde liegen. Das Reportmodul generiert aus allen Projektdaten einen benutzerspezifischen Report. Das Projekt wird mit Hilfe vom Modul Notizen dokumentiert. Weiterhin können für jedes Projekt Verkäufe definiert werden.

Abbildung 5-1 Class Diagram Übersicht

Class Diagram: Subsysteme



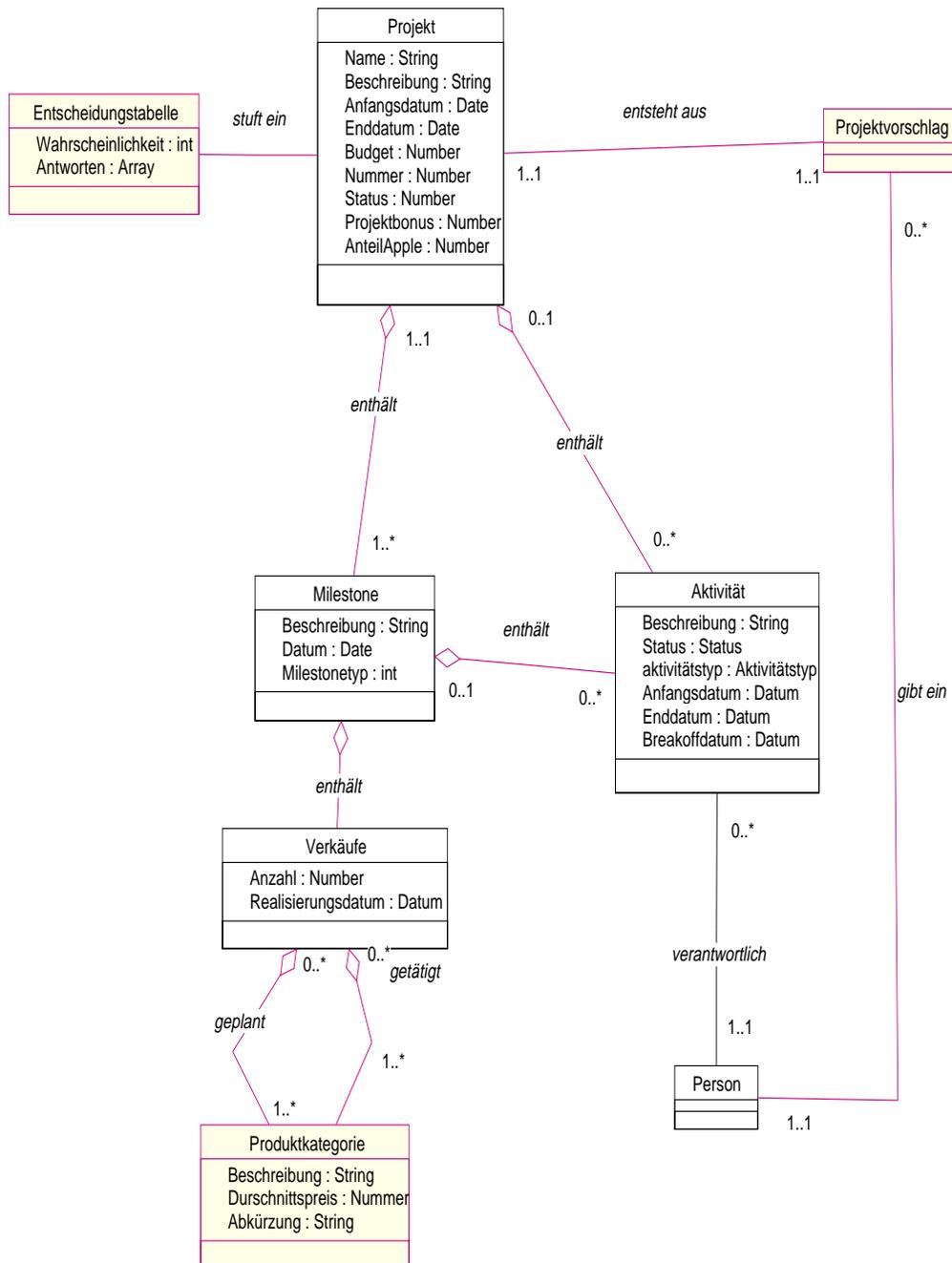
In Abbildung 5–1 ist die Übersicht der verschiedenen Module gegeben. Notizen und Verkäufe sind sehr eng mit der Projektverwaltung verknüpft. Sie haben wenig Eigenfunktionalität. Die Abgrenzung zwischen den anderen Modulen ist hingegen deutlicher. Das Report Modul, z.B. benutzt die vorhandenen Daten in der Projektverwaltung zur Auswertung. Eine Abhängigkeit von der Projektverwaltung zum Reportmodul besteht hingegen nicht.

Für jedes in Abbildung 5–1 dargestellte Modul existiert ein *Class Diagram* mit den einzelnen Klassen.

Abbildung 5-2

Class Diagram Projekte

Class Diagram Projekte



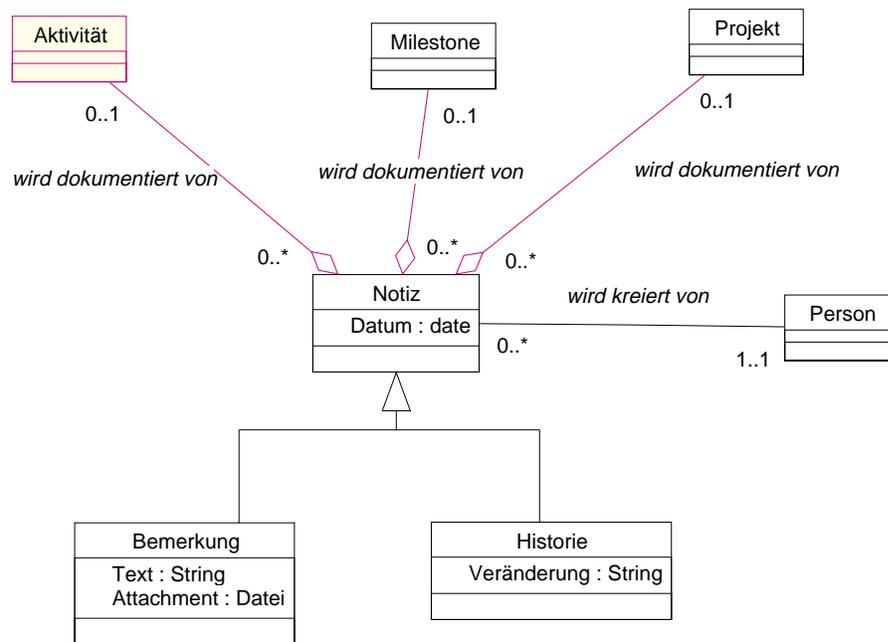
In Abbildung 5–2 ist das Zentralstück der Applikation, die Projektverwaltung, dargestellt. Im Mittelpunkt steht das *Projekt*. Ein *Projekt* enthält mehrere *Milestones* und mehrere *Aktivitäten*. *Milestones* wiederum sind aus mehreren *Aktivitäten* aufgebaut. *Milestones*, *Aktivitäten* und das *Projekt* werden alle durch *Notizen* dokumentiert. Weiterhin enthält ein *Milestone* auch *Verkäufe*, die durch ihre *Produktkategorien* spezifiziert werden.

Milestones bestehen aus einer Beschreibung, einer Phase (Milestonetyp), welcher der *Milestone* zugeordnet ist und einem Datum, an dem der *Milestone* erreicht sein soll. Die *Aktivitäten* enthalten außer Beschreibung, Anfangs- und Enddatum auch noch ein Erinnerungsdatum, an dem durch das Erinnerungsmodul eine Wiedervorlage stattfindet. *Aktivitäten* sind immer einer für sie verantwortlichen *Person* zugeordnet. Auch ein *Projekt* besitzt eine verantwortliche *Person* in Form des Accountmanagers. *Projekte* werden eindeutig identifiziert durch ihre Projektnummer. Außerdem muß ein Projektname und eine Beschreibung vergeben werden. Jedem *Projekt* ist eine *Entscheidungstabelle* zugeordnet. Sie enthält Fragen, durch deren Beantwortung die subjektive Wahrscheinlichkeit ausgerechnet werden kann.

In der obigen Abbildung ist auch das Modul *Verkäufe* enthalten. Ein *Verkauf* besteht aus einem Realisierungsdatum und einer Anzahl zu verkaufender (geplanter) bzw. schon verkaufter (getätigter) Produkte. Eine Kategorisierung der Produkte erfolgt über *Produktkategorien*, welche den Durchschnittspreis enthalten.

Abbildung 5-3 Class Diagram Notiz

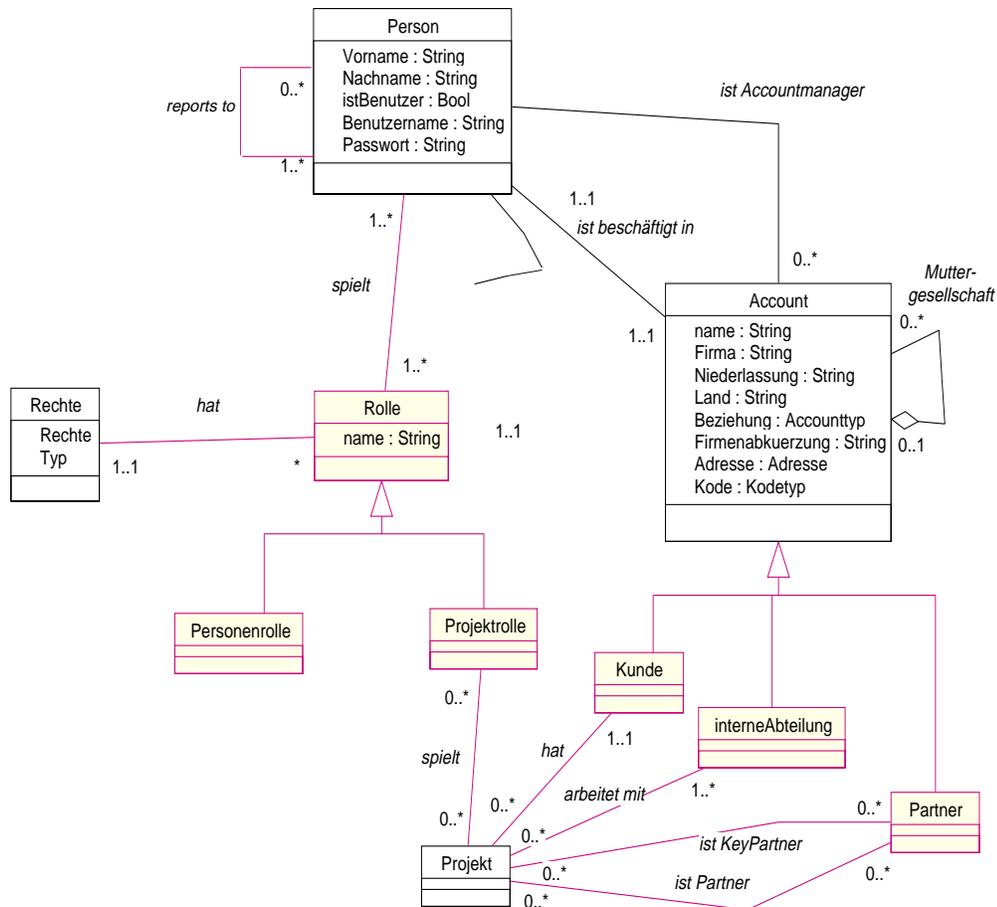
Class Diagram Notiz



Das Modul *Notiz* (Abbildung 5-3) ist sehr stark an die Projektverwaltung gekoppelt. Für jegliche Änderungen von *Aktivitäten*, *Milestones* und *Projekten* wird eine *Notiz* in Form einer automatisch generierten *Historie* erstellt. Sie wird den anderen Projektbeteiligten über das Erinnerungsmodul aufgeschlüsselt nach Projekten, Aktivitäten und Milestones präsentiert und dokumentiert in natürlicher Sprache die Änderungen. Eine andere Anwendung von *Notizen* sind *Bemerkungen*, die vom Benutzer als elektronischer Notizzettel an ein *Projekt*, eine *Aktivität* oder einen *Milestone* geheftet werden können. *Bemerkungen* werden im Gegensatz zur *Historie* nicht über das Erinnerungsmodul verteilt. Außerdem können *Bemerkungen* noch mit einer Datei verbunden werden.

Abbildung 5-4 Class Diagram Person

Class Diagram Person



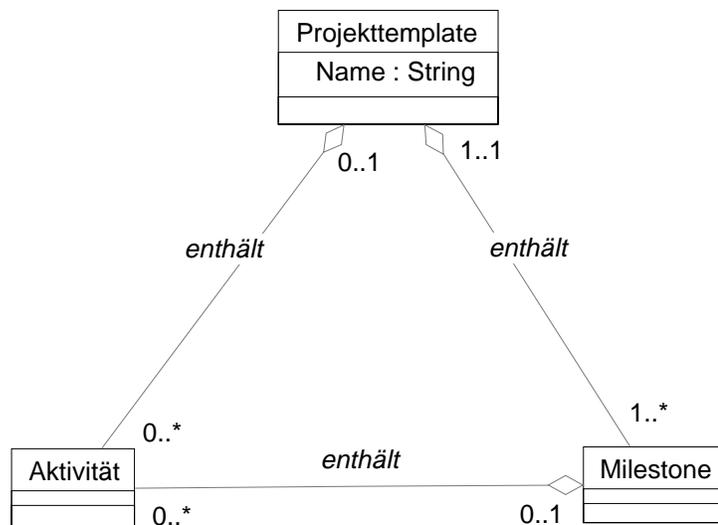
Das Modul Person/Account besteht aus den zwei großen Hauptklassen *Person* und *Account*. Die *Person* ist über die Linkklassen *Personenrolle* und *Projektrolle* an ein *Projekt* gebunden, wobei die *Accounts* je nach Subklasse (*Kunde*, *interneAbteilung*, *Partner*) direkt mit dem *Projekt* verbunden sind. Die *Personenrolle* definiert, welche *Rolle* eine *Person* in einem *Projekt* spielt (z.B. Projektleiter *Partner*, *Accountmanager*, *Promotor*). Außerdem werden über die *Personenrolle* die Berechtigungen der *Person* für das *Projekt* festgelegt. Eine *Person* kann nur eine *Personenrolle* in einem *Projekt* spielen. Die *Personenrolle*

len sind fest und können auch editiert werden. Ein Accountmanager z.B. hat Vollzugriff auf seine Projekte, während ein Promotor nur eingeschränkten (z.B. nur lesenden) Zugriff auf Projektdaten hat.

Eine Firmenhierarchie mit Mutter- und Tochtergesellschaften kann über die Relation *Muttergesellschaft* modelliert werden. Diese Beziehung ist in beide Richtungen vorhanden. Weiterhin können die interne Struktur und Beziehungen von Personen zueinander über die Relation *reports-to* abgebildet werden.

Abbildung 5-5 Class Diagram Projekttemplete

Class Diagram Projekttemplete

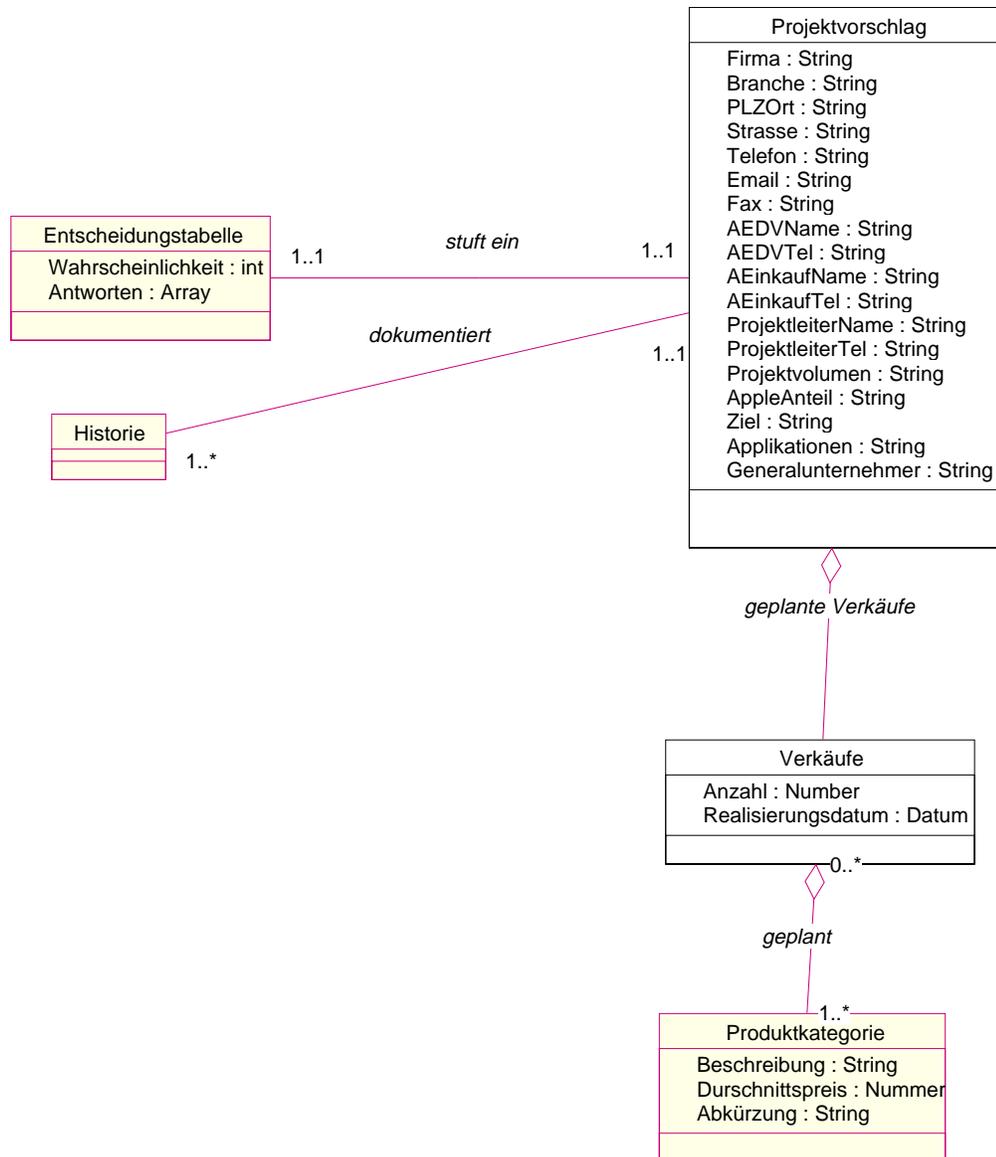


Das Modul Projekttemplete (Abbildung 5-5) ist sehr ähnlich zum Modul Projektverwaltung. Ein *Projekttemplete* besteht aus einem Name und aus zugeordneten *Milestones* und *Aktivitäten*. Bei der Neuerstellung eines *Projekts* werden alle im *Projekttemplete* vorhandenen *Milestones* und *Aktivitäten* ohne Datum in das neue Projekt übernommen und können dort gelöscht oder übernommen werden.

Abbildung 5–6

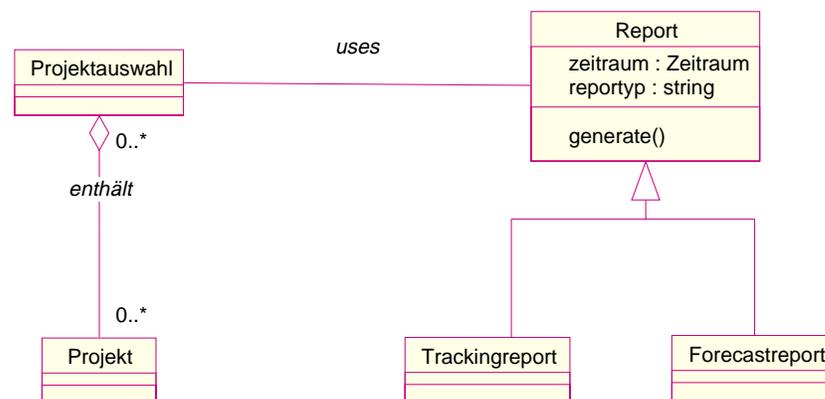
Class Diagram Projektvorschlag

Class Diagram Projektvorschlag



Ein *Projektvorschlag* geht jeder Neueingabe eines *Projekts* voraus. Hierbei müssen alle Daten des Vorschlags eingegeben werden. Die eingegangenen Vorschläge werden gesichtet und bewertet. Die Überführung in ein *Projekt* erfolgt manuell. Einige Daten können aus dem *Projektvorschlag* direkt in das *Projekt* übernommen werden (z.B. Beschreibungen, die *Entscheidungstabelle* und die *Verkäufe*). Zusätzlich zu den Rahmendaten enthält ein *Projektvorschlag* eine genaue Aufschlüsselung der geplanten Verkäufe. Die *Entscheidungstabelle* gibt Aufschluß über die objektive Wahrscheinlichkeit des *Projekts*. Alle diese Informationen werden benötigt, um den *Projektvorschlag* zu bewerten und ihn dann eventuell in ein *Projekt* zu überführen. Nicht angenommene Vorschläge werden nicht sofort gelöscht sondern bleiben als abgelehnt gespeichert. Sie können noch editiert werden und zur Wiedervorlage in die Liste der offenen *Projektvorschläge* übernommen werden. Dazu muß eine Begründung gegeben werden, welche in einer *Historie* dokumentiert wird.

Abbildung 5–7 Class Diagram Report



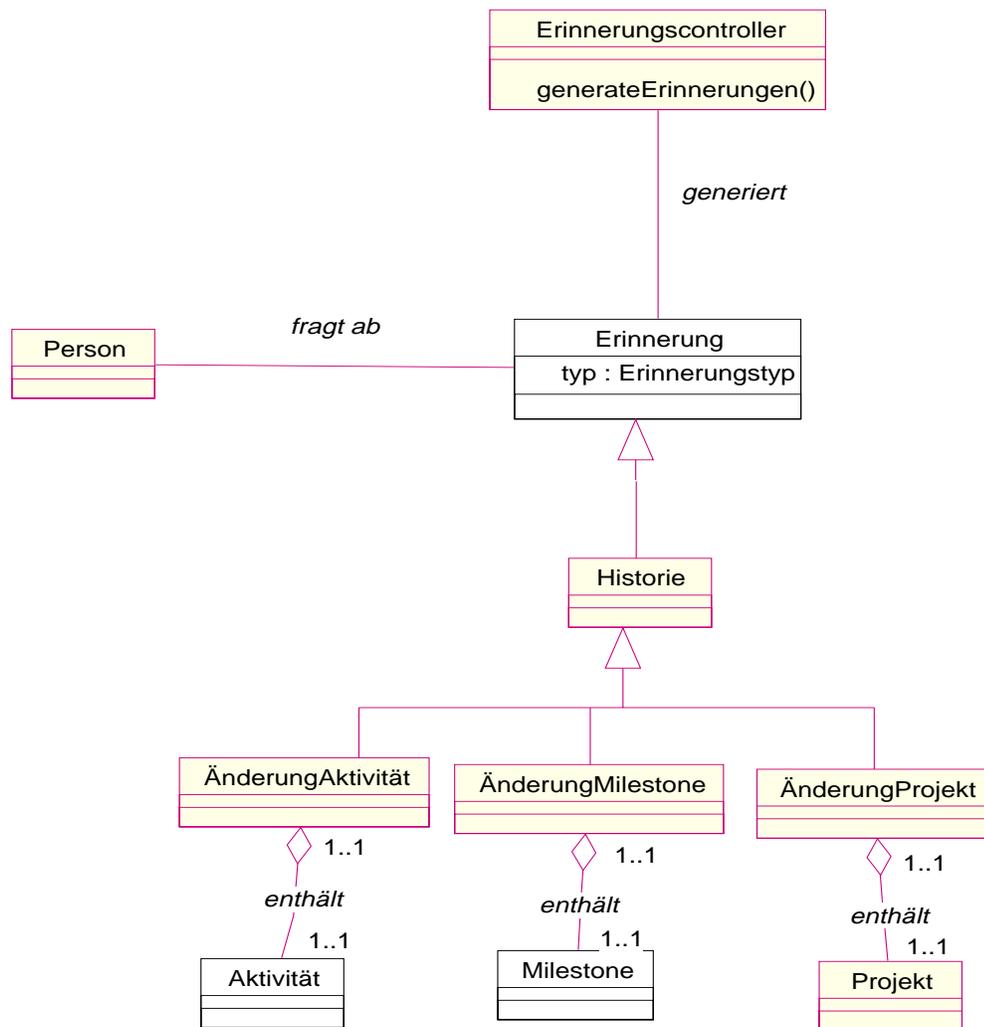
In Abbildung 5–7 ist das Reportmodul dargestellt. Für jeden speziellen Report ist eine Subklasse der allgemeinen Klasse *Report* zu generieren. Im Moment ist ein *Forecastreport* und ein *Trackingreport* vorgesehen. Ein *Report* bezieht sich immer auf eine *Projektauswahl*. Die Auswahl hält eine Menge von *Projekten*, die durch eine vorausgegangene Auswahl entstanden ist. Für die Auswahlkrite-

rien kommen alle, mit dem Projekt in Beziehung stehende Objekte in Betracht. So kann ein Menge von Projekten nach Accounts oder nach Accountmanagern selektiert werden.

Abbildung 5-8

Class Diagram Erinnerungen

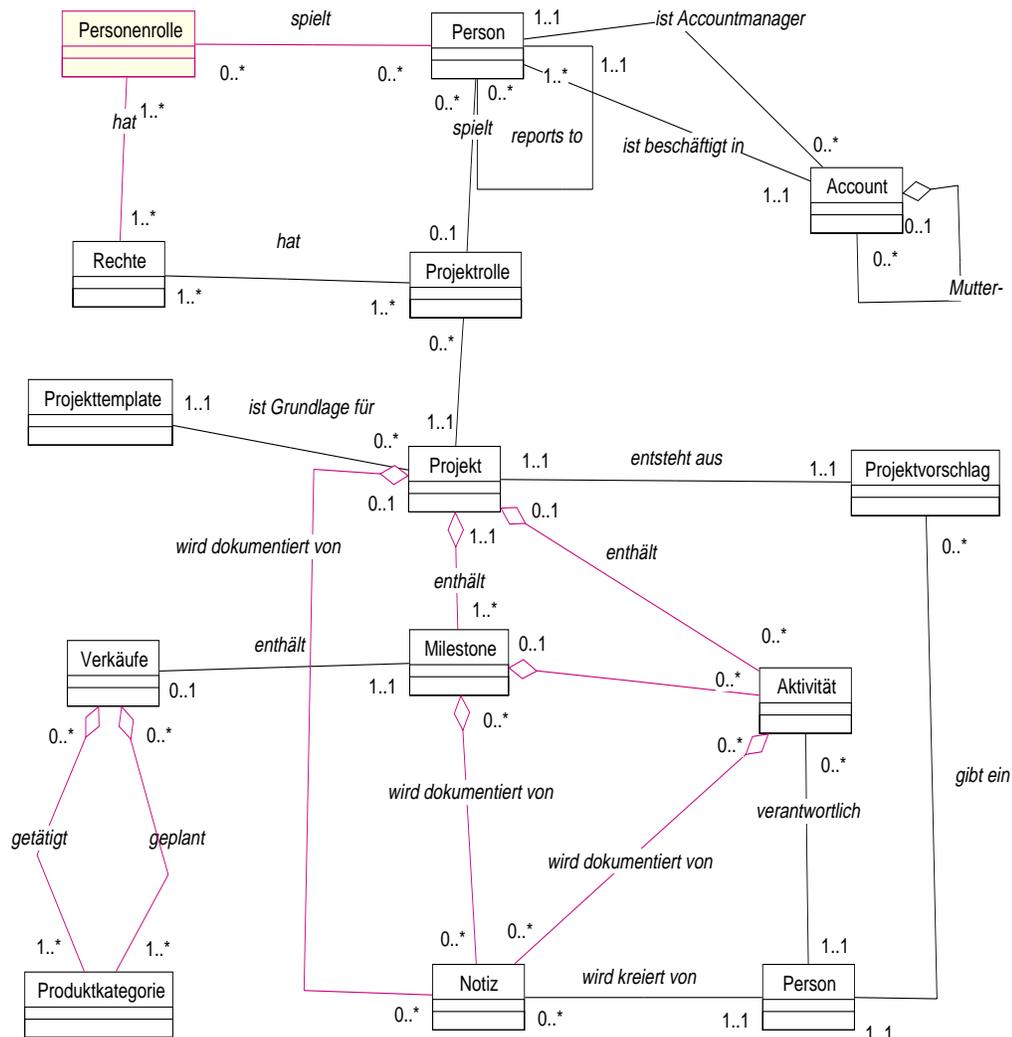
Class Diagram Erinnerungen



Die Erinnerungen stellen eine Art Filter dar, und werden automatisch vom System generiert. Sie zeigen dem Benutzer alle Änderungen und alle von ihm gesetzten Wiedervorlagetermine an. Erinnerungstermine können für Aktivitäten und Milestones gesetzt werden. Die Voraussetzung für den Erhalt von *Erinnerungen*, ist die Teilnahme an dem Projekt und die entsprechenden Berechtigungen für den Zugriff auf *Notizen*, *Aktivitäten* und *Milestones*.

Änderungen dokumentieren alle Veränderungen an den Projekten mit, die ein Projekt erfährt. Benutzer, die an dem Projekt beteiligt sind, bekommen automatisch die Änderungen in Form von Erinnerungen angezeigt. Zusätzlich gehören zu dem Teil Erinnerungen einer Person alle *Aktivitäten*, welche geplant sind und der bestimmten Person zugeordnet sind.

Abbildung 5–9 Class Diagram Zusammenfassung

Class Diagram
Zusammenfassung

Bemerkung: Die Klasse Erinnerungen mit ihren Subklassen ist in diesem Diagramm nicht aufgeführt, da sie einen eigenständigen Filter darstellt.

Abbildung 5–9 zeigt eine Zusammenfassung aller Klassen des Systems. Die Klassen des Moduls Erinnerungen sind nicht aufgenommen worden, da das Diagramm ansonsten sehr komplex geworden wäre. Es käme eine Verbindung

von jeder Erinnerungsklasse zu den Milestones, Aktivitäten, Projekten etc. dazu. (siehe Abbildung 5–8) Außerdem ist das Erinnerungsmodul ein Zusatz und nicht integraler Bestandteil der Projektverwaltung.

Der Mittelpunkt des Systems, das *Projekt*, entsteht aus einem *Projektvorschlag*, der von einer *Person* eingegeben wird. Als Grundlage dient ein *Projekttemplate*. Ein *Projekt* besteht aus *Aktivitäten*, *Milestones*, *Notizen* und *Verkäufen*. *Aktivitäten* sind einer bestimmten Person zugeordnet. *Notizen* haben immer einen Verweis auf die *Person*, die sie eingegeben hat.

Der andere Mittelpunkt des Systems ist die *Person*. Sie hat für jedes *Projekt* eine bestimmte *Personenrolle*, durch die seine *Berechtigungen* definiert sind. Für jede Subklasse von *Account* gibt es eine Relation zu einem *Projekt*. Es muß ein *Kunde* vorhanden sein. Zusätzlich können *Partner* (entweder als normale Partner oder als Keypartner) und *interne Abteilungen* am *Projekt* teilnehmen. Zusätzlich zu der spezifischen *Projektrolle* der *Person* für ein *Projekt* kann eine *Person* noch allgemeine *Personenrolle* innehaben, wie z.B. Administrator oder Vertriebsleiter, welche zusätzliche projektunabhängige *Berechtigungen* beinhaltet. Der *Kunde* hat keine herausragende Position in der Datenhaltung. Er hat die *Accountrolle* "Kunde". Um die internen Strukturen von *Accounts* und den darin beschäftigten *Personen* abzubilden, hat jede *Person* eine andere *Person* als Vorgesetzten. Wiederum hat jeder *Account* einen anderen *Account* als Muttergesellschaft.

5.2 Navigational Model

Das Navigational Model besteht aus zwei verschiedene Diagrammartentypen: das Navigational Schema und das Contextual Schema. Sie sind im Abschnitt OOHDM genauer beschrieben. Auch findet sich dort eine Beschreibung der Diagrammsyntax.

5.2.1 Navigational Schema

Das Navigational Schema stellt die möglichen Navigationspfade zwischen einzelnen Klassen des Conceptual Schemas dar. Sie werden mit Links untereinander verbunden. Es kann sein, daß das Conceptual Schema mit dem Navigational Schema identisch ist. Hier ist dies nicht der Fall. Die verwendete Syntax ist UML, da sie schon im Conceptual Model zum Einsatz kam. Die Abbildung 5–10 schaut unterschiedlich zu den vorigen Abbildungen aus, da es mit einem anderen Werkzeug erstellt wurde, welches auch beidseitige Beziehungen zwischen Klassen erlaubt.

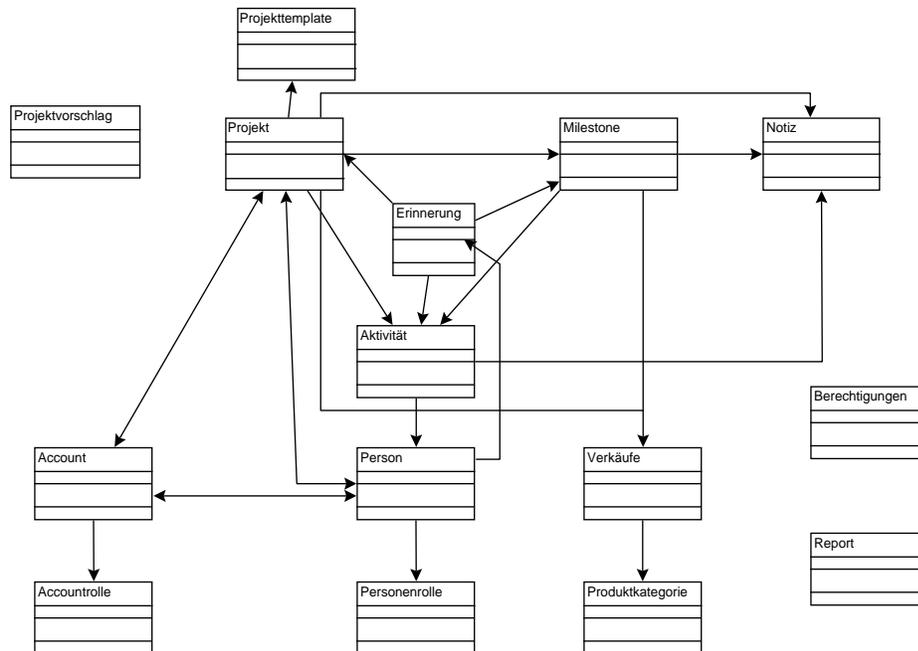
Das Navigational Schema (Abbildung 5–10) zeigt deutlich die möglichen Navigationspfade. So soll die Navigation von einem *Projekt* zu einem *Projekttemplate* möglich sein. Jedoch kann für ein gegebenes *Projekttemplate* nicht das *Projekt* ermittelt werden, da er in das *Projekt* übergegangen ist und nicht mehr existiert. So kann von einem gegebenen *Projekt* zu den jeweiligen *Milestones* navigiert werden, jedoch ist die Navigation zurück nicht möglich. Natürlich kann in der Projektverwaltung von einem *Milestone*, der zu dem *Projekt* gehört, wieder zurück zu dem selektierten *Projekt* navigiert werden. Die Fragestellung ist aber: Ist es für *irgendeinen* gegebenen *Milestone* möglich, zu dem zugehörigen *Projekt* zu navigieren? Dies ist für die Aufgabenstellung nicht nötig, deswegen ist der Pfeil nur in eine Richtung vorhanden. Solche Fragestellungen sind zu berücksichtigen, damit die Links auch ein eindeutige Semantik haben.

Die drei isolierten Klassen sind der *Projektvorschlag*, der *Report* und die *Berechtigungen*. Sie nehmen gar nicht an der Navigation teil. Zentral ist das Dreieck *Projekt*, *Aktivität* und *Milestone* inklusive den *Erinnerungen*. Das Modul Account und Personen kann untereinander durchnavigiert werden. Ein *Account* ist dabei immer einem *Projekt* direkt zugewiesen, wobei die *Person* noch Verbindungen zu den *Erinnerungen*, *Aktivitäten* und *Notizen* hat.

Abbildung 5–10

Navigational Schema

Navigational Schema

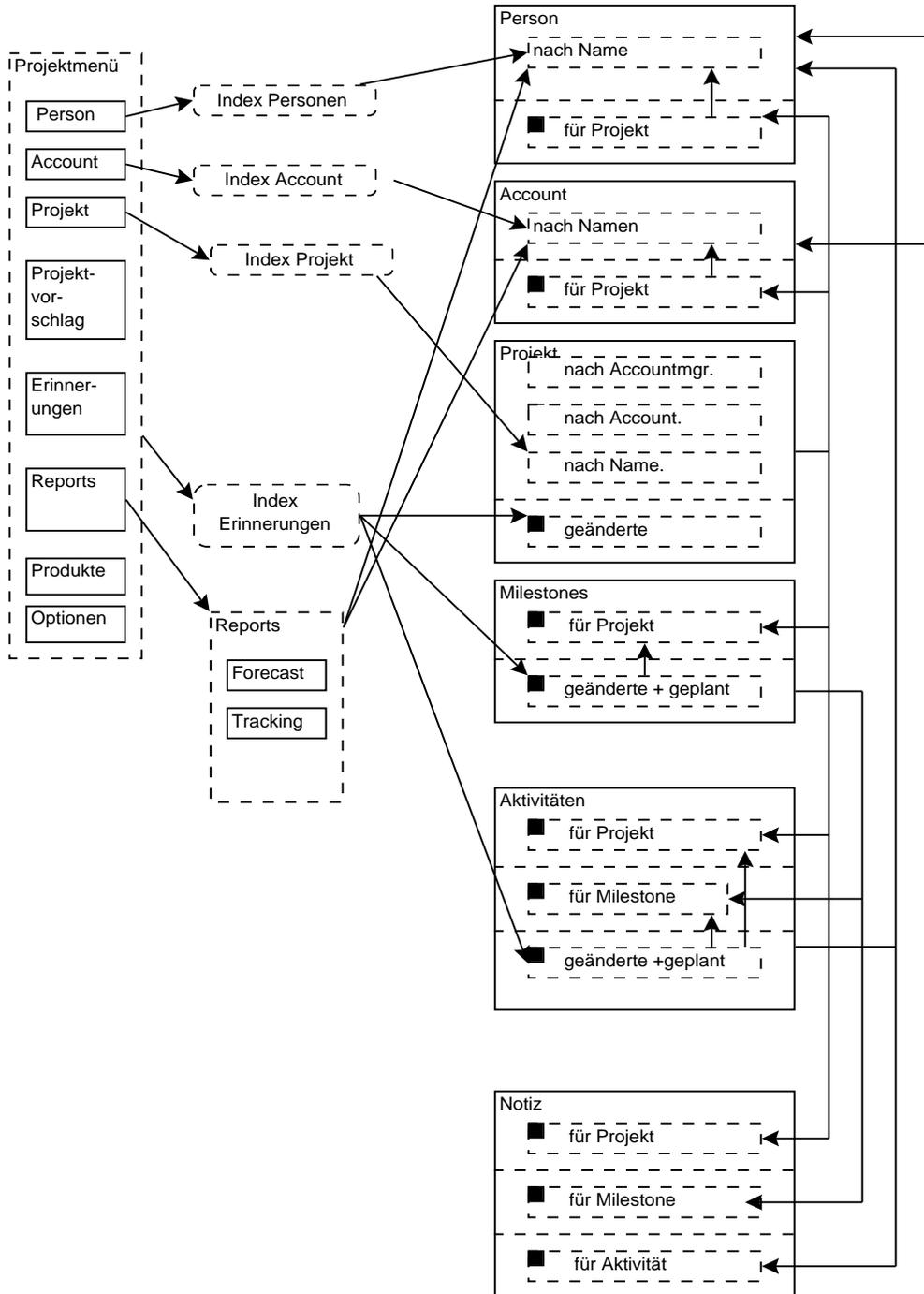


5.2.2 Contextual Schema

In Abbildung 5–11 ist das Contextual Schema für die Menüpunkte Projekt, Erinnerung, Personen und Accounts angegeben. Bei der Projektverwaltung geht dem Index eine Auswahl aller Projekte nach den Projektnamen, Accountnamen, der Projektnummer und dem Partnernamen voran. Es kann danach immer nur eine Projekt ausgewählt werden. Im Projekt kann wiederum zu den Milestones und Aktivitäten navigiert werden.

Die Erinnerungen werden alle angezeigt. Der Benutzer hat dabei die Möglichkeit entweder in dem Kontext der Erinnerungen zu bleiben und sich die Aktivitäten oder Milestonedetails anzuschauen oder den Kontext zu ändern und direkt in das dazugehörige Projekt zu wechseln, um dort die Änderungen durchzuführen.

Abbildung 5-11 Contextual Schema Projekt Erinnerung

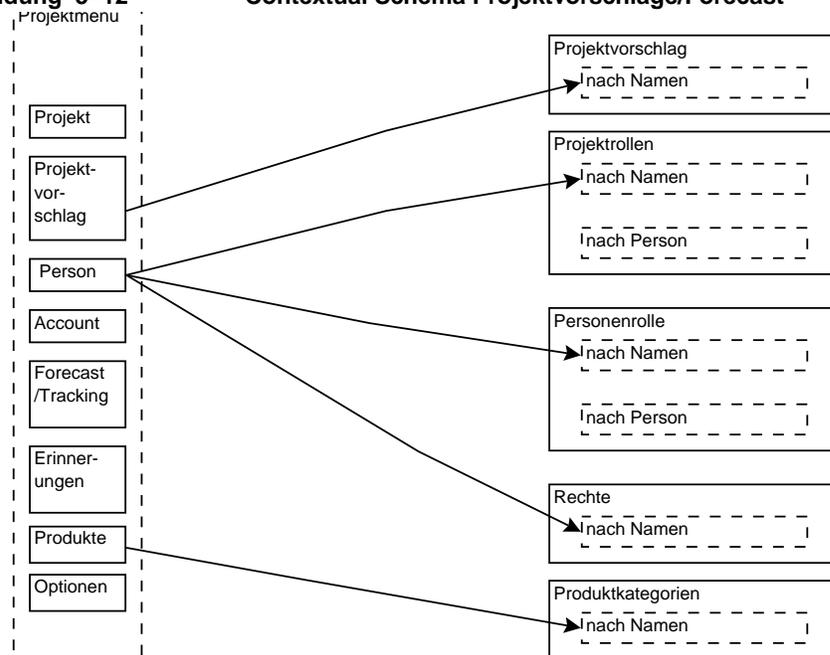


Auch für die Anzeige der Personen und Accounts geht dem Index eine Auswahl über gewisse Kriterien voraus. Von einer Person kann zum Account navigiert werden und auch die Navigation von einem Account zu den zugehörigen Personen ist unterstützt. Es ist auch sehr deutlich die enge Integration zwischen den Personen/Accounts und der Projektverwaltung zu erkennen. Für jedes Projekt, jede Aktivität und Notiz gibt es eine verantwortliche Person, die wiederum angewählt werden kann.

Für das Reportmodul können für jeden Reporttyp eine Anzahl von Projekten ausgewählt werden. Jedoch ist keine Navigation durch die einzelnen Projekte möglich.

In Abbildung 5–12 ist das Contextual Schema für den Bereich Rollenverwaltung, Projektvorschlag und den Produktkategorien angezeigt. Alle diese Module sind, wie auch schon im Navigational Schema angezeigt, von dem Rest der Navigationsklassen isoliert.

Abbildung 5–12 Contextual Schema Projektvorschläge/Forecast

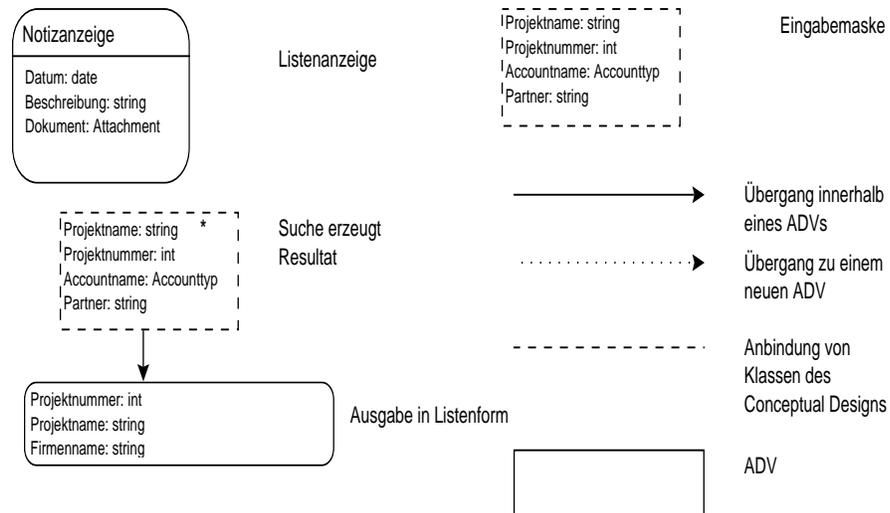


5.3 Abstract Interface Design

Im Abstract Interface Design wird losgelöst von der genauen implementations-spezifischen Darstellung das Aussehen und die Transformation der Applikation bei externen Events beschrieben. Außerdem wird die Verbindung zu den Navigationsobjekten hergestellt. Dies wird mit Hilfe der ADV (Abstract Data View) Diagrammen und ADV Charts verdeutlicht. Sie zeigen die generelle Struktur der Benutzeroberfläche und die Effekte, die Events vom Benutzer (wie ein Mouseclick) auf die Benutzeroberfläche haben. Eine genauere Beschreibung ist in Abschnitt 4.4 des Kapitels OOHDM enthalten.

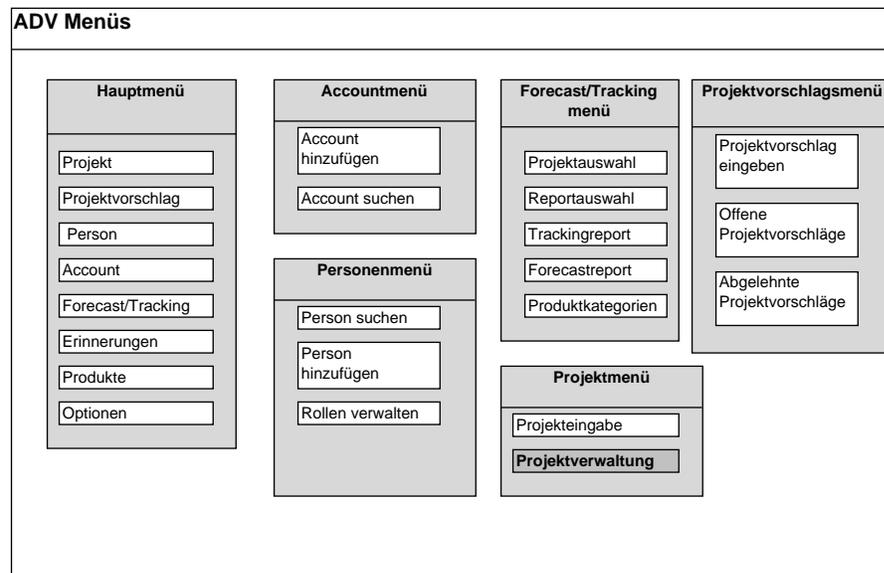
Zusätzlich zu der Notation die in OOHDM benutzt wird, sind die in Abbildung 5–13 gezeigten Elemente hinzugefügt worden. ADVs sind eine standardisierte Vorgehensweise, die, wie sich im Verlaufe des Designs gezeigt hat, nicht vollständig den Bedarf der hier zu entwickelnden Applikation abdecken. Da in OOHDM kein Unterschied zwischen Eingabe und reiner Darstellung (Ausgabe) gemacht wird, wurde ein spezielles Eingabeelement eingeführt. Auch die Ausgabe von einem oder mehreren Datensätzen wird in OOHDM nur implizit durch Beschriftung der Elemente verdeutlicht. Auch hier wurde eine Erweiterung der Syntax vorgenommen durch Einführung eines Listenelements. Interne Übergänge von ADV Elementen werden in ADV Diagrammen normalerweise nicht berücksichtigt. Wegen der Übersichtlichkeit wurden aber auch in den ADV Diagrammen Übergänge eingeführt.

Abbildung 5–13 Angepaßte Syntax von ADV Diagrammen



5.3.1 Menüs

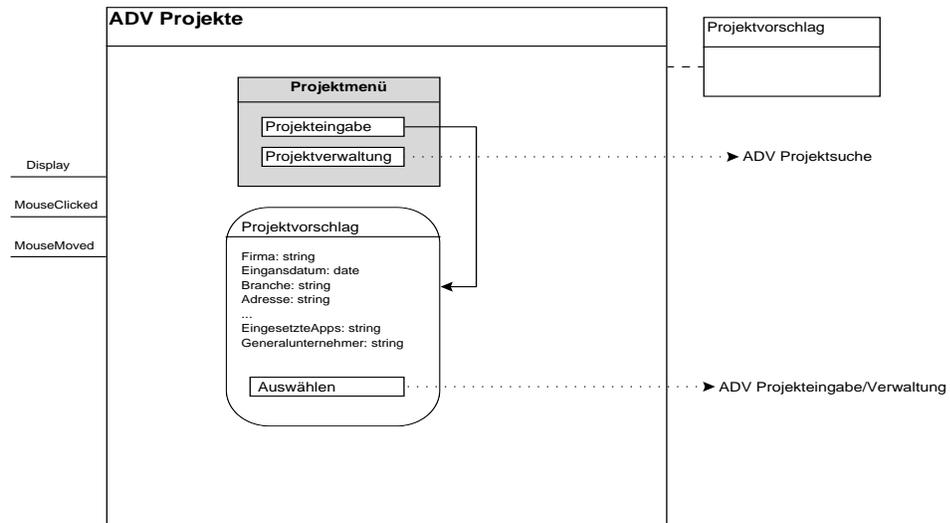
Abbildung 5–14 ADV Diagramm Menüs



In Abbildung 5–14 sind alle Menüpunkte aufgeführt. In nachfolgenden ADVs wird das Hauptmenü aus Gründen der Übersichtlichkeit und der Redundanzvermeidung immer weggelassen.

5.3.2 Projektverwaltung

Abbildung 5–15 ADV Diagramm Projektverwaltung Übersicht



Die Verwaltung von Projekten besteht aus den zwei Menüpunkten Projektverwaltung (für die Verwaltung existierender Projekte) und der Projekteingabe. Der Projektverwaltung selber ist die Projektsuche vorgeschaltet. Dieser Zentralpunkt ist in Abbildung 5–16 und Abbildung 5–17 dargestellt, die aus Platzgründen auf zwei ADV Diagrammen untergebracht sind.

Abbildung 5–16 ADV Diagramm Projektverwaltung Projekteingabe/verwaltung

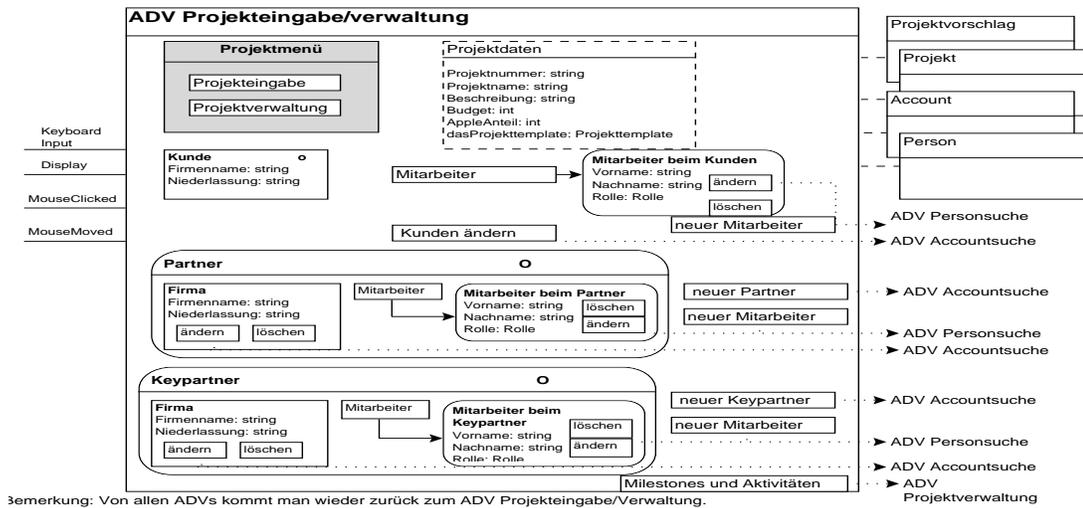
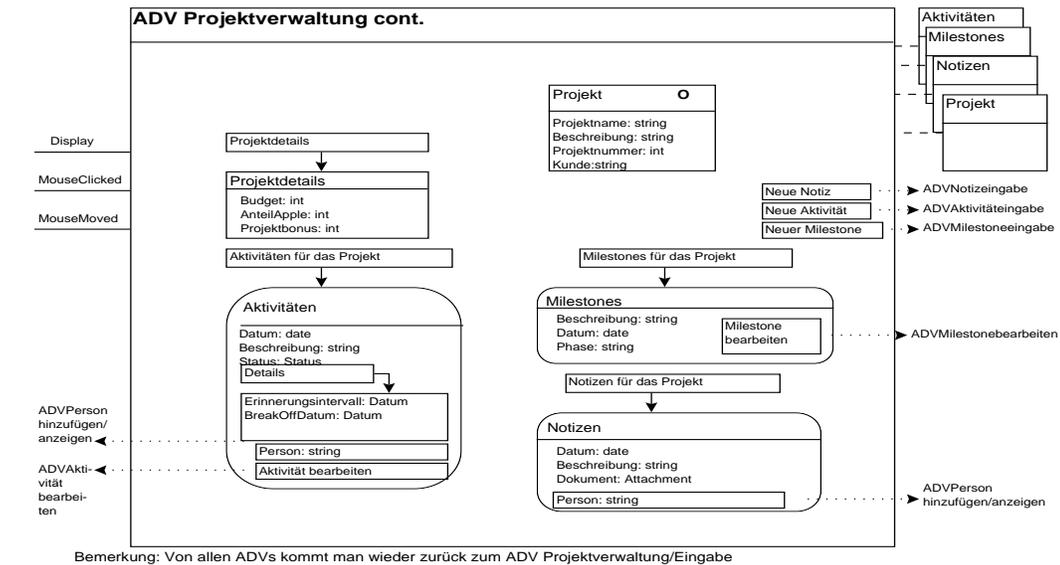


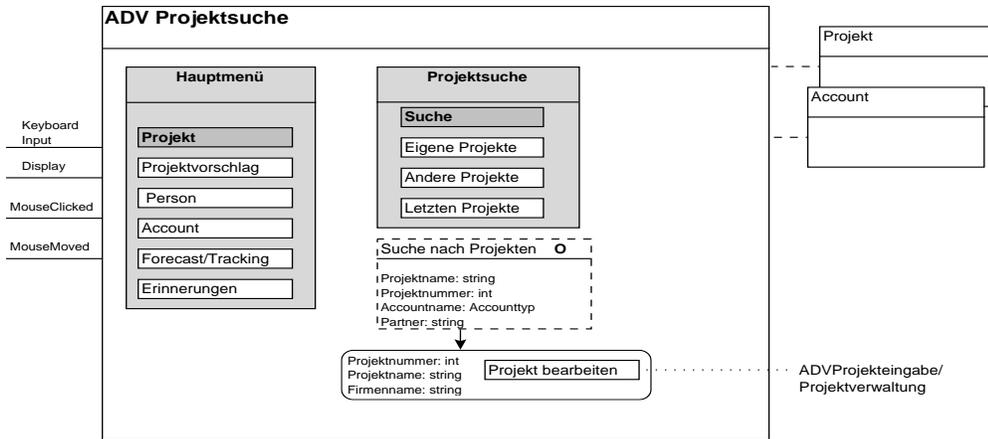
Abbildung 5–17 ADV Diagramm Projektverwaltung continued



Auf den ersten Blick sind die beiden Diagramme sehr unübersichtlich und überladen. Da aber ein Großteil der Informationen nur auf Benutzerwunsch präsentiert wird, reduziert sich die Komplexität erheblich. Dieses Konzept findet sich immer wieder in den Diagrammen. Z.B. in Abbildung 5–17 werden die Aktivitäten für ein Projekt nur

nach Aktivierung des Links “Aktivitäten für das Projekt” gezeigt. In dieser Anzeige wiederum versteckt sich ein weitere Stufe der Detailansicht. Das Erinnerungsintervall und das Breakoff Datum werden nur auf weiteren Wunsch angezeigt.

Abbildung 5-18 ADV Diagram Projektsuche



Die Projektsuche in Abbildung 5-18 ist der Projektverwaltung vorgeschaltet.

Abbildung 5-19 ADV Projektverwaltung Milestone

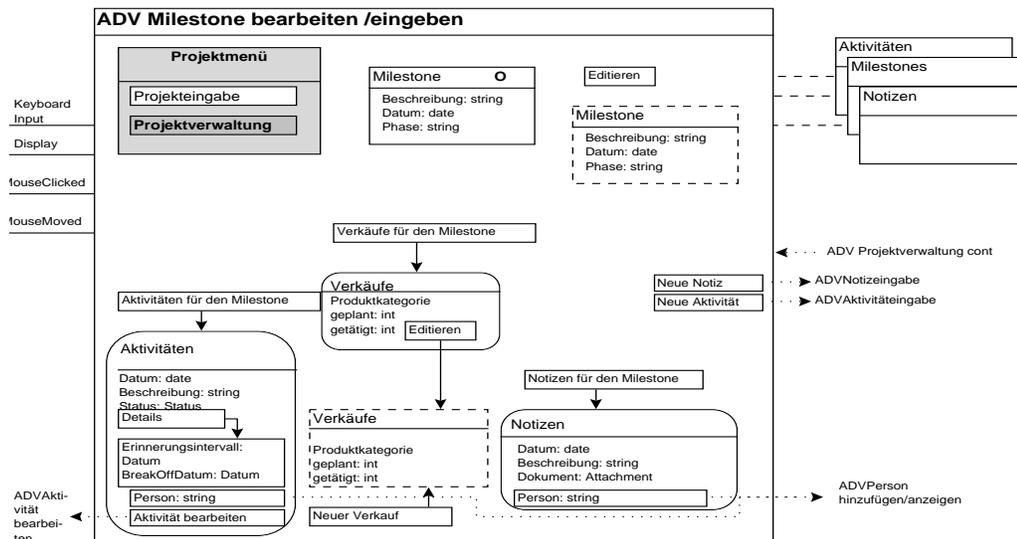


Abbildung 5–20 ADV Diagram Projektverwaltung Aktivitäten

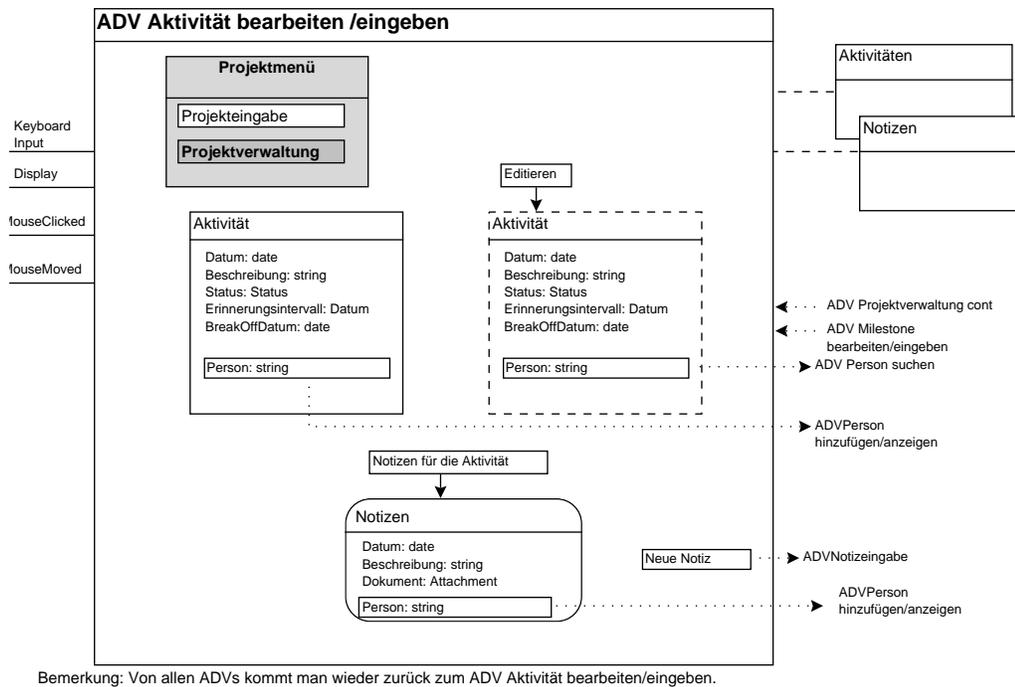
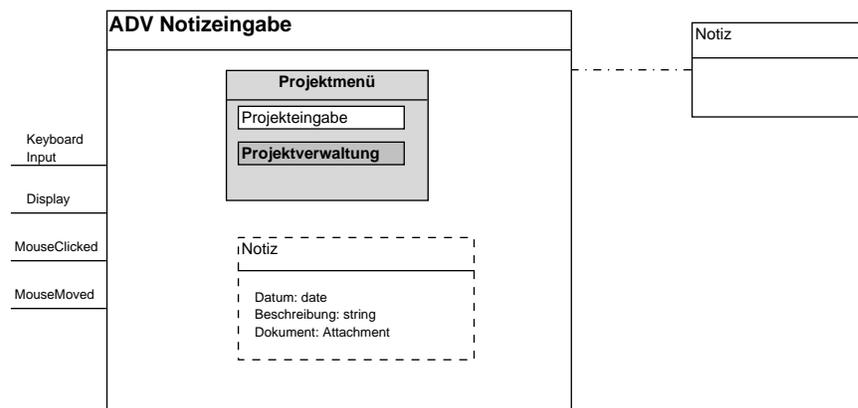


Abbildung 5–21 ADV Diagram Projektverwaltung Notizen



In Abbildung 5–19, Abbildung 5–20 und Abbildung 5–21 sind die Unteransichten der Projektverwaltung für die Bearbeitung von Milestones, Aktivitäten und Notizen dargestellt. Sie sind alle erreichbar über die Projektverwaltung.

5.3.3 Accounts und Personen

Der Unterbereich Accounts ist sehr eng mit dem Bereich Personen verbunden. Sie ermöglichen die Eingabe, Suche und Veränderung von Accounts und Personen. Es gibt jeweils ein ADV Diagramm für die Eingabe/Verwaltung und eins für die Suche. In den Bereich Personen fällt auch noch die Rolleneingabe und Verwaltung.

Abbildung 5-22 ADV Diagram Accounts suchen

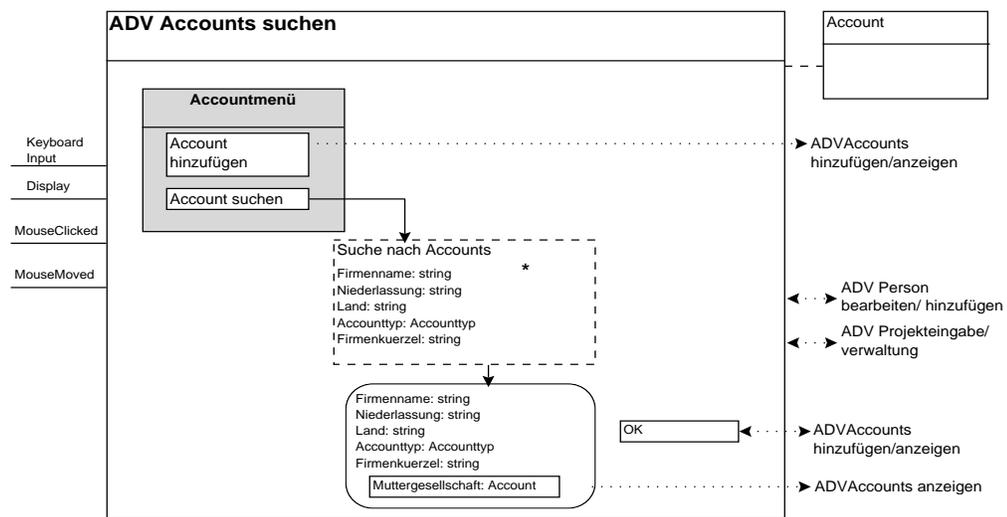


Abbildung 5-23 ADV Diagram Accounts eingeben/verwalten

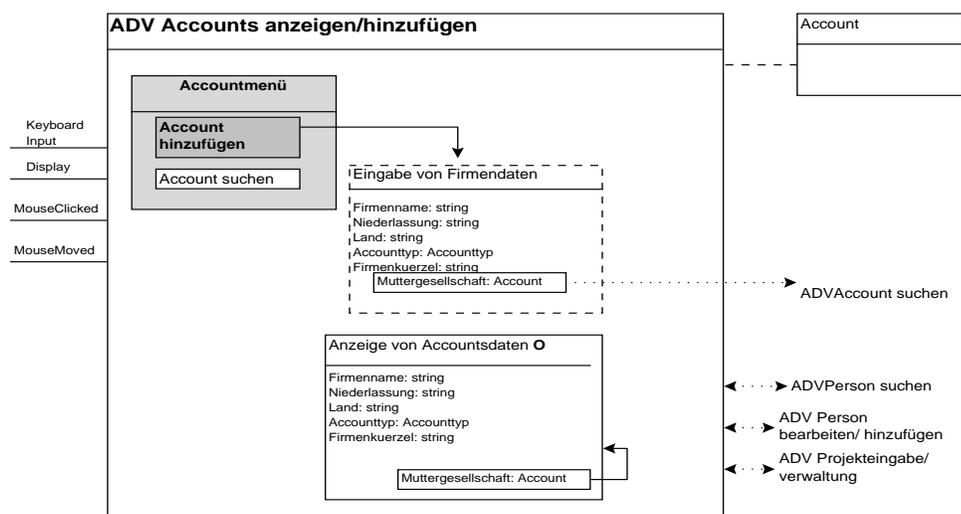


Abbildung 5-24 ADV Diagram Person suchen

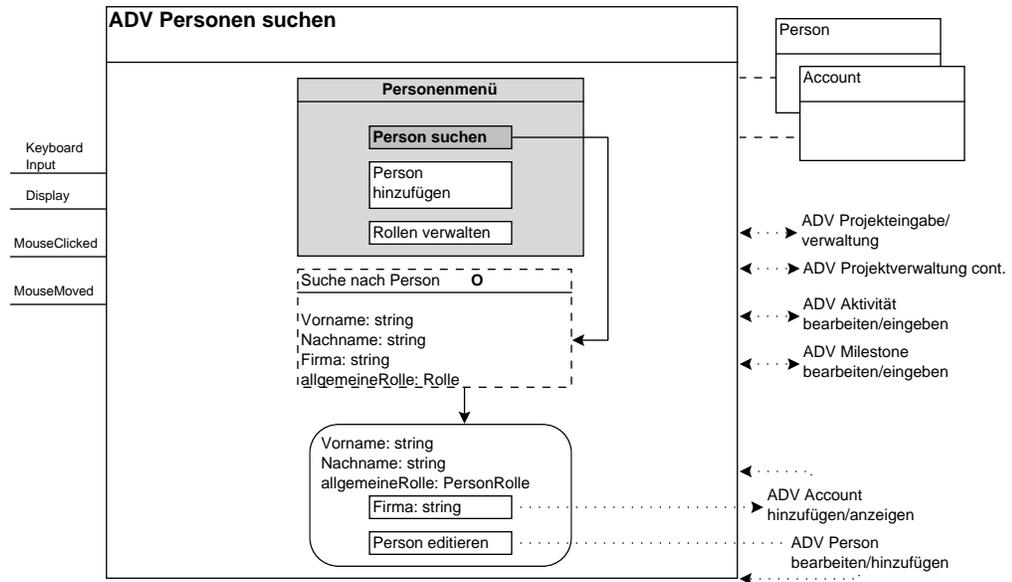


Abbildung 5-25 ADV Diagram Person eingeben/verwalten

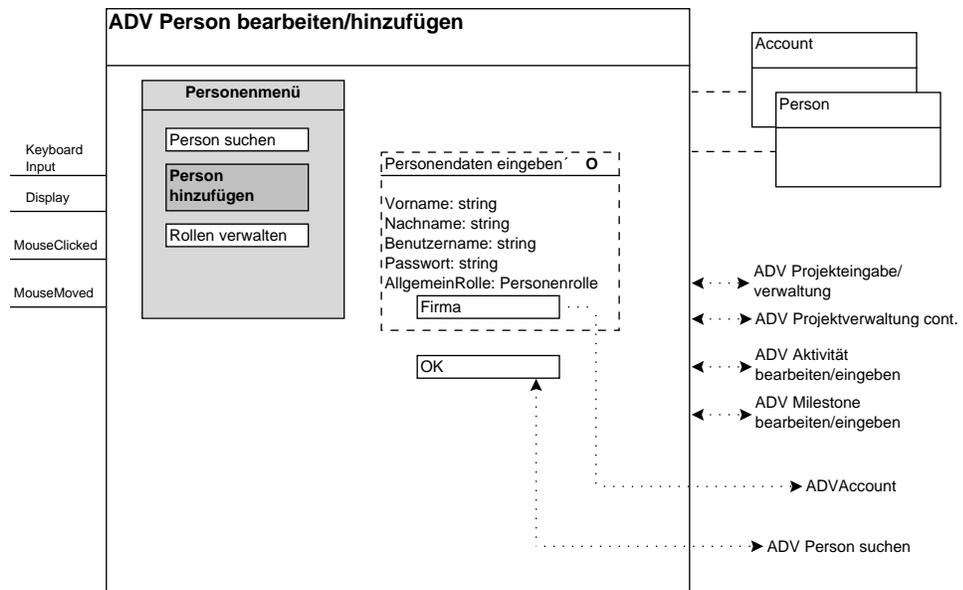
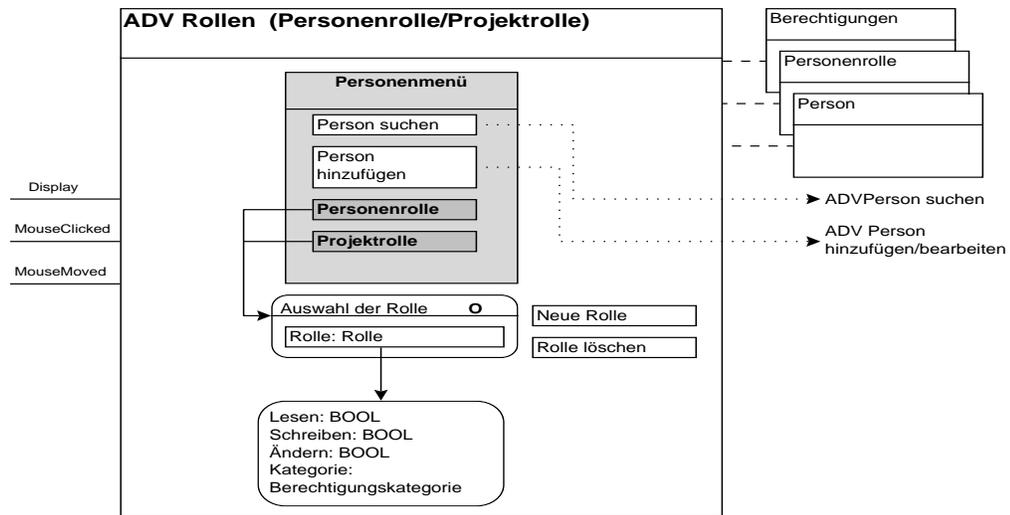


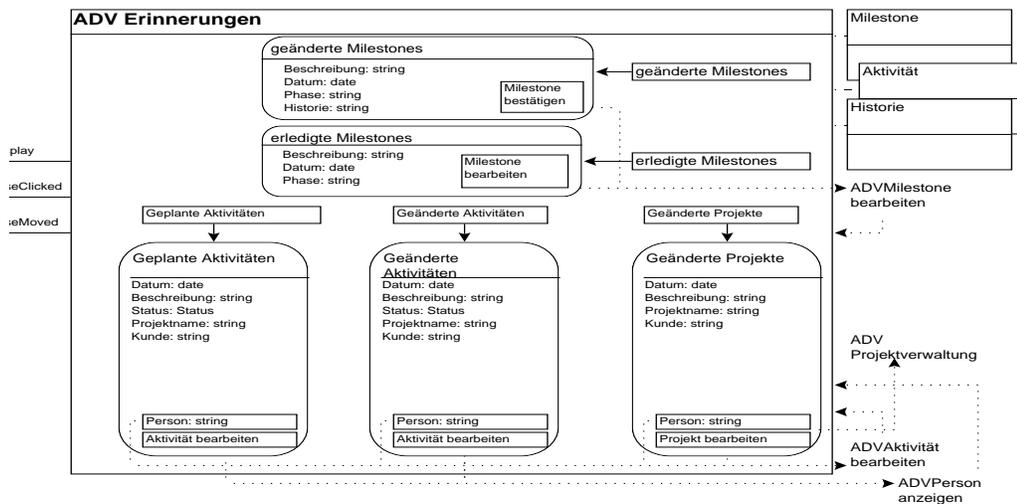
Abbildung 5-26 ADV Diagram Rollen eingeben/verwalten



5.3.4 Erinnerungen

Diese Oberfläche wird dem Benutzer direkt nach dem Einloggen präsentiert. Übersichtlich werden alle Erinnerungen an geänderte oder erledigte Aktivitäten und Milestones präsentiert. Auch hier wird das Konstrukt der Detailansicht, das auch in der Projektverwaltung verwendet wird, eingesetzt.

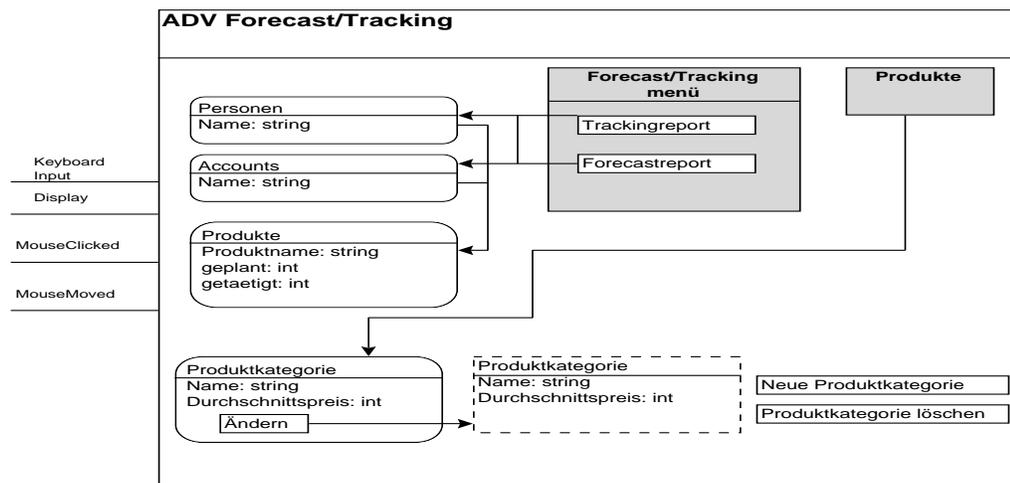
Abbildung 5-27 ADV Diagram Erinnerungen



5.3.5 Reporting

Für das Reporting werden die Projekte nach Accountmanager oder nach Abteilung ausgewählt. Die Projekte der ausgewählten Abteilung oder der ausgewählten Accountmanager werden für das Reporting herangezogen. Es steht ein Tracking- und ein Forecastreport zur Verfügung. Es werden die Projekte und die Verkäufe in den einzelnen Produktkategorien aufgelistet. Je nach dem ob ein Forecast- oder Trackingreport gewählt wurde, werden die geplanten oder die getätigten Stückzahlen angezeigt. Der Report kann auf einen Zeitraum beschränkt werden.

Abbildung 5–28 ADV Diagram Reporting

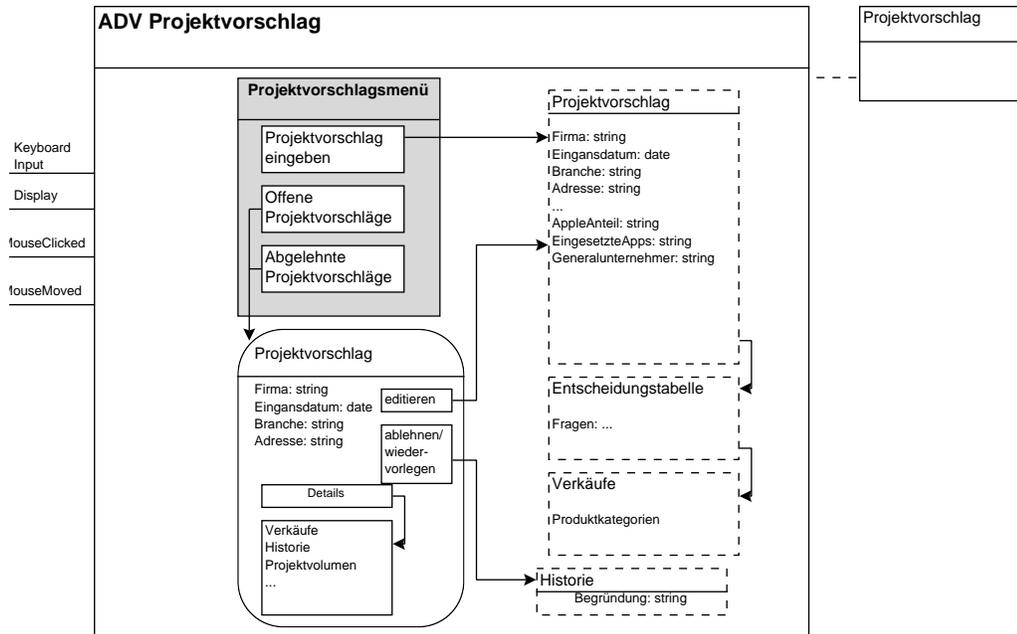


5.3.6 Projektvorschlag

Jeder Projekteingabe geht ein Projektvorschlag voraus. Für die Eingabe muß ein Formular mit allen geforderten Daten ausgefüllt werden. Anschließend wird die Eingabemaske für die Berechnung der Wahrscheinlichkeiten mit mehreren Fragen angezeigt. Zum Schluß müssen die Verkäufe, nach Produktkategorien aufgeschlüsselt, eingegeben werden. Die eingegebenen Projektvorschläge haben zuerst den Status *open*. Sie können unter dem Punkt "offenen Projektvorschläge" eingesehen werden. Dort kann aus einem offenen Projektvorschlag ein abgelehnter Vorschlag durch Eingabe der Ablehnungsbegründung gemacht

werden. Dies hat zur Folge, daß der Projektvorschlag nun unter der Rubrik “Abgelehnte Projektvorschläge” zu finden ist. Bei beiden Projektvorschlagsanzeigen können die Vorschläge editiert werden, wobei der gleiche Prozeß wie bei der Eingabe zu durchlaufen ist.

Abbildung 5–29 Projektvorschlag eingeben/verwalten



6. Implementierung

Die Implementierung stützt sich auf die Designphase. Dieses Kapitel beschreibt die eingesetzte Implementierungstechnologie, sowie die Umsetzung des Conceptual Designs, Navigational Designs und des Abstract Interface Designs. Weiterhin werden einige Implementierungskonzepte erläutert.

6.1 Einleitung

Die Implementierung eines Prototyps folgte dem ersten Conceptual und Navigational Design. Er besaß keine Funktionalität aber stellte die erste Oberfläche sowie Navigationsmöglichkeiten dar und wurde parallel zu dem Abstract Interface Design erstellt. Dies gab dem Kunden die Möglichkeit, sich die Applikation vorzustellen und seine Arbeitsabläufe daran zu testen. Auch beim späteren Umsetzen diente diese abgenommene Vorlage als Grundstein für die Applikation.

Die Vorgehensweise während der Implementierung war nach Modulen geordnet:

- Projektvorschlag
- Projekte
- Personen

- Accounts
- Berechtigungen
- Erinnerungen
- Reporting

In dieser Reihenfolge wurden die Module implementiert. Von der Implementierungsdauer ist das Projektmodul das zeitaufwendigste, dicht gefolgt von dem Personen und Accountmodul. Diese zwei Module sind eng miteinander verknüpft, wohingegen die anderen Module mehr oder weniger eigenständig sind. Im Design wurde nach der Methode OOADM vorgegangen. Drei Designphasen wurden durchlaufen: Conceptual Design, Navigational Design und Abstract Interface Design. Die Umsetzung der einzelnen Phasen ist nachfolgend beschrieben. Als Implementierungssprache wurde Englisch gewählt. Deswegen werden in nachfolgenden Diagrammen die englischen Übersetzungen der in der Analyse und im Design verwendeten Begriffe benutzt.

6.2 Implementierung des Conceptual Designs

Im Conceptual Design wurden die einzelnen Klassen und ihre Relationen zueinander definiert. In der Implementierung diente das Conceptual Design als Grundlage für das Datenbankdesign, welches hier beschrieben werden soll.

Die einzelnen Klassen des Conceptual Schemas wurden in Tabellen der Datenbank umgewandelt. Die Verknüpfungen zwischen den Klassen wurde in 1-n, 1-1, bzw. n-m Relationen einer relationalen Datenbank überführt. Dies impliziert die Einführung von Verknüpfungstabellen. Außerdem mußten die Attribute um Primärschlüssel und Fremdschlüssel erweitert werden. Das Enterprise Object Framework, die Middleware zwischen Applikation und Datenbank, unterstützt auch Vererbung. Es existieren mehrere Möglichkeiten, um Vererbung auf einer relationalen Datenbank abzubilden. Aus Leistungsgründen wurde die Eintabellenvererbung genommen. Hierbei werden alle Attribute aller Klassen einer Vererbungshierarchie in einer Tabelle gespeichert. Für eine gegebene Klasse werden die nichtgebrauchten Attribute mit Nullwerten gefüllt. Da die einzelnen

Klassen der Vererbungshierarchie nicht stark in den Attributen untereinander abweichen, kommt es trotzdem zu keinem großen Verlust an Datenbankplatz. Bei dieser Vorgehensweise ist für die Restaurierung eines jeglichen Objekts nur ein Zugriff auf die Datenbank notwendig.

Im Folgenden ist das Datenbankschema als Entity-Relationship Modell dargestellt. Die Diagramme stammen von dem Entwicklungswerkzeug Enterprise Object Modeller, der im Kapitel 6.5.3 kurz beschrieben ist.

Exemplarisch wird die verwendete Notation an Abbildung 6–1 erläutert. Eine Klasse besteht aus zwei Teilen: den Attributen und den Relationen. To-One Relationen sind mit einem Einfachpfeil markiert wohingegen To-Many Relationen mit einem Doppelpfeil markiert sind. In der Klasse sind alle Relationen aufgeführt. Für die Relationen zu Klassen die in dem aktuellen Diagramm sichtbar sind, werden die Relationen auch graphisch durch Verbindungslinien dargestellt. Leider bietet der Diagrammodus des Enterprise Object Modellers keine Möglichkeit für die Darstellung der Vererbungshierarchien. Deshalb wird auf die Vererbung nur im Beschreibungstext eingegangen.

Alle Primärschlüsselattribute sind mit einem Schlüssel gekennzeichnet. Nicht alle aufgeführten Attribute sind auch Teil der Klasse. Nur die mit einer Raute markierten Attribute sind Klassenattribute. So werden z.B. Fremdschlüssel und Primärschlüssel nicht in die Klasse übernommen.

Abbildung 6–1 zeigt das Datenbankschema der Projektverwaltung. Es entspricht bis auf die hinzugekommene Verbindungsklasse *SalesPerCategory* dem Conceptual Design. Das Modul Notizen ist mit in dem Diagramm enthalten. Die Vererbungen, welche im Conceptual Design modelliert wurden, können hier nicht gezeigt werden. So erbt *Comment* von *Note*.

Abbildung 6-2 Datenbankschema für Accounts/Projekt

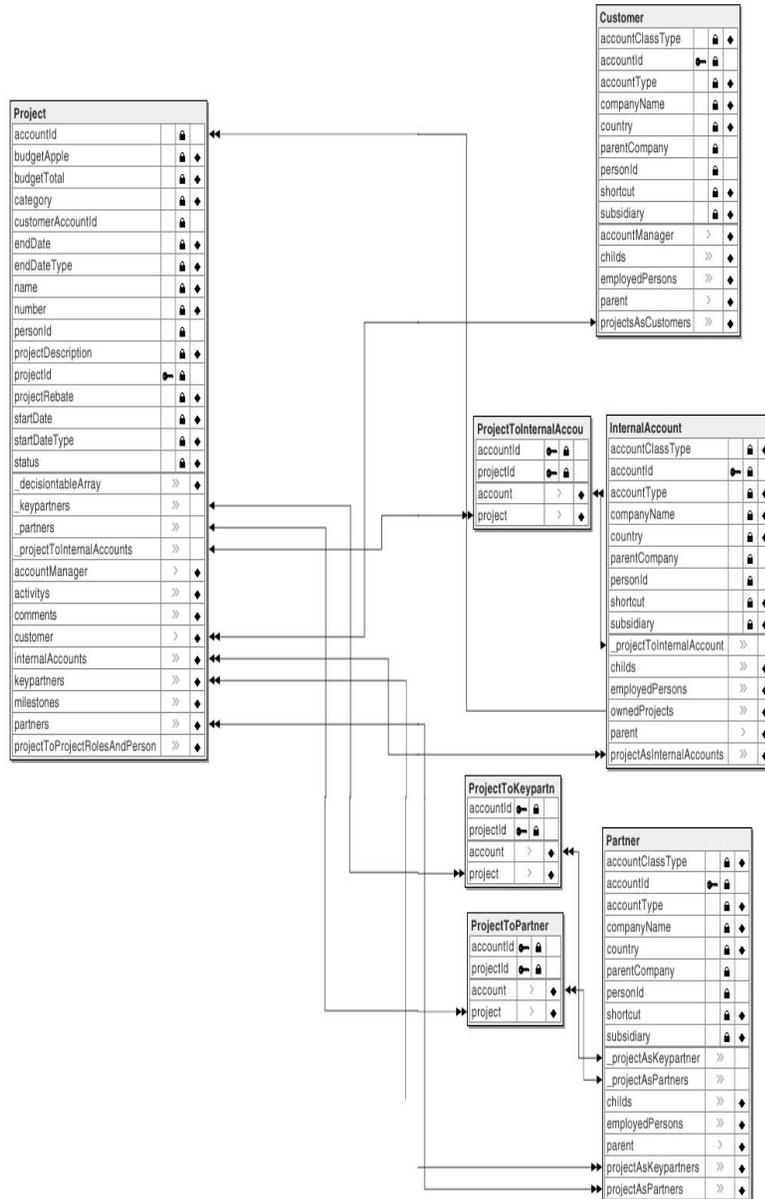


Abbildung 6–2 zeigt die Integration von Accounts und Projekten. Die drei Klassen *Kunde*, *interne Abteilung* und *Partner* sind alles Subklassen von *Account* (hier nicht gezeigt). Sie werden verbunden über jeweils eine Verbindungsklasse (*ProjektToInternalAccount*, *ProjektToKeypartner*, *ProjektToPartner*), wobei ein Partner in zwei verschiedenen Bezügen zum Projekt stehen kann, nämlich als Keypartner oder als Partner.

In Abbildung 6–3 ist das Datenbankschema für das Conceptual Schema der Personen angegeben. Auch hier sind wiederum Verbindungsklassen eingefügt worden (*ProjektToProjectRoleAndPerson*, *PersonToPersonRole*, *PrivilegeToCategoryAndRole*). Die Verbindungen, die als n-m Relationen mit zwei Doppelpfeilen markiert sind, werden eigentlich über die angezeigten Verbindungen über die Linkklassen realisiert. Z.B. hat die Klasse *Persons* keine direkte Verbindung zu der Klasse *PersonRole*. Diese wird über die Linkklasse *PersonToPersonRole* realisiert. Trotzdem ist in dem Diagramm die Verbindung eingezeichnet. In diesem Datenbankschema befinden sich auch die zwei tertiären Relationen zwischen *Persons*, *Projekt* und *Projektrolle*, sowie zwischen *Projektrolle*, *PrivilegienCategory* und *Privilegien*, die wiederum über eine Linkklasse realisiert sind. Die Klassen *Rolle* (nicht eingezeichnet), *Projektrolle* und *Personenrolle* bilden eine Vererbungshierarchie mit der Klasse *Rolle* als Wurzel.

Abbildung 6-3 Datenbankschema Personen und Berechtigungen

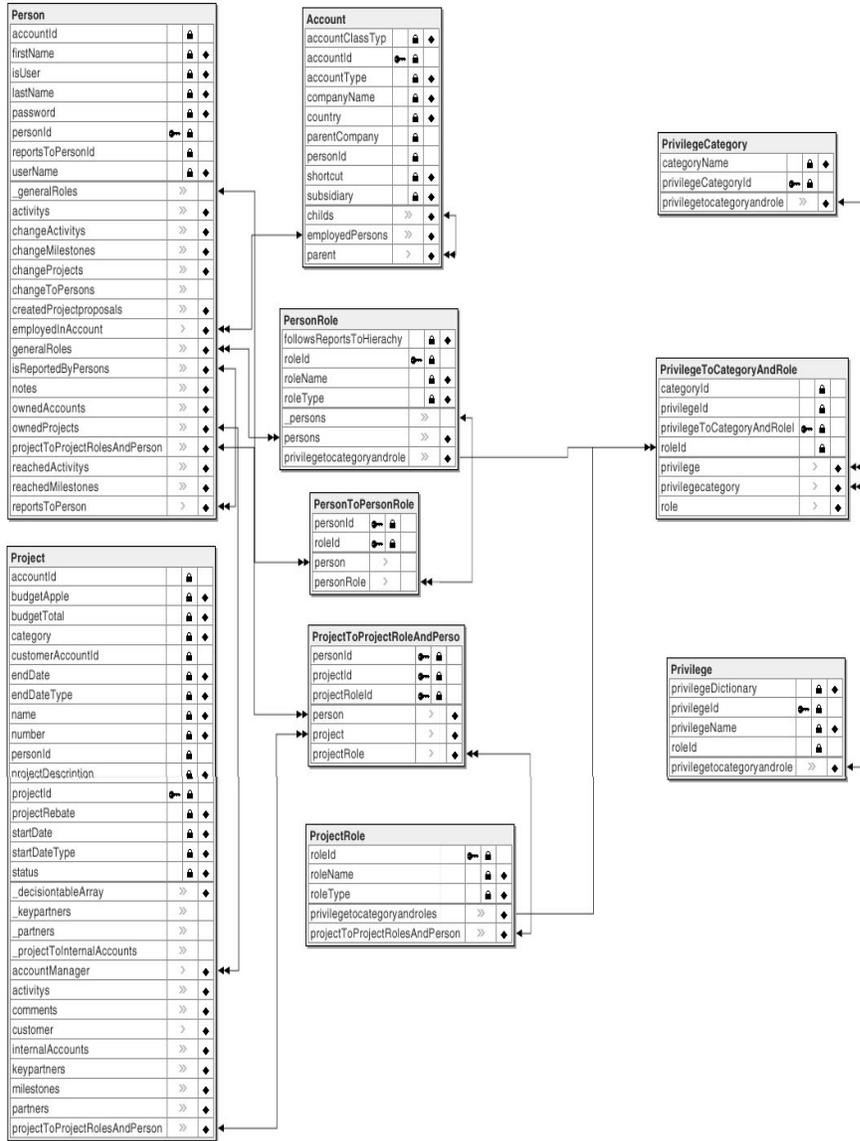
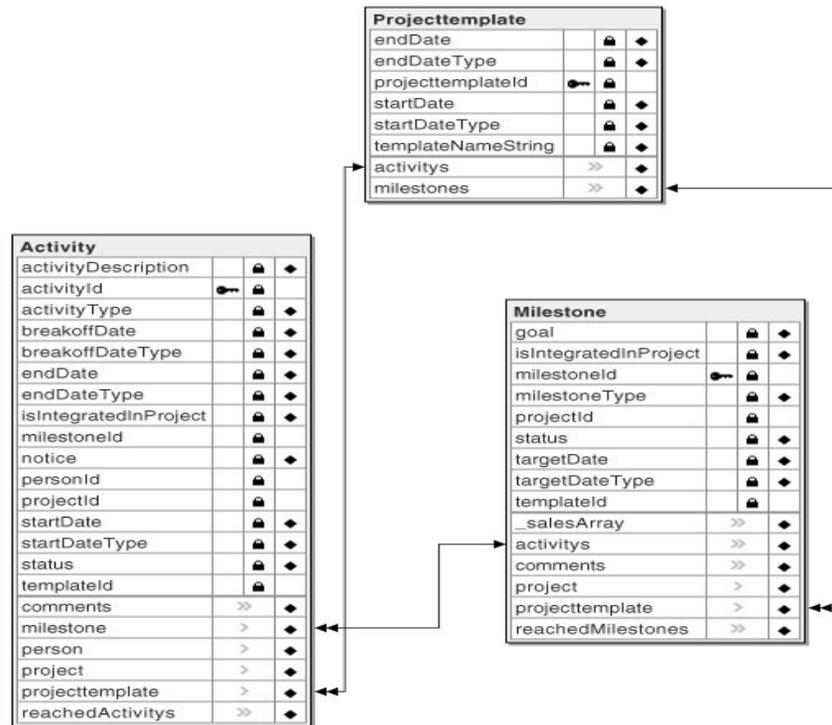


Abbildung 6-4 Datenbankdesign Projekttemplate



In Abbildung 6-4 ist das Datenbankdesign für die Projekttemplates gezeigt. Im Conceptual Schema ist eine Datumsangabe als ein Attribut modelliert. Jedoch existieren drei verschiedene Repräsentationen für ein Datum, je nach dem ob es sich um ein exaktes Datum, eine Monatsangabe bzw. eine Angabe der Kalenderwoche handelt. Dazu mußte für jedes Datum ein Datumstyp eingeführt werden, der angibt, ob es sich um ein exaktes Datum, eine Kalenderwoche oder einen Monat handelt.

Abbildung 6-5 Datenbankmodell Projektvorschlag

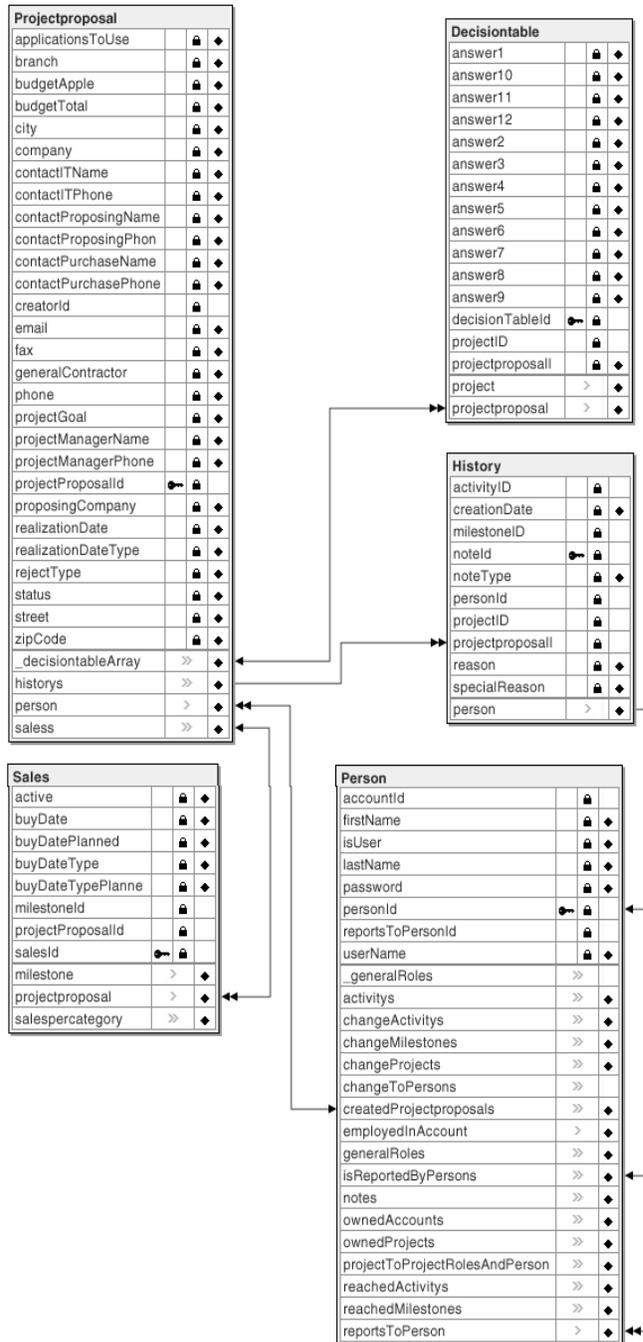


Abbildung 6–5 zeigt das Datenbankschema für den Projektvorschlag. Die Struktur der Verkäufe und der Entscheidungstabelle ist identisch zu der Projektverwaltung. Wenn ein *Projektvorschlag* in ein *Projekt* übergeht, werden die *Verkäufe* übernommen, einem *Milestone* zugeordnet und die *Entscheidungstabelle* mit dem *Projekt* verknüpft. *Verkäufe* in einem *Projektvorschlag* sind durch den Kontext natürlich Verkaufsannahmen, aber im Datenbankschema werden sie sowohl in der Projektverwaltung wie auch im *Projektvorschlag* durch eine Klasse *Verkäufe* repräsentiert.

6.3 Implementierung des Navigational Designs

In diesem Abschnitt wird gezeigt, wie das Navigational Design umgesetzt wurde.

Im Navigational Design wurden die Kontextklassen und ihre Beziehungen untereinander definiert. Wie in Abschnitt 4.3 beschrieben, gibt es sechs verschiedenen Kontextarten. In *WebObjects* ist eine Klasse vorhanden, welche die Funktionalität eines Kontexts sehr gut modelliert, die *WODisplayGroup*. Ihre Aufgabe ist es, Objekte einer Klasse anzuzeigen und zu verwalten. Hierzu gehören u.a. die Verwaltung einer Selektionsmenge, das Sortieren der angezeigten Objekte und die Auswahl der Objekte nach Suchkriterien. Außerdem besteht die Möglichkeit alle Objekte, die von einer *WODisplayGroup* verwaltet werden entweder nacheinander durchzunavigieren oder sie in disjunkte Gruppen von n Objekten aufzuteilen und dann durchzunavigieren. Deshalb wurde für die Implementierung die *WODisplayGroup* als Generalisierung eines jeglichen Kontexts ausgewählt. Nicht in allen Fällen kann jedoch eine *WODisplayGroup* alle Funktionalität eines Kontexts erfüllen. Im Folgenden soll aufgezeigt werden, wie die sechs Kontextarten zur Funktionalität einer *WODisplayGroup* passen und welche anderen Schritte notwendig sind, um einen Kontext in *WebObjects* abzubilden. Es sei hier erwähnt, daß die Klasse *NSMutableArray* auch über die Funktionalität Sortieren und Auswählen verfügt, jedoch im Gegensatz zur *WODisplayGroup* nicht direkt mit der Datenbank arbeitet.

- **Klassenbasierter Kontext:**

Dieser Kontext enthält alle Objekte einer Klasse C, welche durch ein gemeinsames Attribut ausgewählt werden. Diese Eigenschaft erfüllt die *WODisplayGroup*. Die Fixierung eines Attributs kommt einer Einschränkung der Anzeigemenge gleich. Dafür muß eine einschränkende Bedingung (Qualifier) für die *WODisplayGroup* gesetzt werden, welche die Form: "attribut = wert" hat. Für Kontexte dieses Typs wird in der Implementierung der Name NC<Klassenname> verwendet.
- **Klassenbasierte Kontextgruppe:**

Diese Gruppe von Kontexten wird durch die Enumerierung aller möglichen Werte eines gegebenen Attributs gebildet. Jeder Kontext der Gruppe zeichnet sich durch die Fixierung eines Wertes aus der Menge aller möglichen Werte für das gegebene Attribut aus. Da ein Kontext aus der Kontextgruppe sich durch eine bestimmte einschränkende Bedingung (Qualifier) für das gegebene Attribut auszeichnet kann eine *WODisplayGroup* seine Kontextidentität während der Laufzeit ändern, indem die entsprechende Bedingung während der Laufzeit gesetzt wird. Demnach ist es möglich, die Kontextgruppe mit einer *WODisplayGroup* zu simulieren. Zusätzlich muß die Menge aller einschränkenden Bedingungen in der *WODisplayGroup* gespeichert werden. Diese Art von Kontext wurde im Programm nicht verwendet.
- **Link-basierter Kontext:**

Diese Art des Kontexts wird von der *WODisplayGroup* defaultmäßig unterstützt. Anstatt die Anzeige von allen Objekten zu erlauben, kann auch spezifiziert werden, nur die Objekte für eine 1-n Relation X mit Masterobjekt Y zu erlauben. Hierfür wird das Masterobjekt und der Relationsname gesetzt. Kontexte dieses Typs wurden in der Implementierung als NC<Klassenname>Detail bezeichnet.
- **Link-basierte Kontextgruppe:**

In WebObjects ist diese Art von Kontextgruppe mit Hilfe einer Master-Detail Konfiguration von zwei *WODisplayGroups* zu implementieren. Die erste *WODisplayGroup* enthält hierbei die Masterobjekte der 1-n Relation und die zweite *WODisplayGroup* enthält die Detailobjekte der Relation. Es ist jedoch nicht möglich mit dieser Konfiguration mehrere Masterobjekte auszuwählen und dann durch die Detailobjekte zu navigieren. Eine Möglichkeit dies zu implementieren, ist die erste *WODisplayGroup* mit einem Array für alle Masterobjekte zu erweitern und die Detailobjekte in der zweiten *WODisplayGroup* per Hand zu setzen. Da alle Masterobjekte einen Verweis auf ihre Detailobjekte haben, ist das kein größerer Imple-

mentierungsaufwand. Die letztere Möglichkeit wurde jedoch im Programm nicht gebraucht.

- **Enumerierte Kontextklasse:**

Hierfür eignet sich die *WODisplayGroup* nicht, da sie nur Objekte einer Klasse Anzeigen kann. Diese Kontextklasse muß unter Verwendung von Standarddatentypen implementiert werden. Ein Denkansatz ist die Klasse *Array*, die auch Sortierungen ihrer Elemente zuläßt und einen Enumerator definiert.

- **Dynamische Kontextklasse:**

WODisplayGroup ist per se eine dynamische Kontextklasse in allen oben beschriebenen Varianten. Sie bietet die Methoden *insert* und *delete* an. Da die vorliegende Applikation sehr oft Objekte aus Kontexten löscht oder einfügt, ist es essentiell, daß die Implementierung der Kontexte dynamisch ist.

In OOHDM werden linkbasierte Kontexte für eine spezielle Relation definiert. Da in WebObjects jegliche Relation eingesetzt werden kann für die das Zielobjekt von der Klasse ist, die der Kontext verwaltet, ist nur eine linkbasierte Kontextklasse pro Datenbankklasse notwendig. Da auch die *WODisplayGroup* als klassenbasierter Kontext für die Einschränkung auf ein bestimmtes Attribut nur einen neuen Qualifier benötigt, wurde auch nur ein klassenbasierter Kontext pro Datenbankklasse implementiert.

6.4 Implementierung des User Interfaces

WebObjects unterstützt die Aufteilung einer Seite in Komponenten, die wiederum aus Komponenten aufgebaut werden können. Das Abstract Interface Design schlägt sich somit in der Aufteilung des Interfaces in Unterkomponenten wieder. Ein Beispiel der Überführung des Designs in die Implementierung gibt Abschnitt 6.6.2 auf Seite 104.

Das Abstract Interface Design teilt die Menüs schon in ein Hauptmenü und mehrere Untermenüs auf (Abbildung 5–14 auf Seite 75). Der Bildschirm wurde also in drei Bereiche mittels Frames unterteilt. Dabei wurde das Hauptmenü nicht oben sondern unten angelegt und jeweils ein Untermenü wurde auf der

linke Seite placiert. Die Menüs sind graphisch hervorgehoben durch einen eigenen Hintergrund. Umrandet von den Menüs befindet sich die Hauptarbeitsfläche, in der das aktuelle Fenster angezeigt wird. Da der Benutzer nicht im Hyperspace verloren gehen soll wurde im Menü zusätzlich eine Anzeige der zuletzt angezeigten Seiten implementiert. Die Anordnung der Menüs und des Hauptarbeitsbereichs wurde nach einer Studie einiger Webseiten nachempfunden. Auf multimediale, bessere Darstellungen wurde verzichtet, da sie sehr zeitaufwendig in der Erstellung sind und später hinzugefügt werden können. Des weiteren schaden sie der Effizienz (v.a. Java).

Weiterhin sollte die Benutzung möglichst durchgängig sein. Deshalb wurden verschiedene Icons ausgesucht (Tabelle 7-1), die verschiedene Operationen repräsentieren und diese durchgängig in der Applikation verwendet (z.B. der Stift bedeutet überall "Editieren"). Ein weiterer, viel benutzter Punkt sind die Detailansichten. Sie wurden mittels des speziellen Icons "Pfeil" realisiert. Siehe hierzu auch Abschnitt 6.6.3.

Es wird an dieser Stelle darauf verzichtet, die Interfacemasken abzubilden, da sie im Benutzerhandbuch im Abschnitt 7 schon dargestellt sind.

6.5 Die Implementierungstechnologien

6.5.1 WebObjects - ein Überblick

Als Implementierungstechnologie für das Projekt wurde WebObjects in der Version 4.0 eingesetzt. Für den Datenbankzugriff wird das Enterprise Object Framework (EOF) in der Version 3.0 benutzt. EOF ist die Datenbankmiddleware, welche über einen Adaptor auf eine relationale Datenbank zugreift. Es existieren Adaptoren für mehrere Datenbanken.

Die Entwicklungsumgebung ist für MacOS X Server und für Windows NT verfügbar. Die WebObjects Applikation kann jedoch auf einer viel breiteren Betriebssystembasis eingesetzt werden, da sie nur die Laufzeitumgebung ohne Graphical User Interface benötigt.

Für eine Webapplikation wird ein Webserver benötigt. WebObjects beinhaltet keinen Webserver. Es wird über die Webserverschnittstelle (CGI, NSAPI, IISAPI, WAI) an den jeweilig vorhandenen Webserver angeschlossen.

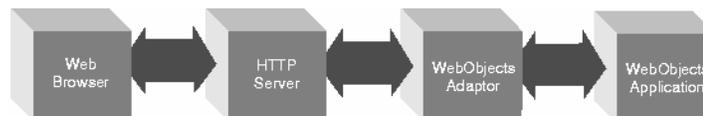
Weiterhin unterstützt WebObjects das Verteilen der Applikation auf mehrere Rechner und verwaltet die Rechnerlast (Load Balancing). Weiterhin kann mit dem Statistikmodul rudimentäre Analysen der Antwortzeit der Applikation erstellt werden.

Die unterstützten Programmiersprachen sind Java, Objective-C und WebScript. WebScript ist eine zur Laufzeit interpretierte Sprache mit Java-ähnlicher Syntax.

6.5.2 Struktur von WebObjects

WebObjects liegt zwischen Datenbank und Webserver. Es ist über die Webserverschnittstelle (z.B. CGI) an den Webserver angeschlossen und über EOF an die Datenbank. (siehe Abbildung 6–6)

Abbildung 6–6 Position einer WebObjects Applikation



Da das http Protokoll nicht statuserhaltend ist, muß WebObjects den Status sichern. Dies geschieht automatisch. Jedem Benutzer wird während seines Zugriffs auf die Applikation eine eindeutige Nummer zugeordnet (die Session ID), anhand derer WebObjects den Zustand der Applikation wiederherstellen kann.

WebObjects ist objektorientiert. Demnach wird auf der Applikationsebene nur mit Objekten gearbeitet. Eine Seite (oder auch Component) ist ein Objekt. Sie kann wiederum aus Unterkomponenten bestehen. Ein Menge von Grundkomponenten wie z.B. Eingabefelder, PopUp Buttons und Submit Buttons sind schon implementiert. Interessant ist es nun, aus diesen Grundkomponenten eigene,

wiederverwendbare, etwas größere Komponenten zu bauen und diese dann zu verwenden. WebObjects erlaubt eine Aufteilung der Seite in Subkomponenten, sie kann somit nach dem Baukastenprinzip erstellt werden.

Wie funktioniert nun die Generierung der Seiten? Eine Seite (Component) besteht aus folgenden Teilen:

- HTML Template: Es gibt die Grundstruktur der Seite an und enthält spezielle WebObjects Tags, die auf Objekte verweisen, welche im WebObjects Definitions File (wod) spezifiziert sind.
- Definition File: Es ist die Verbindung zwischen dem HTML Template und der Applikation. Hierin enthalten sind Definitionen von dem im HTML Template verwendeten Objekten. Die hier verwendeten Objekte sind entweder Grundkomponenten oder eigene Subkomponenten. Zu einer Definition gehören:
 - Name des Objekts
 - Name der Klasse des Objekts
 - Attributlisten für die Verknüpfung von Programmvariablen mit den internen Variablen des Objekts
- Eine Klasse, die von WOComponent erbt. Diese Klasse ist in entweder Java, Objective-C oder WebScript zu programmieren.

Weiterhin nehmen an der Seitengenerierung noch die Objekte *Application* und *Session* teil.

Zum Verständnis ist es besser, erst die Generierung einer Seite zu betrachten. Auf den Anfragezyklus wird danach eingegangen. Auf die Anfrage einer Seite wird zuerst die zur Seite gehörende Klasse instanziiert. Danach wird das HTML Template und das Definition File geparsed und in einem Syntaxbaum überführt. An der Wurzel steht das gerade instanziierte Seitenobjekt. Die Knoten sind die im HTML eingelagerten speziellen WebObjects Tags. Für jedes Tag wird die Definition im Definitionfile nachgeschlagen, die dort spezifizierte Klasse instanziiert und zum Knoten hinzugefügt. Beginnend bei der Wurzel wird nun der

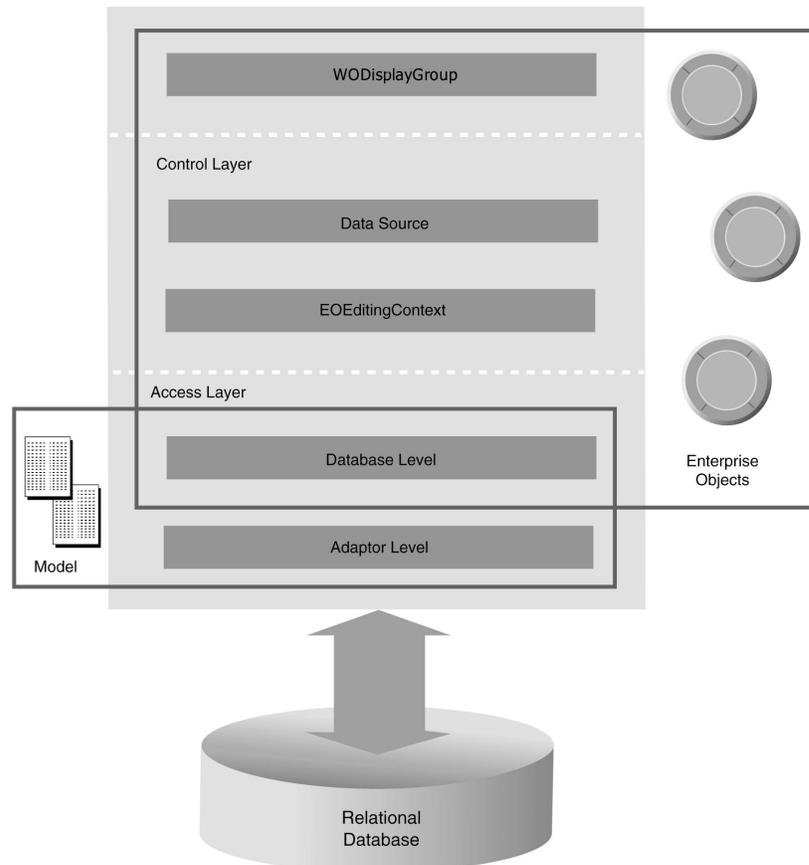
Befehl gegeben sich zu initialisieren, die Attributlisten, die im Definitionfile stehen abzuarbeiten, um danach das zugehörige HTML zu generieren. Damit ist die Generierung abgeschlossen.

Wie sieht nun der Anfang des Zyklusses aus? Ein Benutzeranfrage wird über den *WOAdaptor* an das *Application* Objekt als *Request* Objekt weitergereicht. Dieses findet nach der enthaltenen Session ID das zugehörige *Session* Objekt und reicht die Anfrage an das *Session* Objekt weiter. Die Session schaut nun in ihrem Speicher nach dem noch vorhandenen Baum (der von der vorigen Anfrage stammt). Falls das *Request* Eingabewerte des Benutzers beinhaltet, werden diese an das Wurzelobjekt weitergegeben. Anhand der Attributlisten weiß das Objekt, welche Werte für es bestimmt sind und reicht die Informationen an seine Kinder im Baum weiter die genauso verfahren. Zusätzlich zu Eingabewerten enthält ein Request immer ein Event (d.h. ein Link wurde geklickt, ein Submitbutton gedrückt oder ähnliches). Dieses Event ist einer bestimmten Subkomponente im Baum zugeordnet. In der Attributliste aus dem Definitionfile steht der Name der Methode, die bei diesem Event aufgerufen werden soll. Der Rückgabewert dieser Methode muß eine neue Seite in Form des Wurzelobjekts der neuen Seite sein. Um den vorher beschriebenen Mechanismus muß sich der Programmierer nicht kümmern. Er erlangt die Kontrolle durch den Methodenaufruf.

Wichtige Klassen für den Programmierer sind demnach die *Session*, die *Application* und die eigenen Subklassen von *WOComponent*.

6.5.3 Der Datenbankanschluß mit EOF

Abbildung 6–7 EOF Architektur



In Abbildung 6–7 ist die Architektur des Datenbankframeworks EOF dargestellt. Über ein Modell wird eine Abbildung von Tabellen zu Objekten definiert. Dieses Modell kann mit dem Entwicklungswerkzeug EOModeler bearbeitet werden. Zusätzlich zu den Tabellen nach Objekten Abbildung werden hierin auch die Relationen der einzelnen Objekte zueinander über Datenbankjoins definiert. Die Abbildung schließt auch die Übersetzung von internen Datenbanktypen zu den im Objekt vorhandenen Attributstypen ein. Das Ziel ist, sich nicht mehr um die Datenbank kümmern zu müssen, sondern mit dem auf dem EOControl Level angesiedelten objektorientierten API zu arbeiten.

Das einzige datenbankspezifische ist der Adaptor. Dieses Protokoll ist offen und es kann für jegliche Datenquelle ein Adaptor geschrieben werden. Auf Anfrage der oberen Frameworkschichten generiert der Adaptor SQL Aufrufe und verpackt die Resultate der Abfrage in Objekte. EOF hält die Objekte in einem Objektgraphen und notiert jegliche Änderungen, die vorgenommen werden. Der Programmierer arbeitet mit den Objekten der oberen Schicht des Frameworks.

Der EOEditingContext ist die verantwortliche Klasse, die den Objektgraphen verwaltet. Er repräsentiert eine Transaktionseinheit und notiert die Änderungen. Enterprise Objects (Datenobjekte) sind somit ganz normale Objekte mit dem kleinen Unterschied, daß sie zu einem EOEditingContext gehören und von ihm überwacht werden, wenn sie sich ändern. Der Benutzer dieser Objekte merkt jedoch nichts davon, daß sich EOF um die Persistenz kümmert. Der EOEditingContext bedient sich wiederum untergeordneter Klassen, um Datenbankobjekte zu holen und zu speichern. So wie auch Datenbanktransaktionen geschachtelt werden können, ist es auch möglich auch EOEditingContexts zu schachteln.

Zur Anzeige von Enterprise Objects gibt es die Klasse WODisplayGroup. Während sich der EOEditingContext um alle Objekte kümmert, beschränkt sich die WODisplayGroup auf eine Entität (Klasse), wie zum Beispiel die Anzeige von Personen. Sie bedient sich des EOEditingContexts, um Objekte zu holen und zu speichern. Die Aufgabe der WODisplayGroup ist:

- Vereinfachtes Interface des EOEditingContexts zur Verfügung zu stellen.
- Methoden zu Sortierungen und einschränkende Abfragen zu implementieren
- Möglichkeiten ein Suchresultat in Untermengen zu X Objekten anzuzeigen und durchzunavigieren
- Darstellung von Master-Detail Relationen wie z.B. Projekte zu Aktivitäten ermöglichen.
- Verwaltung einer Selektionsmenge des Benutzers.

Somit beschränkt sich die Interaktion mit dem Datenbankframework EOF hauptsächlich auf die zwei Klassen EOEditingContext und WODisplayGroup.

Für weitere Informationen kann die EOF Dokumentation von Apple herangezogen werden.

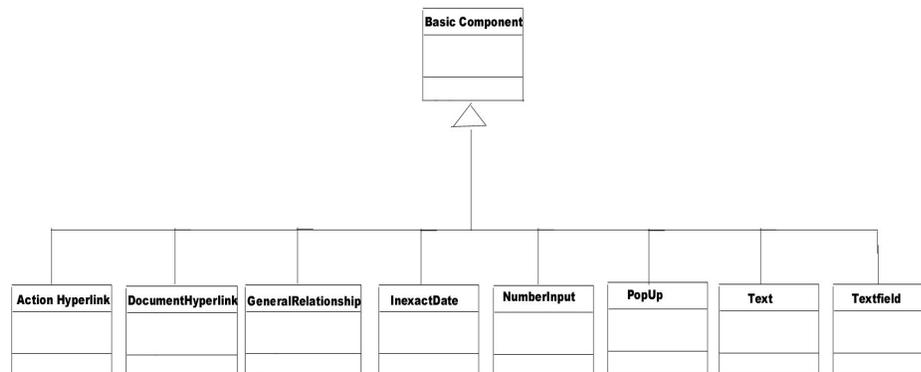
6.6 Einige Realisierungskonzepte

Bei der Realisierung haben sich einige Konzepte als sehr brauchbar erwiesen. Sie sollen hier kurz behandelt werden. Teilweise ist die Problematik nicht offensichtlich, sondern liegt vielmehr in WebObjects oder EOF.

6.6.1 Allgemeine Anzeigeelemente

Für die Anzeige und Eingabe verschiedener Attribute hat es sich angeboten aus den standardisierten Eingabe- und Ausgabeelementen angepaßte Ein- und Ausgabeelemente zu kreieren. Die Vererbungshierarchie dieser Komponenten ist in Abbildung 6–8 dargestellt. Sie haben ein Standard Interface, welches durch die Klasse *BasicComponent* definiert ist. Dazu gehört eine Variable *isInput*, welche definiert, ob das Element für die Eingabe oder Ausgabe dient, dazu das Datenobjekt und den Pfad zum Attribut. Der Hauptvorteil, der sich dadurch ergibt, ist die Konfiguration des Interfaces mit Attributlisten. Siehe dazu den nächsten Punkt.

Abbildung 6–8 Vererbungshierarchie der Anzeigeelemente



6.6.2 Konfiguration der Anzeige durch Attributlisten

Es gibt zwei Möglichkeiten, die Anzeigelemente zu gestalten. Entweder man kodiert alle Anzeigen per Hand oder entwirft ein generisches Element, welches die Anzeige beliebiger Objekte realisiert und durch eine Attributliste konfiguriert wird. Im Laufe der Implementierung hat sich gezeigt das ein Mittelweg das beste Maß ist.

In Abbildung 6–9 ist die Verwendung einer durch Attributlisten konfigurierten Anzeige dargestellt. In Abbildung 6–10 ist die zugehörige Attributliste zu sehen. Der grau unterlegte Bereich stellt die Eingabe des Enddatums dar, der durch die Komponente *InexactDate* dargestellt wird. Die Methode, die an den “OK” Button gebunden ist, braucht nur das *isInput* Attribut der Komponente zu verändern und steuert so, ob die Eingabemaske für ein Datum erscheint (wie der grau unterlegte Bereich) oder die Anzeige, die neben dem Anfangsdatum steht und nur das Datum formatiert.

Abbildung 6–9

Aktivität editieren

Aktivität editieren

<input checked="" type="checkbox"/> Anfangsdatum	3. Jan 1998								
Enddatum	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Tag</th> <th style="text-align: center;">Monat</th> <th style="text-align: center;">KW</th> <th style="text-align: center;">Jahr</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">11</td> <td style="text-align: center;">Jan</td> <td style="text-align: center;">none</td> <td style="text-align: center;">1998</td> </tr> </tbody> </table> <p style="text-align: center;"><input type="button" value="OK"/></p>	Tag	Monat	KW	Jahr	11	Jan	none	1998
Tag	Monat	KW	Jahr						
11	Jan	none	1998						
<input checked="" type="checkbox"/> Beschreibung	Roadshow								
<input checked="" type="checkbox"/> Aktivitätstyp	Sonstige								
<input checked="" type="checkbox"/> Status	geplant								
Verantwortliche Person	nicht eingegeben <input type="button" value="Suchen"/>								
<input type="button" value="OK"/> <input type="button" value="Zurück"/>									

Die Attributliste in Abbildung 6–10 besteht für jedes Element aus einem Titel (*title*), einer internen Bezeichnung des Attributs (*keypath*) und der Anzeigekomponente. So ist der Titel für das Attribut *startDateValue* “Anfangsdatum” und die anzeigenden Komponente ist *InexactDate*. GleichermäÙen lassen sich

die Werte für *endDateValue*, *description*, *activityType* und *status* aus der Attributliste lesen. Eine Erweiterung der Anzeige um ein Attribut ist einfach durch Hinzufügen eines neuen Eintrags in der Attributliste möglich.

Abbildung 6–10 Attributliste für das Editieren der Aktivitäten

```
activityDetails = { config =
  ( { title = "Anfangsdatum";
    keypath = "startDateValue";
    component = "InexactDate"; },
  { title = "Enddatum";
    keypath = "endDateValue";
    component = "InexactDate"; },
  { title = "Beschreibung";
    keypath = "description";
    component = "Text"; },
  { title = "Aktivitätstyp";
    keypath = "activityType";
    component = "PopUp"; },
  { title = "Status";
    keypath = "status";
    component = "PopUp"; }
) };
```

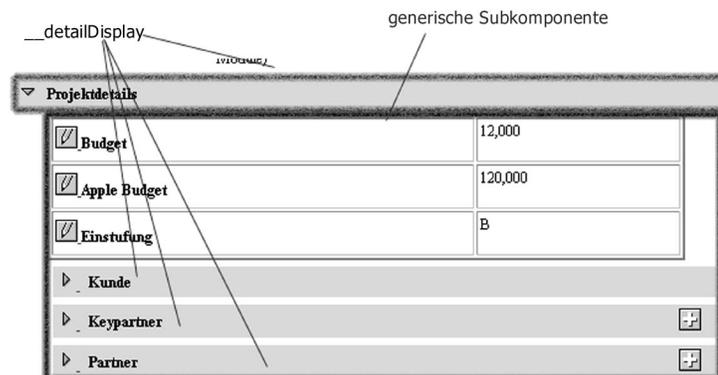
Folgende Komponenten benutzen Attributlisten, um Werte darzustellen:

- *__editDisplayAttributeRow*, welche eine Zeile in einer Tabelle mit dem jeweiligen Titel und dazugehörigen Attribut anzeigt. Außerdem ermöglicht es das Editieren des Attributs. Diese Komponente kommt in Aktivität editieren, Projektverwaltung Hauptseite und Milestone editieren vor.
- *__detailTable*, *DetailRow*, *DetailRowDetail*, *__headersRow*, welche eine Tabelle mit Überschrift und den dazugehörigen Attributen anzeigt. Dies ist eine der meistgenutzten Komponenten. Sie kommt in Projektsuche, Projektverwaltung Hauptseite, Milestone editieren, Aktivität editieren, Erinnerungen anzeigen, Erinnerungsdetails anzeigen und der Personen- und Accountsuche vor.
- *__inputAttributeRow*, welche ein Attribut anzeigt, mit der Möglichkeit es zu editieren. Sie wird in der Hauptseite der Projektverwaltung und bei der Editierung von Milestones eingesetzt.

6.6.3 Detailanzeigen

Eine vielgenutzte Komponente ist die *_detailDisplay* Komponente. Sie wird bei der Projektverwaltung, beim Editieren von Milestones, beim Editieren von Aktivitäten, bei der Anzeige von Erinnerungen und der Anzeige der Details von Erinnerungen benutzt. In Abbildung 6–11 ist eine Benutzung der Detailanzeige in der Applikation exemplarisch herausgegriffen. Die Anzeige besteht aus einem Balken mit Titel und dem Pfeil für das An- und Ausschalten der Detailanzeige. Die Komponente ist so entworfen worden, daß die Detailansicht irgend eine andere Komponente sein kann. Alle Detailansichten haben eine gemeinsame Schnittstelle, um Informationen von der *_detailDisplay* zu bekommen. Deshalb erben alle Komponenten, die als Detailansicht eingesetzt werden sollen, von der Klassen *DetailDisplayRoot*.

Abbildung 6–11 Beispiel der Benutzung der Detailanzeige



In der Abbildung sieht man deutlich, daß die hier verwendete Detailansicht wiederum die Komponente *_detailDisplay* enthält. Es ist außerdem noch möglich der Komponente *_detailDisplay* ein Aktionsitem (in diesem Fall das “Plus” Bild) mit zugehörigem Methodenaufruf zuzuweisen.

6.6.4 Konsistenz der Datenbank sicherstellen

Ein wichtige Aufgabe ist, die Konsistenz des Objektgraphen, also die Konsistenz der Datenbank, zu gewährleisten. Das Problem stammt von den weitreichenden Navigationsmöglichkeiten, welche durch die Navigationshilfe gegeben sind. Als Anschauungsbeispiel stelle man sich den Vorgang des Hinzufügens eines Keypartners vor. Der Benutzer ist im Projekt Management Hauptfenster und möchte einen Keypartner hinzufügen. Die Suche schlägt fehl und er gibt ein neuen Keypartner ein. Das Programm kreiert ein neues Partner Datenbankobjekt. Während der Eingabe möchte er noch die Muttergesellschaft hinzufügen, die auch noch nicht in der Datenbank vorhanden ist. Auch hier wird ein Partner Datenbankobjekt erstellt. Wenn alles nach Plan läuft, müßte der Benutzer nun zweimal die Eingabe bestätigen, die Daten werden validiert und in die Datenbank eingetragen. Falls er aber zurück auf die Projekt Management Hauptseite mit Hilfe des Navigationsmenüs zurückspringt, kommt es zu Inkonsistenzen. Das Programm hält nun zwei Datenbankobjekte, die nicht mehr gültig sind und deren Attribute nicht validiert sind.

Es muß also eine möglichst allgemeingültige Lösung gefunden werden, um die Konsistenz des Objektgraphen in solchen Fällen sicherzustellen. EOF unterstützt das Verwerfen von Änderungen in einem EOEditingContext, der einer Transaktion entspricht. Im Programm hat nun jede Seite, in der etwas eingegeben wird, ihren eigenen EOEditingContext, d.h. ihr eigenes Transaktionsfenster, das in den vorigen EOEditingContext geschachtelt ist. Die Seite merkt sich auch ihre Schachteltiefe. Wird auf die Seite zurücknavigiert, werden alle Änderungen in tiefer geschachtelten EOEditingContexts zurückgenommen. Das hat keinen Effekt, wenn die vorigen Seiten ihre Änderungen gespeichert haben. Falls das nicht der Fall ist, wie im obigen Beispiel, werden die so kreierten Objekte zerstört und die Konsistenz der Datenbank ist wieder hergestellt.

7. Benutzerhandbuch

In diesem Kapitel wird auf die Benutzung des Programms eingegangen und das Aussehen der Applikation durch Bildschirmfotos dokumentiert und beschrieben.

7.1 Allgemeine Bedienhinweise

Das System ist für den Einsatz von Netscape Navigator 4.0 bei einer Auflösung von 800x600 Pixel optimiert worden. Bei der Benutzung von anderen Browsern kann es zu leichten Fehlanzeigen kommen, was die Funktionalität aber nicht beeinträchtigt.

Das Fenster ist in drei Frames unterteilt:

- Detailmenü mit der Historie
- Navigationsleiste für die Teilbereiche des Programms
- Hauptfenster, in dem gearbeitet wird

Das Detailmenü enthält zusätzlich die Historie. Sie ist mit grünen Punkten gekennzeichnet. Hier wird die ganze Navigation, die sich im Hauptfenster abspielt, aufgezeichnet.

Tabelle 7-1 Icons und ihre Bedeutung

Icon	Beschreibung
	Dieses Icon stellt den Befehl "kreieren" dar. Der Befehl bezieht sich auf den Datensatztyp, der auf der gleichen Höhe steht wie das Symbol.
	Dieses Icon stellt den Befehl "editieren" dar. Es bezieht sich entweder auf einen ganzen Datensatz oder nur auf ein Attribut, je nach dem ob es in einer Zeile steht oder neben einem Wert.
	Wenn vom Benutzer eine Wahl aus mehreren Möglichkeiten (z.B. Auswahl einer Person nach einer Suche) erwartet wird, erscheint der Haken in der Zeile des auszuwählenden Datensatzes.
	Dieses Icon stellt den Befehl "löschen" dar. Es bezieht sich auf den Datensatz, welcher in der selben Zeile steht, wie das Symbol.
	Die Lupe bezieht sich auf den Befehl "suchen". Mit Suchen ist hier meistens die Suche nach einem neuen Datensatz für die Relation, welche mit der Lupe in einer Zeile steht, gemeint.
	Der Pfeil <i>nach rechts</i> kennzeichnet eine geschlossene Detailansicht. Um sie zu öffnen, muß der Benutzer das Icon anklicken. Daraufhin öffnet sich die Detailansicht.
	Der Pfeil <i>nach unten</i> markiert eine offene Detailansicht. Um zur geschlossenen Detailansicht zurückzukehren, muß der Pfeil wiederholt angeklickt werden.
	Dieses Zeichen ist sehr oft in Überschriften von Tabellen zu finden. Es zeigt, daß die Tabelle nicht nach dem Attribut sortiert ist. Ein Klicken auf das Icon verändert die Sortierung. Die Tabelle ist dann absteigend nach dem Attribut sortiert (siehe nächste Icons).
	Die Tabelle ist in absteigender Reihenfolge nach dem Attribut sortiert und wechselt als nächstes zur aufsteigenden Sortierung.
	Die Tabelle ist in aufsteigender Reihenfolge nach dem Attribut sortiert und wechselt danach wieder zurück zu einer Tabelle ohne Sortierungen.
	Dies ist das Zeichen für das Bestätigen einer Erinnerung. Sie wird durch das Anklicken gelöscht.

Im System werden durchgängig die in Tabelle 7-1 gezeigten Icons verwendet. Alle, als Hyperlink angezeigten Daten verweisen auf eine Detailansicht. So verzweigt das Klicken auf den Hyperlink mit dem Namen eines Accounts auf die volle Accountinformation. Sie ist jedoch nicht editierbar.

Die Eingabelemente für Datumsangaben erlauben meistens eine Spezifikation in drei Genauigkeitsstufen. Beim exakten Datum muß die Kalenderwoche auf "none" gelassen werden. Bei der Angabe des Monatsnamens ohne weitere Festlegung muß die Kalenderwoche und der Monatstag auf "none" stehen. Die Eingabe der Kalenderwoche erfordert nicht die Eingabe des Monatsnamens oder des Monatsstags.

In der Menüleiste ist immer der Befehl "Alle Detailansichten öffnen" vorhanden. Sie können damit alle Detailansichten aufklappen. Mit dem Befehl "Alle Detailansichten schließen" werden sie wiederum zugeklappt.

7.2 Starten der Applikation

Nach der erfolgreichen Verbindung mit der Applikation, wird ein zweites Fenster geöffnet, welches genau die Größe von 800x600 Pixel hat und auf jegliche Werkzeugleisten verzichtet (Abbildung 7-1). Zuerst muß sich der Benutzer mit seinem Namen und Paßwort identifizieren, um Zugriff zu bekommen.

Abbildung 7-1 Login



7.3 Erinnerungen

Nach dem erfolgreichen Anmelden, bekommt der Benutzer seine persönliche Erinnerungen präsentiert (Abbildung 7-2). Hierzu zählen alle Änderungen, die in seinen Projekten passiert sind (geänderte Projektdetails, Milestones oder Aktivitäten). Eine Person gehört zu einem Projekt, wenn sie in diesem eine Rolle spielt. (siehe auch Abschnitt 7.5). Als weitere Erinnerung sind alle Aktivitäten, die von dem Benutzer erledigt werden müssen und sich noch in der Planungsphase befinden, aufgelistet. Das Bestätigen-Symbol entfernt die zugehörige Erinnerung aus der aktuellen Liste.

Abbildung 7-2 Erinnerungen

The screenshot shows a Netscape browser window titled "Netscape: Sheila - Projektinformationssystem". The main content area is titled "Erinnerungen" and features a lightbulb icon. It is organized into several sections:

- Geänderte Projekte**: 0 items.
- Geänderte Milestones**: 1 item.
- Geänderte Aktivitäten**: 2 items. This section contains a table with columns for "Aktivitätstyp", "Beschreibung", "Person", and "Projekt".

Aktivitätstyp	Beschreibung	Person	Projekt
Sonstige	Weiter so	Christian van der Leeden	CMS
<i>Aktivität mit Beschreibung "Weiter so" wurde am 11.08.1998 angelegt</i>			
Sonstige	Produktvorführung beim Kunden	Christian van der Leeden	INetDesign
<i>Aktivität mit Beschreibung "Produktvorführung beim Kunden" wurde am 14.08.1998 angelegt</i>			
- Geplante Aktivitäten**: 5 items. This section contains a table with columns for "Projektname", "Startdatum", "Beschreibung", and "Typ".

Projektname	Startdatum	Beschreibung	Typ
	1. Feb 1998	Projektinformationen sammeln	Sonstige
	Nov 1998	Beim Support alles nötige veranlassen	Sonstige
CMS	Jan 1998	Weiter so	Sonstige
	Feb 1998	Weiterhin gut auftreten	Sonstige
INetDesign	1. Mar 1998	Produktvorführung beim Kunden	Sonstige

At the bottom of the browser window, there is a navigation bar with the following links: "Vertriebs-informationssystem", "Erinnerungen", "Projekte", "Accounts", "Optionen", "Projektvorschlag", "Reports", "Personen", and "Produkte". An Apple logo is visible on the right side of the navigation bar.

Beim Klicken auf den Button "Detail" wird die Aktivität (bzw. Milestone oder Projekt) mit allen Daten angezeigt. Zusätzlich können noch die Projektdaten bzw. die Milestonedaten eingesehen werden (Abbildung 7-3). Muß der Benutzer die Daten bearbeiten, kann er sofort in die Projektverwaltung springen und dort die Änderungen durchführen.

Abbildung 7-3 Detailerinnerungen



7.4 Projektvorschlag

Jedes Projekt beginnt mit einem Projektvorschlag (Abbildung 7-4), der entweder vom Händler oder von einem Apple Mitarbeiter eingegeben werden kann. Der Menüpunkt "Projektvorschlag eingeben" startet die Aktion. Zuerst müssen alle Informationen für den Projektvorschlag eingegeben werden. Falls dies nicht der Fall ist, wird die weitere Bearbeitung mit einer Fehlermeldung nicht zugelassen. Die einzugebenden Daten sind in Abbildung 7-5 aufgelistet.

Abbildung 7-4 Projektvorschlag

Projektvorschlag

Sie haben hier die Möglichkeit einen Projektvorschlag einzubringen. Bitte füllen sie alle Felder aus. Der Vorschlag kann nur angenommen werden, wenn alle Felder ausgefüllt sind.

Vorschlagende Firma

Firmenname

Kontaktperson:

Telefonnummer für Rückfragen

Kundendaten

Firma

Branche

Strasse

PLZ/Ort

Vertriebs-informationssystem | Erinnerungen | Projekte | Accounts | Optionen | Projektvorschlag | Reports | Personen | Produkte |

Der Projektvorschlag besteht aus drei Teilen: Information zur vorschlagenden Firma (z.B. der Händler), Informationen zum Kunden und Informationen über das Projekt. Bei den Projektinformationen muß das Realisierungsdatum eingegeben werden. Es kennzeichnet das Ende des Projekts. Das Anfangsdatum braucht nicht eingegeben zu werden. Die allgemeine Projektgröße ist das Volumen des Gesamtprojekts, wobei der Anteil von Apple nur die Leistungen umfaßt, welche im Zuge des Projekts von Apple an den Kunden verkauft werden sollen. Es ist darauf zu achten, daß die hier eingegebenen Zahlen nicht zu stark mit den später einzugebenden Verkäufen in Diskrepanz stehen. Jedoch kann im Nachhinein ein Projektvorschlag immer noch editiert werden.

Abbildung 7-5

Projektvorschlag eingeben

Projektvorschlag

Sie haben hier die Möglichkeit einen Projektvorschlag einzubringen. Bitte füllen sie alle Felder aus. Der Vorschlag kann nur angenommen werden, wenn alle Felder ausgefüllt sind.

Vorschlagende Firma

Firmenname

Kontaktperson:

Telefonnummer für Rückfragen

Kundendaten

Firma

Branche

Strasse

PLZ/Ort

Telefon

E-Mail

Fax

Ansprechpartner EDV

Name

Telefon

Ansprechpartner Einkauf

Name

Telefon

Projektleiter

Name

Telefon

Weiter Projektinformationen

Wann soll das Projekt realisiert werden?

Monat Jahr

Projektvolumen

möglicher Apple Anteil

Welches Ziel will der Endkunde erreichen?

Welche Applikationen sollen eingesetzt werden?

Gibt es eine Generalunternehmer, wenn ja wie heisst er?

weiter

Abbildung 7-6 Entscheidungstabelle eingeben

Netscape: Sheila - Projektinformationssystem

Entscheidungstabelle

Kriterien	10 Punkte	6 Punkte	4 Punkte	2 Punkte
Strategische Bedeutung	<input checked="" type="radio"/> Strategische Entscheidung für die ganze Firma	<input type="radio"/> Strategische Entscheidung für eine Teil der Firma	<input type="radio"/> Projektentscheidung des Accounts	<input type="radio"/> Noch nicht definiert
Vollständigkeit der Lösung	<input type="radio"/> Erfüllt die Bedürfnisse der Firma	<input checked="" type="radio"/> Erfüllt teilweise die Bedürfnisse der Firma	<input type="radio"/> Eine besondere Lösung ist in Entwicklung	<input type="radio"/> Eine besondere Lösung wird gewünscht
KO Kriterien	<input checked="" type="radio"/> Erfüllen nur wir	<input type="radio"/> Könnte von uns erfüllt werden	<input type="radio"/> Wir können nicht alle Bedingungen erfüllen	<input type="radio"/> Wir können die Bedingungen nur mit grossem Kostenaufwand erfüllen
Lösung der Wettbewerber	<input type="radio"/> Schlechter	<input checked="" type="radio"/> Ähnlich	<input type="radio"/> Besser	<input type="radio"/> Weit besser
Wichtige Partner	<input checked="" type="radio"/> Ziehen uns vor	<input type="radio"/> Sind neutral	<input type="radio"/> Ziehen die Wettbewerber vor	<input type="radio"/> Sind eindeutig auf der Seite der Wettbewerber
Mentor	<input type="radio"/> Entscheider und Fachleute stehen uns positiv gegenüber	<input checked="" type="radio"/> Entscheider oder Fachleute stehen uns positiv gegenüber	<input type="radio"/> Ein Mentor wird gerade aufgebaut	<input type="radio"/> Es gibt keinen Mentor
Zugang zu den Entscheidern	<input type="radio"/> Entscheider verhandeln direkt mit uns	<input type="radio"/> Entscheider innerhalb des Fachgremiums verhandeln mit uns	<input checked="" type="radio"/> Des Fachgremium ist noch nicht definiert	<input type="radio"/> Die Entscheidung fällt in einem anderen Bereich der Firma
Projektkoordination	<input checked="" type="radio"/> Wir leiten das Projekt	<input type="radio"/> Partner leiten das Projekt	<input type="radio"/> Es gibt noch keine Projektleitung	<input type="radio"/> Wettbewerber leiten das Projekt
Preis	<input checked="" type="radio"/> innerhalb des Budgets	<input type="radio"/> leicht über dem Budget	<input type="radio"/> deutlich über dem Budget	<input type="radio"/> sehr weit über dem Budget
Preisvergleich mit Wettbewerbern	<input checked="" type="radio"/> Gleich oder niedriger	<input type="radio"/> etwas höher	<input type="radio"/> Mehr als 25 % höher	<input type="radio"/> Mehr als 50 % höher
Wie gross ist das Projekt	<input type="radio"/> grösser als 2 Mio DM	<input checked="" type="radio"/> 500 kDM bis 2 Mio DM	<input type="radio"/> 100 kDM bis 500 kDM	<input type="radio"/> kleiner als 100 kDM
Zeitachse der Entscheidung	<input type="radio"/> Entscheidung in diesem Quartal	<input type="radio"/> Entscheidung im nächsten Quartal	<input checked="" type="radio"/> Entscheidung in den nächsten 12 Monaten	<input type="radio"/> Entscheidung wird später fallen

Vertriebs-
informations-
system

Erinnerungen	Projekte	Accounts	Optionen
Projektvorschlag	Reports	Personen	Produkte

Nach der erfolgreichen Eingabe der Projektvorschlagsdaten, muß nun die subjektive Wahrscheinlichkeit des Projekts gemessen werden. Dies geschieht durch die Beantwortung der in Abbildung 7-6 gezeigten Fragen.

Abbildung 7-7 Verkäufe eingeben

Netscape: Sheila - Projektinformationssystem

Verkaufsplanung eingeben

←
Delete
2 - 2
→

Realisierungszeitraum:

Tag	Monat	KW	Jahr
none	Dez	none	1998

geplante Verkäufe

Produktkategorie	Preis	Geplant
Lowend	3,000.00	0
Monitore	4,000.00	10
Midrange	7,000.00	0
Highend	10,000.00	10
Powerbooks	7,000.00	0
Drucker	4,000.00	0
A	0.00	0
Summe Stückzahlen		20
Geschätzter Umsatz		140,000 DM

Fertig
weitere Verkaufsplanung

Vertriebs-
informati-
ons-
system

Erinnerungen

Projekte

Accounts

Optionen

Projektvorschlag

Reports

Personen

Produkte

Nach der Bestimmung der Wahrscheinlichkeit des Projekts müssen die zu erwartenden Verkäufe eingegeben werden (Abbildung 7-7). Eine Mehrfacheingabe ist möglich. Das geplante Verkaufsdatum ist für jeden Verkauf unter "Realisierungszeitraum" einzugeben. Mit den grünen Pfeiltasten kann zwischen den eingegebenen Verkäufen navigiert werden. Mit dem "Löschen" Button können schon eingegebene Verkäufe aus der Planung genommen werden. Es muß jedoch mindestens ein Verkauf übrig bleiben. Mit "Fertig" wird die Projektvorschlagseingabe abgeschlossen und in der Datenbank abgespeichert.

Abbildung 7-8

Projektvorschlagsanzeige

Offene Projektvorschläge

	Datum	Von wem?	Kunde	Projektbeschreibung	
▶	Feb 1999	HSD	Hasit Trockenmoertel	Internet basiertes Bauen	ablehnen
▶	Dez 1999	Computer & Co	TU Munchen	Neueinführung Studiengang Multimedia Design	ablehnen
▶	Dez 1998	Muller Computer	Springer Verlag	Neues Redaktionssystem	ablehnen
▼	Dez 1998	Lustig Computer	Aloah Design	Einrichtung von Multimediaarbeitsplätzen	ablehnen

Vorschlagende Firma	Lustig Computer
Ansprechpartner:	Peter Lustig Tel:089-25345
Firma	Aloah Design
Branche	Internetdesign
Adresse	Kurfürstenplatz 1032 80034 München Tel: 089-2345234 Email:aloha@design.de
Ansprechpartner EDV	Maria Schnell Tel:089-234524
Ansprechpartner Einkauf	Florian Maier Tel:089-2345234
Projektleiter	Maria Eberle Tel:089-2345243
Realisierungsdatum	Dez 1998
Projektvolumen	350.000 DM
Möglicheranteil von Apple	250.000 DM
Ziel des Kunden	Einrichtung von Multimediaarbeitsplätzen
Eingesetzte Applikationen	Cyberstudio 3.0 Photoshop 5.0 Appleshare IP
Generalunternehmer	Lustig Computer
Eintrittswahrscheinlichkeit nach Sarismodell	50 %

Historie des Projektvorschlags		
Datum	Person	Beschreibung
14. Aug 1998	Christian van der Leeden	Ersteingabe des Projektvorschlags

Geändert von am:

geplant bis: Sep 1998

Produktkategorie	Durchschnittspreis	Geplante Stückzahl
Lowend	3,000	0
Monitore	4,000	0
Midrange	7,000	0
Highend	10,000	10
Powerbooks	7,000	0
Drucker	4,000	10
A		0
Summe Stückzahlen		20
Summe Umsatz		140,000

geplant bis: nicht eingegeben

Produktkategorie	Durchschnittspreis	Geplante Stückzahl
Lowend	3,000	0
Monitore	4,000	0
Midrange	7,000	0
Highend	10,000	0
Powerbooks	7,000	0
Drucker	4,000	0

Eingegebene Projektvorschläge können unter dem Menüpunkt “offenen Projektvorschläge” angesehen werden. (Abbildung 7–8). Es werden alle Daten angezeigt. Durch Anklicken der Prozentzahl der Eintrittswahrscheinlichkeit wird die Entscheidungstabelle angezeigt. (Abbildung 7–6). Weiterhin kann ein Projektvorschlag editiert werden, hierzu dient das “Editieren” Symbol. Es werden die gleichen Schritte wie bei der Eingabe eines Projektvorschlags durchlaufen.

Abbildung 7–9 Projektvorschlag ablehnen

The screenshot shows a Netscape browser window titled "Netscape: Sheila - Projektinformationssystem". The main content area is titled "Projektvorschlag ablehnen". Below the title, there is a text box for a standard reason, currently set to "Der Aufwand fuer das Projekt ist zu klein, um anerkannt zu werden". A larger text area for "Zusätzlicher Kommentar" is empty. A "Projektvorschlag ablehnen" button is visible. Below the form, the details of the proposing company are listed: "Vorschlagende Firma: Lustig Computer", "Ansprechpartner: Peter Lustig, Tel: 089-23454", "Firma: Aloah Design", "Branche: Internetdesign", and "Adresse: Kurfürstenstr. 1032, 80034 München, Tel: 089-2345234, Email: aloah@design.de". At the bottom, there is a navigation bar with buttons for "Vertriebs-informationssystem", "Erinnerungen", "Projekte", "Accounts", "Optionen", "Projektvorschlag", "Reports", "Personen", and "Produkte", along with an Apple logo.

Ein Projektvorschlag ist nach der Eingabe als “offen” markiert. Nach der Revision durch einen Accountmanager kann er entweder angenommen (Abbildung 7–10) oder abgelehnt werden. Durch betätigen des Button “Ablehnen” wird der Projektvorschlag nach der Spezifikation einer Begründung (Abbildung 7–9) als abgelehnt markiert. Weiterhin wird die Historie des Projektvorschlags fortgeschrieben und die Ablehnung so dokumentiert. Unter dem Menüpunkt “abgelehnte Projektvorschläge” befinden sich alle als abgelehnt markierte Projektvorschläge. Durch die Option “Wiedervorlegen” kann ein abgelehnter Projektvor-

schlag wieder in die Liste der offenen Projektvorschläge übernommen werden. Auch hier wird die Historie fortgeschrieben und ist in der Detailansicht des Vorschlags sichtbar.

7.5 Projektverwaltung

Abbildung 7-10 Projektvorschlagsanzeige zur Projekteingabe



Um aus einem Projektvorschlag ein Projekt zu kreieren, wird in die Projektverwaltung gewechselt (Navigationsleiste "Projekte"). Unter dem Menüpunkt "Neues Projekt eingeben" erscheint eine Liste aller offenen Projektvorschläge (Abbildung 7-10). Durch die Selektion eines Vorschlags mit dem Selektionsymbol wird der Vorschlag als Grundlage zur Eingabe eines Projekts genommen.

Der nächste Schritt ist, die fehlenden Informationen einzugeben (Abbildung 7-11). Die Projektnummer wird automatisch generiert. Der Projektname muß vom Accountmanager selber vergeben werden. Weiterhin muß das Anfangsdatum

eingetragen werden. Ein Abgleich mit der Accounts- und Personendatenbank ist erforderlich. Das Anlicken der Lupe neben dem Punkt "Kunde" führt zur Kundensuche. (Abbildung 7-12) Hier können auch Kunden neu angelegt werden. Der eingetragene Accountmanager übernimmt automatisch den Kunden zu seinen bereits betreuten Kunden. Um die Suche zu erleichtern, werden die Daten aus dem Projektvorschlag zur Hilfe in das Feld mit dem Firmennamen eingefügt.

Abbildung 7-11 Projekt eingeben

The screenshot shows a web browser window titled "Netscape: Sheila - Projektinformationssystem". The main content area is titled "Projekt eingeben" and contains the following form fields:

- Projektnummer:** P-199808151749
- Projektname:** INet Design
- Beschreibung:** Einrichtung von Multimediaarbeitsplätzen
- Anfangsdatum:** Tag: none, Monat: Jan, KW: none, Jahr: 1998
- Enddatum:** Tag: none, Monat: Dez, KW: none, Jahr: 1998
- Kunde:** Aloah Design
- Projektmitarbeiter beim Kunden:**
 - Ansprechpartner Einkauf: Florian Meier
 - Ansprechpartner EDV: Maria Schnell
 - Projektleiter: Maria Eberle
- Budget:** 350 kDM
- Anteil von Apple:** 250 kDM
- Projektbonus:** (empty field)
- Einstufung:** C
- Projekttemplate:** Kein Template

At the bottom of the form is a button labeled "weiter zur Projektverwaltung". Below the form is a navigation bar with the following items:

Vertriebs- informations- system	Erinnerungen	Projekte	Accounts	Optionen	Apple Logo
	Projektvorschlag	Reports	Personen	Produkte	

Nach der Eingabe des Kunden müssen nun die Projektmitarbeiter auf Seiten des Kunden eingegeben werden. Es erscheint eine ähnliche Maske wie in Abbildung 7-12 gezeigt. Auch hier ist es möglich, eine Person neu anzulegen. Neue Personen haben per se keine Benutzerrechte. Dies muß in der Personenverwaltung separat festgelegt werden.

Abbildung 7-12 Kunden suchen und hinzufügen

The screenshot shows a web application interface for searching and adding accounts. The main heading is "Account suchen". Below it are several input fields: "Firmenname" (containing "Aloah Design"), "Niederlassung", "Land", and "Firmenkürzel". There are two buttons: "Suchen" and "Neuer Account". Below these is a table with columns "Firmenname", "Niederlassung", and "Account Typ". The table contains two rows: "BMW" and "Aloah Design", both with "Kunde" in the "Account Typ" column. A sidebar on the left contains navigation links like "Neues Projekt eingeben", "Projekttemplate", "Projekte verwalten", "Projekt eingeben", "Account auswählen", and "Alle Details anzeigen öffnen". At the bottom, there is a navigation bar with buttons for "Erinnerungen", "Projekte", "Accounts", "Optionen", "Projektvorschlag", "Reports", "Personen", "Produkte", and an Apple logo.

Sind die Kundendaten mit der Datenbank abgeglichen, kann das Projekt unter Spezifizierung eines Projekttemplates in ein neues Projekt überführt werden. Die Eingabe des Projektbonus ist nicht zwingend erforderlich.

Die im Projekttemplate enthaltenen Aktivitäten und Milestones werden in das neue Projekt übernommen. Der Zeitraum des Projekttemplates wird auf den Zeitraum des Projekts skaliert und das Datum der Aktivitäten und Milestones geändert. Zusätzlich zu den Milestones aus dem Projekttemplate wird jeder geplante Verkauf in einen Absatzplanungsmilestone überführt. Das Projekt gehört nun dem Accountmanager, der es eingegeben hat.

Abbildung 7-13 Projektverwaltung Übersicht



Die Verwaltung eines Projektes enthält folgende Bestandteile: die Projektdetails, die Verkäufe für das Projekt, die Notizen, Aktivitäten und Milestones für das Projekt. Für jeden Teil gibt es eine Detailansicht. Die Zahl in dem Balken kennzeichnet die Anzahl der Elemente, die sich hinter der Detailansicht verbergen. Hier erweist sich der Befehl "Alle Detailansichten öffnen" als hilfreich, da damit einen Überblick über alle wichtigen Projektdaten gewonnen werden kann.

Abbildung 7-14 Projektverwaltung - Milestones und Aktivitäten



Projekt verwalten

Projektnummer: P-199808141838
 Projektname INetDesign
 Projektbeschreibung Einrichtung von Multimediaarbeitsplätzen
Kunde: Aloah Design
Account Manager: Christian van der Leeden
Abteilung: Corporate Accounts

▶ **Projektdetails**

▶ **Verkäufe für das Projekt**

▶ **Notizen für das Projekt** 0

▼ **Aktivitäten für das Projekt** 2

	Anfangsdatum	Beschreibung	Status	Typ	
▶	1. Feb 1998	Projektinformationen sammeln	aus Projektvorschlag	Sonstige	<input checked="" type="checkbox"/>
▶	Jan 1998	Den Aha Effekt produzieren	aus Projektvorschlag	Sonstige	<input checked="" type="checkbox"/>

▼ **Milestones für das Projekt** 6

	Datum	Beschreibung	Milestonetyp	Status	
▶	Sep 1998	Automatisch Generierter Milestone, bitte anpassen!	Absatzplanungsmilestone	geplant	<input checked="" type="checkbox"/>
▶	Dez 1998	Automatisch Generierter Milestone, bitte anpassen!	Absatzplanungsmilestone	geplant	<input checked="" type="checkbox"/>
▶	Feb 1998	Ansprechpartner, Budget etc. sind bekannt	Phase I: Aqise	aus Projektvorschlag	<input checked="" type="checkbox"/>
▶	Mai 1998	Der Kunde hat 2-3 Lösungsmöglichkeiten erstellt.	Phase III: Konzept- und Lösung	aus Projektvorschlag	<input checked="" type="checkbox"/>
▶	Okt 1998	Der Auftrag gehört UNS!	Phase IV: Angebot und Vertragsabschluss	aus Projektvorschlag	<input checked="" type="checkbox"/>
▶	31. Dez 1998	Die Lösung läuft	Phase V: Auftragsabwicklungs	aus Projektvorschlag	<input checked="" type="checkbox"/>

Die Teile *Aktivitäten* und *Milestones* geben ein Übersicht über die Planung des Projekts. Alle Aktivitäten und Milestones, welche aus dem Projekttemplate kommen, sind mit dem Status "aus Projekttemplate" markiert und können des-

halb gelöscht werden. Eine Übernahme in das Projekt erfolgt durch die Abänderung des Status “geplant”, indem der Milestone oder die Aktivität editiert wird. Jeder Milestone enthält nochmals eine Detailansicht unter der die Aktivitäten des Milestones zu finden sind. Es können Notizen, Aktivitäten oder Milestones hinzugefügt werden, indem das “Hinzufügen” Symbol geklickt wird. Jeder Milestone, der vom Typ “Absatzplanungsmilestone” oder “Phase IV” ist, enthält geplante Verkäufe. Diese können durch Editieren des Milestones verändert werden.

Abbildung 7-15 Projektverwaltung - Details



Projekt verwalten

Projektnummer: P-199808141838
 Projektname INetDesign
 Projektbeschreibung Einrichtung von Multimediaarbeitsplätzen
Kunde: [Aloah Design](#)
Account Manager: Christian van der Leeden
Abteilung: Corporate Accounts

▼ Projektdetails

<input checked="" type="checkbox"/> Budget	350.000
<input checked="" type="checkbox"/> Apple Budget	250.000
<input checked="" type="checkbox"/> Status	offen
<input checked="" type="checkbox"/> Einstufung	C

▼ Kunde

[Aloah Design](#)

▼ Projektmitarbeiter

<input checked="" type="checkbox"/> Florian Mater	Ansprechpartner Einkauf	<input type="checkbox"/>
<input checked="" type="checkbox"/> Maria Eberle	Projektleiter Kunde	<input type="checkbox"/>
<input checked="" type="checkbox"/> Maria Schnell	Ansprechpartner EDV	<input type="checkbox"/>

▼ Interne Abteilungen

Corporate Accounts

▼ Projektmitarbeiter

<input checked="" type="checkbox"/> Christian van der Leeden	Accountmanager	<input type="checkbox"/>
--	----------------	--------------------------

▼ Keypartner

Lustig Computer

▼ Projektmitarbeiter

<input checked="" type="checkbox"/> Peter Lustig	Projektleiter Kunde	<input type="checkbox"/>
--	---------------------	--------------------------

▶ Partner

▶ Verkäufe für das Projekt

▶ Notizen für das Projekt 0

▶ Aktivitäten für das Projekt 2

▶ Milestones für das Projekt 6

In Abbildung 7–15 sind die Projektdetails aufgeklappt. Hierunter fallen die verschiedenen Projektpartner und -keypartner, sowie beteiligte interne Abteilungen und der Kunde. Für jeden Account können noch zusätzlich beteiligte Personen mit Rollen hinzugefügt werden. Falls diese Personen Benutzerrechte haben, erlangen sie die Berechtigungen, die mit dieser Rolle einhergehen. Das “Editieren” Symbol neben den Personen bezieht sich auf die Projektrolle. Es können auch Personen bzw. ganze Accounts inklusive der Personen aus dem Projekt gelöscht werden. Es sei darauf hingewiesen, daß das Löschen nur die Beteiligung am Projekt, nicht aber den Datensatz aus der Account- und Personendatenbank betrifft. Das “Hinzufügen” Symbol neben den Accounts und Personen bezieht sich auch auf das Hinzufügen von schon existierenden Accounts oder Personen, die durch eine Suche ausgewählt werden.

Abbildung 7–16

Projektverwaltung - Verkäufe



Projekt verwalten

Projektnummer: P-199808141838
 Projektname INetDesign
 Projektbeschreibung Einrichtung von Multimediaarbeitsplätzen
Kunde: [Aloah Design](#)
Account Manager: Christian van der Leeden
Abteilung: Corporate Accounts

Projektdetails
 Verkäufe für das Projekt

Wahrscheinlichkeit nach Sarimodell:		50 %	
Produktkategorie	Durchschnittspreis	Geplante Units	Verkaufte Units
Lowend	3.000 DM		
Monitore	4.000 DM	10	
Midrange	7.000 DM		
Highend	10.000 DM	20	
Powerbooks	7.000 DM		
Drucker	4.000 DM	10	
A	DM		
Summe Umsatz		280.000 DM	DM
Summe Units		40	

Notizen für das Projekt 0 
 Aktivitäten für das Projekt 2 
 Milestones für das Projekt 6 

Der Teil *Verkäufe* der Projektverwaltung ist in Abbildung 7–16 dargestellt. Die Ansicht enthält die Summe aller Verkäufe für das Projekt. Einzelne Verkäufe müssen in den zugehörigen Milestones editiert werden. Jedoch ist eine Veränderung der Wahrscheinlichkeit in der Entscheidungstabelle möglich. Durch anklicken der Prozentzahl der Wahrscheinlichkeit erscheint die Maske mit der Entscheidungstabelle (Abbildung 7–17).

Abbildung 7–17 Projektverwaltung - Entscheidungstabelle editieren



Während der Projektplanungsphase und auch später werden Milestones zum Projekt hinzugefügt, da entweder das benutzte Projekttemplate nicht vollständig auf das vorliegende Projekt paßt, beziehungsweise das Projekt ohne Projekttemplate angelegt wurde. Hierzu wird auf das “Hinzufügen” Symbol in der Detailanzeige für die Projektmilestones geklickt. Das Zieldatum und die Beschrei-

bung müssen eingegeben werden. Hinzu kommt der Milestonetyp. Bei den Milestonetypen “Phase IV” und “Absatzplanungsmilestone” werden automatisch die notwendigen Verkäufe generiert.

Abbildung 7–18 Projektverwaltung - Milestones hinzufügen

The screenshot shows a Netscape browser window titled "Netscape: Sheila - Projektinformationssystem". The main content area displays a form titled "Milestone eingeben". The form has the following fields and controls:

- Datum:** Four dropdown menus for "Tag" (set to "none"), "Monat" (set to "Sep"), "KW" (set to "none"), and "Jahr" (set to "1998").
- Beschreibung:** A text input field containing "Verkaufsmilestone" with vertical and horizontal scrollbars.
- Milestonetyp:** A dropdown menu set to "Phase I: Aquis".
- Status:** A dropdown menu set to "geplant".

At the bottom of the form are two buttons: "OK" and "Zurück". On the left side of the browser window, there is a sidebar menu with the following items:

- Neues Projekt eingeben
- Projekttemplat
- Projekte verwalten
- Projektverwaltung
- Milestone eingeben
- Alle Detailanzeigen öffnen

At the bottom of the browser window, there is a navigation bar with the following buttons: "Erinnerungen", "Projekte", "Accounts", "Optionen", "Projektvorschlag", "Reports", "Personen", "Produkte", and an Apple logo.

Nach der Eingabe oder nach dem Anklicken des “Editieren” Symbols neben dem zugehörigen Milestone kommt man zur Editiermaske (Abbildung 7–19). Hier können die Milestonedaten inklusive des Milestonetyps geändert werden. Bei der Änderung des Typs von “Phase IV” oder “Absatzplanungsmilestone” zu einem anderen Typ werden die Verkäufe automatisch gelöscht. Sie sind jedoch nicht verloren sondern werden erhalten. So können bei einer zufälligen Änderung des Milestonetyps die Verkaufsdaten nicht verlorengehen und werden bei einer weiteren Änderung in einen Milestonetyp mit Verkäufen wieder restauriert.

Zusätzlich zur Änderung der Basisdaten können hier auch Aktivitäten für den Milestone eingegeben werden. Außerdem ist die Dokumentation eines Milestones in Form von Notizen möglich. Weiterhin lassen sich die Verkäufe hier editieren, falls der Milestone Verkäufe enthält.

Abbildung 7–19 Projektverwaltung - Milestone editieren



Projekte wie Milestones können mit Aktivitäten versehen werden. Dies geschieht durch Anklicken des ‘‘Hinzufügen’’ Symbols in der Detailansicht der Aktivitäten. Die Eingabemaske ist in Abbildung 7–20 gezeigt. Es muß das Anfangs- sowie das Enddatum eingegeben werden. Weiterhin muß eine Beschreibung vorhanden sein, sowie der Aktivitätstyp gesetzt werden. Standardmäßig ist die Aktivität der Person zugeordnet, die sie eingibt. Sie ist damit für die Aktivität verantwortlich und erhält einen Eintrag in den Erinnerungen unter geplanten Aktivitäten. In der Editiermaske kann die Personenzuordnung jedoch geändert werden.

Abbildung 7-20 Projektverwaltung - Aktivität eingeben

The screenshot shows a Netscape browser window titled "Netscape: Sheila - Projektinformationssystem". The main content area displays a form titled "Aktivität eingeben". The form includes the following fields:

- Anfangsdatum:** Tag (1), Monat (Mar), KW (none), Jahr (1998)
- Enddatum:** Tag (13), Monat (Apr), KW (none), Jahr (1998)
- Beschreibung:** A text area containing "Produktvorführung beim Kunden".
- Aktivitätstyp:** A dropdown menu with "Sonstige" selected.
- Status:** A dropdown menu with "geplant" selected.

At the bottom of the form are "OK" and "Zurück" buttons. On the left side of the browser window, there is a navigation menu with the following items:

- Neues Projekt eingeben
- Projekttemplates
- Projekte verwalten
- Projektverwaltung
- Aktivität eingeben
- Alle Details anzeigen öffnen

At the bottom of the browser window, there is a navigation bar with the following buttons:

Vertriebs- informations- system	Erinnerungen	Projekte	Accounts	Optionen	
	Projektvorschlag	Reports	Personen	Produkte	

Das Editieren von Aktivitäten geschieht durch Anklicken des "Editieren" Symbol entweder aus dem Hauptfenster der Projektverwaltung oder aus dem Editierfenster eines Milestones. Wie Milestones können auch Aktivitäten durch Notizen dokumentiert werden.

Abbildung 7-21 Projektverwaltung - Notiz eingeben

Notiz eingeben

Datum 15. Aug 1998

Notiztext Projektdefinition des Kundens

Fileupload Icon

Alle Details anzeigen öffnen

Vertriebs- information- system	Erinnerungen	Projekte	Accounts	Optionen	
	Projektvorschlag	Reports	Personen	Produkte	

Alle Aktivitäten, Milestones und Projekte können durch Notizen dokumentiert werden. Einer Notiz wird automatisch das Erstellungsdatum und die eingebende Person gesetzt. Der Benutzer kann eine Beschreibung sowie eine Datei an die Notiz anhängen. Dateien können jeglicher Art sein und werden auf dem Server gespeichert, so daß sie für jedermann zugänglich sind.

Abbildung 7-22 Projektverwaltung - Projekt suchen



Der Einstieg in die Projektverwaltung beginnt immer mit der Suche des zu bearbeitenden Projekts (Abbildung 7-22), falls der Einstieg nicht über die Erinnerungen gewählt wird. Hierbei kann nach der Projektnummer, dem Projektnamen und dem Kundennamen gesucht werden. Projekte sind in vier Mengen unterteilt, über die gesucht werden kann. Einerseits existieren die eigenen Projekte, bei welchen der Benutzer eine Rolle spielt. Weiterhin gibt es die Abteilungsprojekte bei welchen ein Angestellter der Abteilung (oder Account) eine Rolle spielt. Die Unterabteilungsprojekte sind wie die Abteilungsprojekte definiert. Es werden jedoch alle Mitarbeiter der Unterabteilungen mit einbezogen. Zuletzt gibt es noch die Gesamtmenge aller Projekte, die durchsucht werden kann. Je nach Berechtigung entfallen hier Möglichkeiten der Suche. Die Selektion zur Bearbeitung eines Projekts geschieht durch Anklicken des "Selektieren" Symbols.

7.6 Accounts

Abbildung 7-23 Partner hinzufügen

The screenshot shows a Netscape browser window titled 'Netscape: Sheila - Projektinformationssystem'. The main content area displays the 'Partner hinzufügen' form. The form fields are as follows:

- Firmenname:** Input field containing 'CKS'.
- Niederlassung:** Input field containing 'Berlin'.
- Land:** Input field containing 'Deutschland'.
- Muttergesellschaft:** Text 'nicht eingegeben' with a magnifying glass icon to its right.
- Accounttyp:** Dropdown menu with 'Sonstige' selected.
- Firmenkürzel:** Input field containing 'CKS'.

Below the form is an 'OK' button. On the left side of the browser window, a navigation menu is visible with the following items:

- Partner hinzufügen
- Kunde hinzufügen
- Internen Account hinzufügen
- Account suchen
- Account eingeben
- Alle Detailanzeigen öffnen

At the bottom of the browser window, there is a navigation bar with the following buttons: Vertriebs-informationssystem, Erinnerungen, Projekte, Accounts, Optionen, Projektvorschlag, Reports, Personen, Produkte, and an Apple logo.

Die Accountverwaltung, welche über die Navigationsleiste erreichbar ist, enthält Menüpunkte für das Hinzufügen verschiedener Accounttypen, wie Partner, Kunden oder interne Abteilungen. Die Eingabemaske für Partner ist in Abbildung 7-23 gezeigt. Erforderlich ist hier nur der Firmenname. Die Niederlassung stellt den Ort der Zweigstelle des Betriebs dar. Durch das Klicken des Lupesymbols kann die Muttergesellschaft gesetzt werden. Hierbei kommt der Benutzer zuerst in die Accountsuche und danach erst in die Accounteingabe, falls die Muttergesellschaft nicht eingegeben ist. Als Accounttyp kann z.B. Partner, Systemintegrator, Handelspartner etc. gesetzt werden.

Abbildung 7–24 Account suchen

Kunden suchen

Firmenname

Niederlassung

Land

Firmenkürzel

Suchen

Partner Intern

Firmenname	Niederlassung	Kürzel	Land	
BMW				<input type="checkbox"/>
Aloah Design			Deutschland	<input type="checkbox"/>

Vertriebs-informations-system | Erinnerungen | Projekte | Accounts | Optionen | Projektvorschlag | Reports | Personen | Produkte |

Die Accountsuche ist für alle drei Accounttypen identisch (Abbildung 7–24). Durch Klicken auf den Button “Partner” bzw. “Intern” kann zwischen den Accounttypen gewechselt werden. Eine Sortierung der Resultate ist wie immer durch Klicken der Sortiersymbole in der Kopfzeile der Tabelle vorhanden. Die Accountsuche ist die einzige Möglichkeit Accounts zu editieren.

7.7 Personen

Abbildung 7–25 Person hinzufügen

The screenshot shows a Netscape browser window titled "Netscape: Sheila - Projektinformationssystem". The main content area displays a form titled "Person hinzufügen". The form fields are as follows:

- Vorname:** Peter
- Nachname:** Barlage
- Reports to:** nicht eingegeben
- Arbeitet bei:** Aloah Design
- hat Benutzerrechte:**
- Username:** Peter Barlage
- Passwort:** barlage

Below the main form is a table for role assignments:

Accountmanager	<input type="checkbox"/>
Promotor	<input checked="" type="checkbox"/>
Vertriebsleiter	<input type="checkbox"/>

At the bottom of the browser window, there is a navigation bar with the following buttons: Vertriebs-informations-system, Erinnerungen, Projekte, Accounts, Optionen, Projektvorschlag, Reports, Personen, and Produkte. An Apple logo is visible on the right side of the navigation bar.

In der Personenverwaltung, welche über die Navigationsleiste erreichbar ist, können Personen angelegt werden, Personen gesucht, Personen- und Projektrollen definiert sowie Rechte bearbeitet werden. In Abbildung 7–25 ist die Personeneingabe dargestellt. Die Zuordnung zu Accounts erfolgt über das Klicken des Lupensymbols neben der Eigenschaft "Arbeitet bei". Über die Accountsuche und die Selektion eines Accounts wird der Person ein Account zugeordnet.

Die Verwaltung der Personenrollen (Abbildung 7–26) ist identisch mit der Verwaltung der Projektrollen, bis auf die Einschränkung der letzteren auf die Kategorie "Eigene Projekte". Die gewählte Rolle steht im Popupmenü der Eingabemaske. Hier werden die Rechte der einzelnen Personen definiert, welche sie durch ihre Rollenzuordnung in der Personeneingabe bzw. beim Editieren der

Person zugewiesen bekommen. Die vier Projektkategorien sind schon aus der Projektsuche bekannt. Zusätzlich existiert der allgemeine Punkt "Administration", welcher sich auf alle Berechtigungen außerhalb der Projektverwaltung bezieht. Für jede der fünf Kategorien kann genau eine Sicherheitsstufe festgelegt werden. Die Stufen sind in der Rechteverwaltung genauer definiert. Die Definition von Personen- und Projektrollen wird am Anfang des Einsatzes des Vertriebsinformationssystems vorgenommen. Jeder Benutzer sollte beim Neuanlegen von Personen die existierenden Rollen verwenden.

Abbildung 7-26 Personenrollen definieren

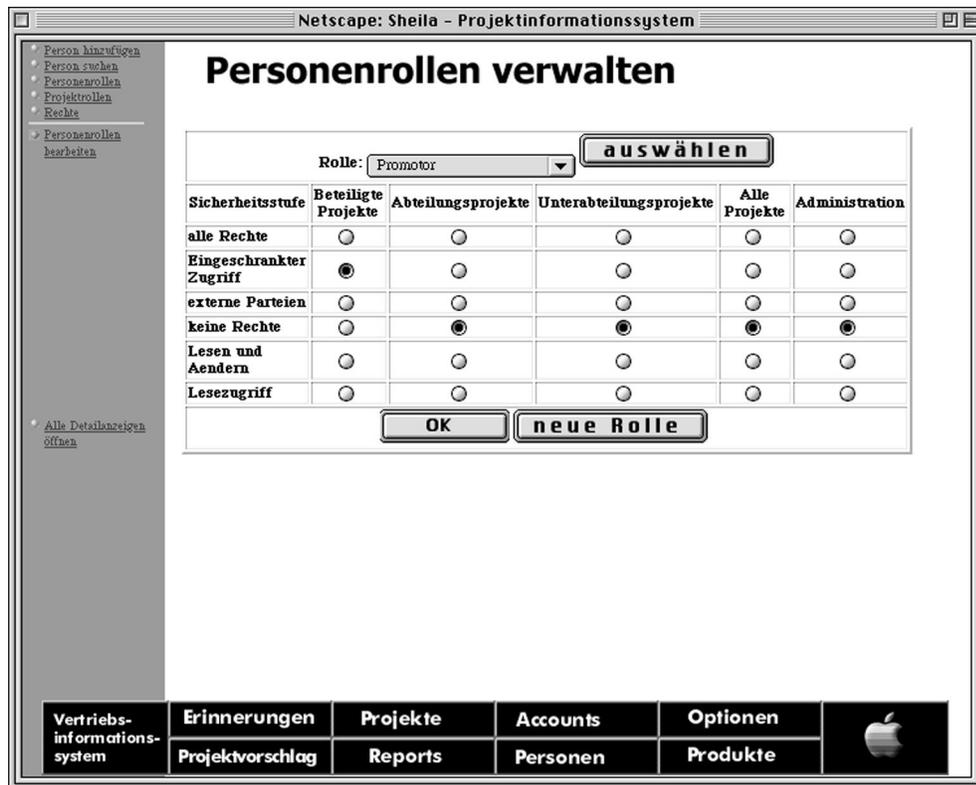
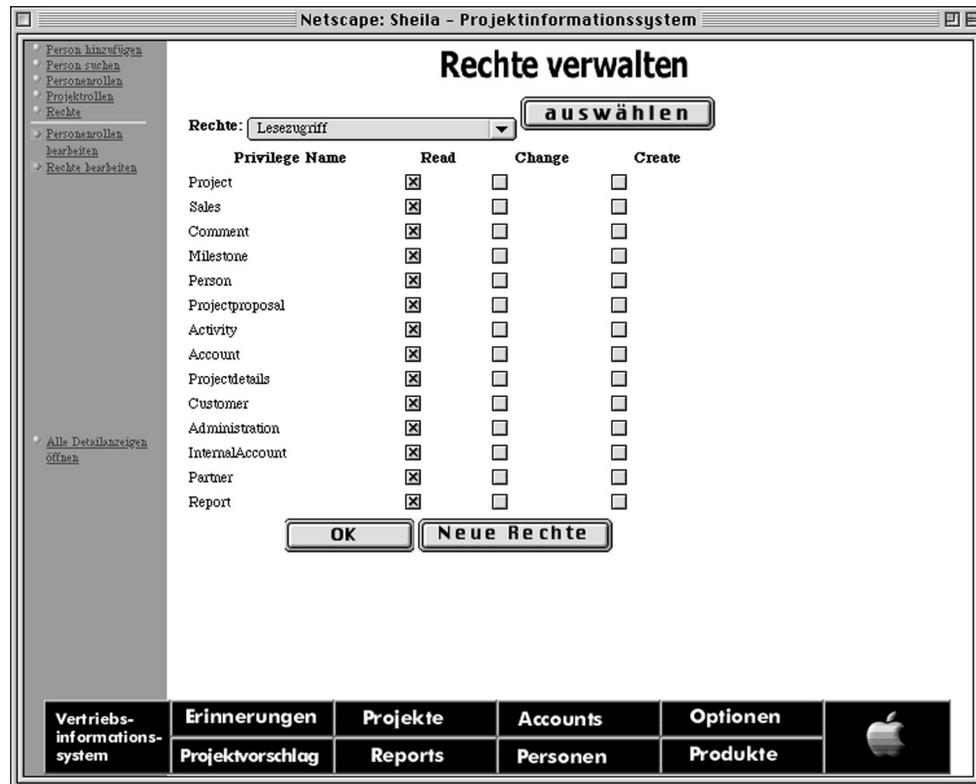


Abbildung 7-27 Rechte verwalten



In der Rechteverwaltung (Abbildung 7-27) werden die einzelnen Sicherheitsstufen genau definiert. Einzig die projektverwaltungsrelevanten Rechte (wie Aktivitäten, Milestones und Projektdetails) sind bei einer Zuordnung der Sicherheitsstufe zu einer Projektmenge ausschlaggebend. Der Rest wird nur für die Kategorie “Administration” verwandt.

Die Auswirkungen verschiedener Rechte beziehen sich auf die Deaktivierung von Menüeinträgen oder auch Detailansichten in der Projektverwaltung.

7.8 Reports

Abbildung 7–28 Reportgenerierung

Netscape: Sheila - Projektinformationssystem

Reporting nach Abteilung

Wählen Sie die Abteilungen aus, über deren Projekte ein Report erstellt werden soll.

Von: Monat: Jun, Jahr: 1998 Bis: Monat: Sep, Jahr: 1998

Reporttyp: Forecast

Corporate Accounts	<input checked="" type="checkbox"/>
Design und Publishing	<input type="checkbox"/>
Education	<input type="checkbox"/>
K12	<input type="checkbox"/>

Report

Alle Details anzeigen öffnen

Vertriebs- infor- mations- system	Erinnerungen	Projekte	Accounts	Optionen	Apple
	Projektvorschlag	Reports	Personen	Produkte	

Ein weiterer Teil des Vertriebsinformationssystems umfaßt das Reporting. Es wird über die Navigationsleiste erreicht. Hierbei kann der Benutzer zwischen Reports nach Accountmanagern auswählen bzw. nach Abteilungen. Für einen Report sind außerdem ein Startdatum und ein Enddatum relevant. Im Moment sind zwei Arten von Reports unterstützt. Es existieren einerseits der Trackingreport, welcher die Summation der getätigten Verkäufe in dem spezifizierten Zeitraum enthält und andererseits der Forecast, welcher im Gegensatz zum Trackingreport, die geplanten Verkäufe im Zeitraum subsummiert. Ein möglicher Report ist in Abbildung 7–29 dargestellt.

Abbildung 7-29 Projektreport

Netscape: Sheila - Projektinformationssystem

Projektreport

Zeitraum von: Jun 1998 bis: Sep 1998

Projektnummer	Kunde	A	LE	MO	D	MR	PB	HE	Category	Wahrscheinlichkeit
P-199808141838	Aloah Design	0	0	10	0	0	10	C		50 %
P-199808110334	BMW	0	1	2	0	30	0	2	C	30 %
Summe		0	1	2	10	30	0	12		

Legende:
A = A LE = Lowend MO = Monitore D = Drucker MR = Midrange PB = Powerbooks HE = Highend

Vertriebs-
informations-
system

Erinnerungen

Projektvorschlag

Projekte

Reports

Accounts

Personen

Optionen

Produkte



7.9 Produktkategorien

Abbildung 7-30 Produktkategorien verwalten

The screenshot shows a Netscape browser window titled 'Netscape: Sheila - Projektinformationssystem'. The main content area is titled 'Produktkategorien verwalten'. It features a table with three columns: 'Name', 'Kürzel', and 'Preis'. The table contains the following data:

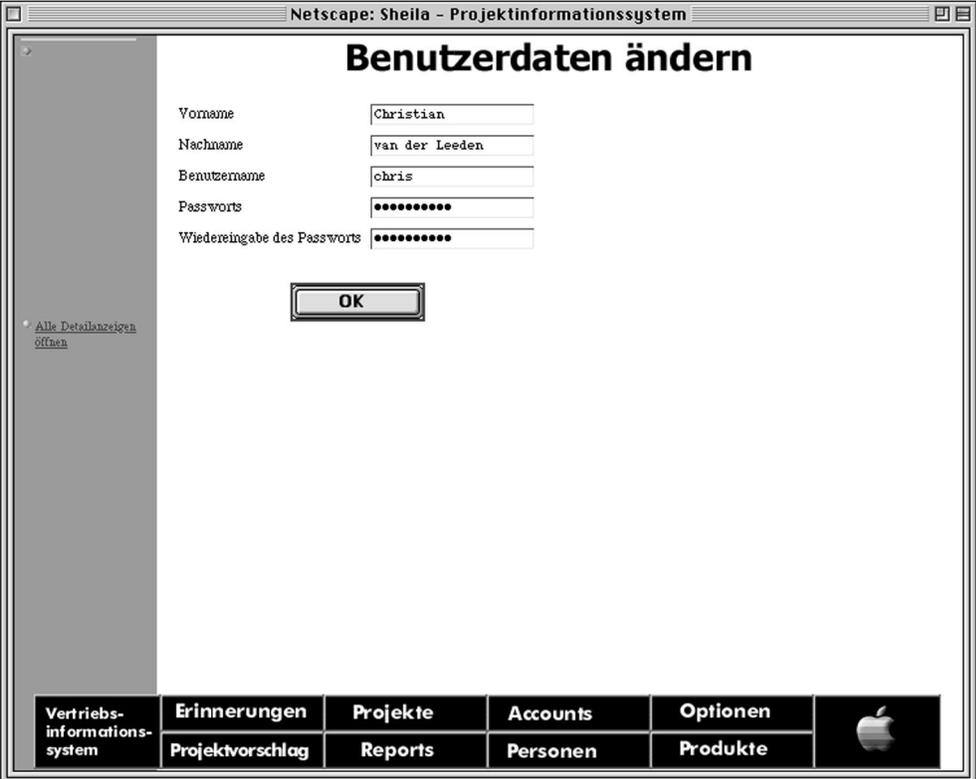
Name	Kürzel	Preis
Lowend	LE	3000
Monitore	MD	4000
Midrange	MR	7000
Highend	HE	10000
Powerbooks	PB	7000
Drucker	D	4000
A	A	

Below the table are two buttons: 'OK' and 'neues Produkt'. On the left side of the form, there is a link: 'Alle Detailkategorien öffnen'. At the bottom of the browser window, there is a navigation bar with the following items: 'Vertriebs-informations-system', 'Erinnerungen', 'Projekte', 'Accounts', 'Optionen', 'Projektvorschlag', 'Reports', 'Personen', 'Produkte', and an Apple logo.

Verkäufe werden immer in Produktkategorien aufgeteilt. Für die Verwaltung dieser Kategorien ist die Produktverwaltung vorhanden, welche über die Navigationsleiste erreichbar ist. Außer dem Anfügen von neuen Produktkategorien können auch existierende geändert werden. Das angegebene Kürzel der Kategorien wird vor allem im Reporting genutzt, damit die Bildschirmbreite nicht überschritten wird.

7.10 Optionen

Abbildung 7-31 Benutzerdaten ändern



The screenshot shows a Netscape browser window titled "Netscape: Sheila - Projektinformationssystem". The main content area displays a form titled "Benutzerdaten ändern". The form contains the following fields:

Vorname	<input type="text" value="Christian"/>
Nachname	<input type="text" value="van der Leeden"/>
Benutzername	<input type="text" value="chris"/>
Passworts	<input type="password" value="....."/>
Wiedereingabe des Passworts	<input type="password" value="....."/>

Below the form is an "OK" button. On the left side of the form area, there is a link: "Alle Detailanzeigen öffnen".

At the bottom of the browser window, there is a navigation bar with the following items:

Vertriebs- informations- system	Erinnerungen	Projekte	Accounts	Optionen	
	Projektvorschlag	Reports	Personen	Produkte	

Das System ist benutzerspezifisch ausgelegt. Unter dem Menüpunkt "Optionen" in der Navigationsleiste befindet sich die Editierung der eigenen Eigenschaft, wie Vorname, Nachname, Loginname und Paßwort. Eine Zuordnung zu anderen Personenrollen kann hier jedoch nicht durchgeführt werden.

8. Bewertung und Zusammenfassung

In dieser Diplomarbeit wurde ein Prototyp für ein Vertriebsinformationssystem geschaffen, das sich besonders an den Bedürfnissen der Großkundenabteilung von Apple orientiert. Mitarbeiter und Partner können das Fortschreiten eines Projekts verfolgen. Besonders hervorzuheben ist die Funktionlität der Erinnerung, die neben dem normalen Ablauf der Projektverwaltung einen sofortigen Überblick über Aktivitäten verschafft, welche die betreffende Person geplant hat, sowie alle Änderungen an Aktivitäten, Milestones und Projektdaten, die von anderen Personen durchgeführt wurden. Somit wird der tagtägliche Betrieb durch die Erinnerungen gesteuert, wobei die Projektplanung und Projekteingabe in der Projektverwaltung stattfindet. Zusätzliche Vorteile bietet die Standardisierung von Projektabläufen, den sogenannten Vertriebsprozessen, durch die Verwendung von Projekttemplates als Grundlage eines Projekts. Die möglichst einfache Bedienung soll den Mitarbeiter und Partner zur Nutzung des Systems animieren. Den Nutzen ziehen aber auch die Manager, in dem sie aus der eingegebenen Datenbasis sich Reports generieren können. Durch die Vergabe von Benutzerrechten und den Einsatz des World Wide Webs können auch externe Partner mit dem System arbeiten.

Das Vertriebsinformationssystem wurde an die Bedürfnisse von der Großkundenabteilung von Apple angepaßt, es ist jedoch auch in anderen Unternehmen einsetzbar. Die Grundbausteine, nämlich das projektbezogene Verkaufen von

Produkten, die Zusammenarbeit mit externen Projektteilnehmern und die Strukturierung von Projekten in Milestones und Aktivitäten findet sich in vielen Firmen wieder. Die speziellen Teile, wie den Projektvorschlag, die Entscheidungstabelle oder die Art der Reports, müssen an die Bedürfnisse von anderen Firmen angepaßt werden. Im Kern bleibt die Applikation jedoch gleich.

Schwierigkeiten traten vor allem während der Designphase auf. Hier gab es Probleme beim Verständnis der Methode OOHDM, da es sich um eine neue Methode handelt, die sich immer noch in der Entwicklung befindet. Der größte Nachteil war ein fehlendes Werkzeug zur Erstellung der Diagramme und die mangelnde Dokumentation. Die Methode ist nur in Veröffentlichungen beschrieben, jedoch ist daraus kein einheitliches Bild zu gewinnen, da in den verschiedenen Veröffentlichungen wiederum unterschiedliche Notationen verwendet wurden. Dadurch mußten die Diagramme öfters dem neusten Stand der Veröffentlichungen angepaßt werden. Während der Entwicklung des Abstract Interface Designs wurde offensichtlich, daß es nicht vollständig für die hier zu entwickelnde Applikation hergenommen werden konnte, da keine Symbole für die Eingabe von Daten bzw. für die Ausgabe von Listen vorhanden waren. So wurde die Notation des Abstract Interface Designs noch erweitert.

Für die Implementierung wurde WebObjects hergenommen. Hervorzuheben ist die Objektorientiertheit des Systems, welche sich vom Datenbankzugriff bis hin zu den HTML Anzeigenelementen erstreckt. Die einhergehende Komplexität erfordert jedoch eine längeren Lernprozeß als bei einfacheren Werkzeugen. Die großen Vorteile von WebObjects liegen im mehrschichtigen Datenbankframework, welches einen einfachen Datenbankzugriff erlaubt und in der Wiederverwendung von eigenen Anzeigenelementen.

Die verwendete Methodik zur Softwareentwicklung von der Erstellung der Anwendungsfälle anhand von Interviews, des nachfolgenden Designs mit OOHDM und die Implementierung hat maßgeblich zum Gelingen der Diplomarbeit beigetragen. Die Modellierung der Interviewergebnisse in Anwendungsfälle definierte ein klar umrissenes Einsatzgebiet der Applikation. Anhand dieser Benutzungsfälle konnte das Objektmodell im Conceptual Design entwor-

fen werden, welches später für das Datenbankdesign verwendet wurde. Ein gut durchdachtes Datenbankdesign ist eine der Grundlagen einer stabilen, leicht erweiterbaren Applikation. Das Navigational Design half bei der Verdeutlichung der Pfade, welche ein Benutzer wählt. Auf Basis der definierten Navigational Kontexte wurden die DisplayGroups in WebObjects konfiguriert. Das Verwenden einer Methode hat gegenüber einem adhoc Ansatz viel größere Chancen, eine stabile, den Nutzungswünschen des Kunden entsprechende und leicht erweiterbare Applikation hervorzubringen, so wie in diesem Fall. Weiterhin entstehen durch die angewandte Methodik eine Dokumentation der Applikation.

Zusammenfassend sind die durchlaufenen Schritte und verwendeten Techniken sehr geeignet für die Entwicklung einer Multimediaanwendung. Voraussetzung für den wirklichen Einsatz ist vor allem eine vollständige Dokumentation der Methode OOHDM und das Vorhandensein eines Werkzeuges, um die Diagramme zu erstellen. Solange dies nicht gegeben ist, wird die Verwendung der Methode aus dem in einem normalen Projekt immer vorhandenen Zeitdruck scheitern. Es wäre höchst wünschenswert ein solches Werkzeug zur Hand zu haben inklusive der Dokumentation.

Die vorgestellte Lösung stellt nur einen Prototypen dar. So ist die gesamte Antwortzeit der Applikation, von der Anfrage des Browsers bis hin zur fertigen Darstellung der Seite noch verbesserungsbedürftig. Sie liegt im Moment bei ca. 1s, wobei die Zeit für die Generierung der HTML Seite auf Seiten der Applikation gemessen mit dem Statistikmodul zwischen 0.1 und 0.5 Sekunden beträgt. Die Differenz wird für den Verbindungsaufbau und das Rendering der Seite gebraucht. Vor allem beim Rendering liegt noch Verbesserungspotential durch den verringerten Einsatz von geschachtelten HTML Tabellen. Weiterhin kann einige Funktionalität durch den Einsatz von JavaScript auf die Client-Seite verlegt werden, wie z.B. das Aufklappen der Detailansichten. Weiterhin sollte das Look-and-Feel mehr Macintosh-ähnlich werden. Ein weiterer Punkt ist die Integration des Systems hin zur Finanzabteilung, um die Umsätze und andere Kenn-

größen weiterzugeben. Eine Anbindung an existierende Personaldatenbanken bzw. Produktdatenbanken ist geplant. Es ist eine intensive Testphase mit weiteren Nachbesserungen in der Bedienerführung angesetzt.

A.Anhang

CRITERIA	10 POINTS	6 POINTS	4 POINTS	2 POINTS	SCORE
1. Strategic significance	Strategic decision concerning the account's company as a whole	Strategic decision concerning parts of the account's company	Project decision for the account	Not yet defined	
2. Complete solution	Fulfils the needs completely	Partly fulfils the account's needs	A special solution is being developed	Special solution is being sought after	
3. Special requirements as a disqualifying criterion	Only met by us	Could be met	We cannot meet all the requirements	We can only meet the requirements at great cost	
4. Competitors' solution (SW/HW/SI solution)	Worse	Similar	Better	Considerably better	
5. Important partners	Are in favour of us	Are neutral	Favour competitors as a rule	Are clearly competitor-oriented	
6. Mentor	Decision-makers and expert team favour us	Decision-makers or expert team favour us	A mentor is being persuaded to assume a pro-company attitude	No mentor exists	
7. Access to decision-makers	Decision-makers negotiate with us	Decision-maker within the expert team negotiates with us	The members of the expert team are not yet defined	The decision is made in another part of the company	
8. Project co-ordination	We control the project	Partner controls the project	No clear project control defined	Competitor controls the project	
9. Price	Within budget limits	Slightly above budget	Clearly above budget	Far above budget	
10. Comparison with competitors' price	Equal or lower	Slightly higher	More than 25% higher	More than 50% higher	
11. How big is the deal?	> \$ 2 million net	\$ 500 K to \$ 2 million net	\$ 100 K to \$ 500 K net	< \$ 100 K net	
12. Time frame for the decision	Decision will be made within this quarter	Decision will be made within next quarter	Decision will be made within next 12 months	Decision will be made later, i.e. ...	
⇒	⇒	⇒	⇒	TOTAL POINTS	

8.1 Beispiel einer Bewertungstabelle für Projekte

B.Literaturverzeichnis

-
- [1] G. Rossi, PhD Thesis Object-Oriented Hypermedia Design Method (in Portuguese) 1996
 - [2] D. Schwabe, G.Rossi, S. Barbosa, Systematic Hypermedia Application Design with OOHDM, Proceedings of the ACM International Conference on Hypertext (Hypertext '96), Washington, March 1996.
 - [3] D. Schwabe, Resumo de Notcao OOHDM, 1997
 - [4] D. Schwabe, R. de Almeida Pontes, OOHDM-Web: Prototyping of Hypermedia Applications in the WWW, 1998
 - [5] D. Schwabe; S. D. J. Barbosa, "Navigation Modeling of Hypermedia Applications, Technical Report MCC42/94, Departamento de Informatica, PUC-Rio, 1994 (ftp://ftp.info.puc-rio.br/pub/docs/techreports/94_42_barbosa.ps.gz)
 - [6] D. Schwabe, G. Rossi, Developing Hypermedia Applications using OOHDM, 1998
 - [7] D. Schwabe, G. Rossi, An Object Oriented Approach to Web-Based Application Design, 1998
 - [8] D. Schwabe, G. Rossi, S. D. J. Barbosa, Abstraction, Composition and Lay-Out Definition Mechanisms in OOHDM, November 1995, Proceedings of the ACM International Workshop on Effective Abstractions in Multimedia
 - [9] D. Schwabe, G. Rossi, The Object Oriented Hypermedia Design Model, Comm. of the ACM, Vol 38, #8, Aug 1995 (<http://irss.njit.edu:5080/cgi-bin/option.csh?sidebars/schwabe.html>)

- [10] N. Koch, Guide to the OOHDM Notation, 1998, internal Draft
- [11] O.M.F. De Troyer, C.J. Leune, WSDM: a user centered design method for Web-Sites, 1998, <http://www.elsevier.nl/cas/tree/store/-comnet/free/www7/1853/com1853.html>
- [12] Luis Olsina, Building a Web-based Information System applying the Hypermedia Flexible Process Modeling Strategy, 1st International Workshop on Hypermedia Development, 1998
- [13] NASA, Evaluation of Webapplication building software, 1996, <http://stellar.gsfc.nasa.gov/WebStudy/finalm~1.htm>
- [14] SalesLogix Feature Overview, 1998, <http://www.saleslogix.com/products/-saleslogix/details/features.phtml>
- [15] SalesLogix Company Profile, 1998, <http://www.saleslogix.com/company/press/profile.phtml>
- [16] How does SmartSales compare to other SFA?, 1998, SmartSales Inc., <http://www.smartsales.com/ssmdocs.exe>
- [17] Siebel Product Overview, 1998, <http://www.siebel.com/products/qualityteam-po.html>
- [18] Siebel Sales Team, 1998, <http://www.siebel.com/products/salesteam-po.html>

- [19] ACT! Evaluator's Guide, http://www.symantec.com/PressCenter/-reviews/rev_act4.0.zip, Februar 1998
- [20] ACT! Awards, <http://www.symantec.com/mobile/awards/act.html>, 1998
- [21] ACT! Kurzbeschreibung und Preisliste, <http://shop.symantec.com/act40-buy.html>
- [22] Technical Position for Archer Enterprise for ACT!, 1998, <http://www.archerent.com/Articles/ArcherESwp.zip>
- [23] Archer Enterprise for ACT! Overview, 1998, <http://www.archerent.com>
- [24] David Neumann, The Value-Add of Objective-C, 1998, http://www.wosource.com/value-add_of_objective-c.html
- [25] Justin Morgan, An Introduction to Objective-C, June 1997, <http://www.-byte.com/art/9706/sec4/art5.htm>
- [26] Bob Kaehms, Nicolas Popp, A Design Retrospect from One of WebObjects' Top Developers , July 1997, <http://www.webtechniques.com/features/1997/07/kaehms/kaehms.shtml>

- [27] OpenStep Object-Oriented Programming and the Objective C Language, 1995, Apple Computer
- [28] MNL, Getting Started With WebObjects, 1997, Apple Computer
- [29] MNL, WebObjects Developer's Guide, 1997, Apple Computer
- [30] MNL, EOF Developer's Guide for EOF 2.1 (Mach & Windows), 1996, Apple Computer
- [31] WebObjects 3.5 Enterprise Objects 2.2 Documentation, 1998, Apple Computer, <http://developer.apple.com/techpubs/enterprise/WebObjects/WebObjectsTOC.html>
- [32] Nancy Carighill, OpenStep for Enterprises, 1997, Wiley Computer Publishing
- [33] Grady Booch, Object Solutions, Managing the object-oriented project, 1996, Addison-Wesley
- [34] Unified Modeling Language Notation Guide version 1.1, August 1997, Rational Software, http://www.rational.com/uml/ad970805_UML11_Notation2.pdf
- [35] Unified Modeling Language Summary version 1.1, August 1997, Rational Software, http://www.rational.com/uml/ad970803_UML11_Summary.pdf
- [36] Unified Modeling Language Semantics version 1.1, August 1997, Rational Software http://www.rational.com/uml/ad970804_UML11_Semantics2.pdf
- [37] Unified Modeling Language Process-specific extensions version 1.1, August 1997, Rational Software, http://www.rational.com/uml/ad970807_UML11_Business.pdf
- [38] A.W. Scheer, Absatzprognosen, 1983 Springer
- [39] Charles Futrell, Sales Management, 1988, The Dryden Press
- [40] Lawrence B. Choko, Ben M. Enis, John F. Tunner, Jr, Managing Salespeople, 1992, Allyn and Bacon
- [41] Tony Alessandra, Phil Wexler, Rick Barrera, Non-manipulative Selling, 1987, Prentice-Hall
- [42] Rolph E. Anderson, Joseph F. Hair, Jr, Alan J. Bush, Professional Sales Management, 1988, McGraw-Hill
- [43] Klaus E. Goehrmann, Verkaufsmanagement 1984, Kohlhammer
- [44] Ian Graham, Object Oriented Methods, 1993, Addison-Wesley
- [45] Rumbaugh, M. Blaha, W. Premerlani, F. Eddy and W. Lorensen, Object Oriented Modeling and Design, 1991, Prentice Hall