

Projekt-Zeiterfassung

Referenznummer: 23.15

Abgabedatum: 22. April 1999

Autor: Pascal Müller
 Buchenweg 2
 2560 Nidau
 032 / 331 2273

Sulzer Microelectronics AG
 Mattenstrasse 6a
 2555 Brugg
 032 / 374 3227

Klasse: B23

Betreuer: Stefan Leuenberger
 Kirschbaumweg 13
 2563 Ipsach
 032 / 331 7312

Sulzer Microelectronics AG
 Mattenstrasse 6a
 2555 Brugg
 032 / 374 3234

Experte: Stephan Fischli
 Software-Schule Schweiz

Abstract: Dieses Dokument beschreibt ein System zur Erfassung von (projektspezifischer) Arbeitszeit auf Intranetbasis. Das Eingabe-Frontend ist plattformunabhängig und die Daten werden in einer zentralen Datenbank abgelegt. Aus den Daten können aussagekräftige Auswertungen erzeugt werden.

Keywords: Zeiterfassung, Intranet, Java, JDBC, Datenbank, RDBMS, SQL

Nidau, den 11. Dezember 1998

Der Experte

Historie:

Datum	Version	Autor	Bemerkungen
1. 12. 1998	0.5	pm	Initialdokument, Formatdefinitionen
6. 12. 1998	0.9	pm	Dokument komplett
8. 12. 1998	1.0	pm	Ergänzungen nach dem Review mit dem Betreuer / Anwender - Stefan Leuenberger <stefanl@sme.ch> (Betreuer, Leiter Engineering) - Heidi Küng <heidik@sme.ch> (Buchhaltung)
11. 12. 1998	1.1	pm	Anpassungen nach Durchsicht durch den Experten Stephan Fischli
11. 12. 1998	1.2	pm	Zeitplan, Inhaltsverzeichnis

1 • Allgemeines

1.1 Zweck

Dieses Dokument beschreibt die Anforderungen an eine Projekt-Zeiterfassungs-Anwendung und definiert *WAS* gemacht werden soll. Es dient als Basis deren Realisierung und ist Bestandteil der Diplomarbeit 23.15.

1.2 Leserkreis

Das Dokument richtet sich an den Betreuer als Anwender und an den Experten als Bewertungsgrundlage. Ferner sollte sich auch ein anderer Anwender mit Hilfe dieses Pflichtenheftes ein Bild machen können und es soll ihm helfen, ggf. weitere Ideen und Wünsche zu formulieren.

1.3 Anwendungsgebiet

Die Projekt-Zeiterfassung soll in einem System Design Center (Ingenieurbüro, welches elektronische Systeme unter massiver Zuhilfenahme mikroelektronischer Integrationsstechnologien entwickelt) eingesetzt werden. Es soll ein manuelles, papierbasiertes Erfassen (siehe Abbildung in Kapitel 2.1.3 (Ist-Zustand) auf Seite 6) der an Projekten geleisteten Ingenieurarbeiten ablösen und ein automatisches Auswerten ermöglichen.

In diesem Design Center arbeiten 3 Firmen zusammen. Es wird auch mit externen Leuten zusammengearbeitet und es existiert eine Zweigniederlassung in Fremont / USA. Diese werden jedoch nicht mit diesem System erfasst.

1.4 Definitionen und Abkürzungen

1.4.1 Änderungen

Änderungen im Dokument seit der letzten Version werden am linken Rand mit einem schwarzen Balken gekennzeichnet (hier zur Veranschaulichung vorhanden).

1.4.2 Abkürzungen

JDBC	Java Database Connectivity
JDK	Java Development Kit
JFC	Java Foundation Classes (Java GUI Komponenten, siehe [3])
JRE	Java Runtime Environment
RDBMS	Relational Database Management System
URL	Uniform Resource Locator

1.4.3 Begriffe

Da diese Anwendung zwei verschiedenen Benutzergruppen genügen muss, werden in diesem Zusammenhang folgende Begriffe verwendet:

Anwender	sind alle, die in irgendeiner Form mit dem Zeiterfassungssystem in Berührung kommen (um z.B. Reports abfragen)
Entwickler	sind Anwender, welche das System aktiv nutzen, also ihre Zeiten erfassen. Dies sind die Leute aus der Entwicklungsabteilung (Engineering)

1.5 Referenzen

Literaturverweise werden im Text mit [<Nummer>] gekennzeichnet. Die Nummer (#) bezieht sich auf folgende Tabelle:

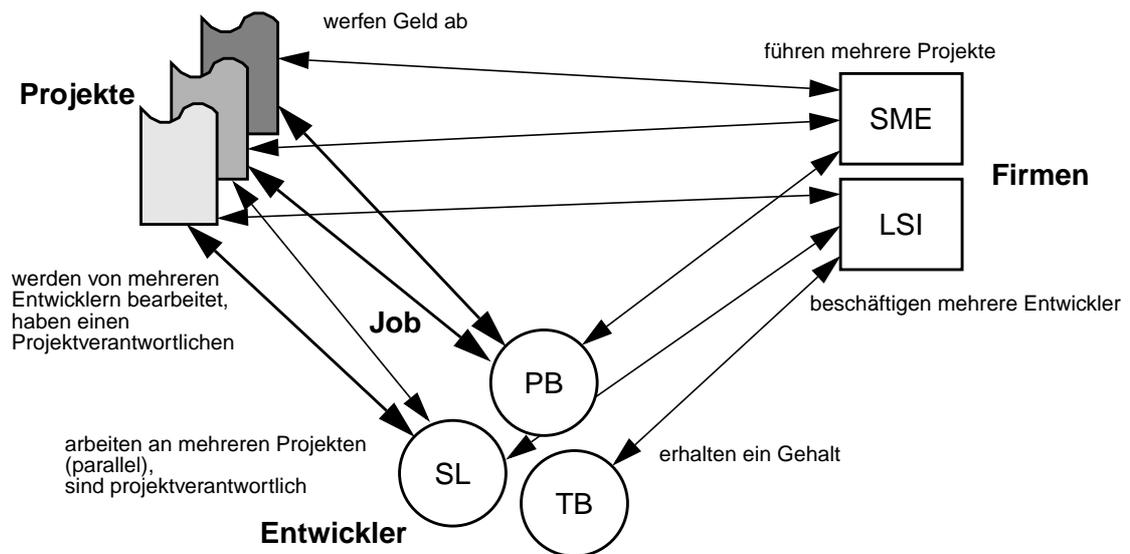
#	Titel	Autor	Bemerkungen / URL
1	Hilfsmittel zur Erstellung von technischen Pflichtenheften	Othmar Bürgi Info Top AG	Version 27 (1998)
2	Java White Papers	Sun Microsystems	http://java.sun.com/docs/white/
3	JFC White Paper	Sun Microsystems	http://java.sun.com/marketing/collateral/foundation_classes.html
4	Java Plugin Software	Sun Microsystems	http://java.sun.com/products/plugin/
5	JDBC API Documentation	Sun Microsystems	http://java.sun.com/products/jdbc/
6	PostgreSQL Website	PostgreSQL Organization	http://www.postgresql.org/
7	Java Code Conventions	Sun Microsystems	http://java.sun.com/docs/codeconv/html/CodeConventionsTOC.doc.html
8	Draft Java Coding Standard	Doug Lea	http://gee.cs.oswego.edu/dl/html/javaCodingStd.html
9	Netscapes's Software Coding Standards Guide for Java	Christie Badeaux	http://developer.netscape.com/docs/technote/java/codestyle.html
10	100% Pure Java CookBook	Sun Microsystems	http://java.sun.com/100percent/cookbook.html
11	JDK Software Compatibility	Sun Microsystems	http://java.sun.com/products/jdk/1.2/compatibility.html
12	JDK 1.2 Deprecated API list	Sun Microsystems	http://java.sun.com/products/jdk/1.2/docs/api/deprecated-list.html
13	CGI - Common Gateway Interface	NCSA, University of Illinois	http://hoohoo.ncsa.uiuc.edu/cgi/
14	Java Servlet API	Sun Microsystems	http://java.sun.com:8081/products/servlet/index.html

2 • Allgemeine Beschreibung

2.1 Übersicht

2.1.1 Situation

Da im Design Center an komplexen Systemen gearbeitet wird, sind mehrere Entwickler mehrerer Firmen längere Zeit in ein Projekt involviert. Die Projekt-Zeiterfassung soll helfen, die Übersicht zu bewahren. Folgende Grafik soll kurz aufzeigen, wie Firmen, Projekte und Entwickler zusammenwirken:



Dazu ist zu vermerken, dass ein Entwickler üblicherweise über längere Zeit einige wenige grosse Projekte betreut und daneben für weitere Projekte sporadisch Abklärungen oder kleine Arbeiten macht. Ferner hat jedes Projekt einen Projektverantwortlichen (hier durch einen etwas dickeren Pfeil symbolisiert).

2.1.2 Grössenordnung

Die aktuelle Applikationsumgebung umfasst:

- 3 Firmen
- 14 Entwickler, Tendenz steigend (ca. 30% pro Jahr)
- ca. 10 aktive Kundenprojekte, mit der Anzahl Entwickler zunehmend
- über 100 abgeschlossene Projekte (diese werden nicht in dieses neue Zeiterfassungssystem übernommen)

2.1.3 Ist-Zustand

Die an Projekten gearbeiteten Zeiten werden von jedem Entwickler von Hand erfasst. Hierzu wird folgendes Formular (ein Blatt pro Entwickler und Woche) ausgefüllt:

TIME SHEET FOR	PASCAL MÜLLER		Week 49 (30. November - 6. December) 1998							Vision:
ACTIVITY	PROJECT	CUSTOMER	MON	TUE	WED	THU	FRI	SAT	SUN	TOTAL
.... DESIGN / SUPPORT										
.... DESIGN / SUPPORT										
.... DESIGN / SUPPORT										
.... DESIGN / SUPPORT										
.... DESIGN / SUPPORT										
.... DESIGN / SUPPORT										
.... DESIGN / SUPPORT										
.... DESIGN / SUPPORT										
.... DESIGN / SUPPORT										
CUSTOMER TRAINING										
ENGINEERING SERVICES										
LABOR										
SCHULUNG VORBEREITUNG										
SYSTEM ADMINISTRATION			8 1/2	8 1/2	8	9				34
BUG HANDLING										
CONFERENCE / FAIR										
INTERNAL MEETING				1/2	1/2					1
LS TOOL SUPPORT										
PRE-SALES SUPPORT										
3RD PARTY SUPPORT										
MANAGEMENT / ADMINISTRATION			1		1/2	1/2				2
PERSONAL TRAINING										
MILITARY DAY										
HOLIDAY / PUBLIC HOLIDAY										
.... TRAVEL / SICKNESS										
TOTAL			9 1/2	9	9	9 1/2				37

Die hier erfassten Zeiten werden zwecks Auswertung (von Hand) in ein Excel-Sheet übertragen. Dies ist sehr fehleranfällig, da die von den Entwicklern verwendeten Projektbezeichnungen nicht konsistent sind (interne Projektbezeichnung, Kundenprojektbezeichnung, Projektname). Eine Nachkalkulation konnte deshalb nur sehr mühsam durchgeführt werden.

2.1.4 Soll-Zustand

Die Entwickler geben die an den verschiedenen Projekten gearbeitete Zeit täglich online an Ihrem Arbeitsplatz ein. Diese Zeiten werden in einer zentralen Datenbank abgelegt. Aus den so erfassten Daten können automatisch Auswertungen und Reports erstellt werden. Projekte können von privilegierten Personen eröffnet und mit eindeutigen Namen versehen. Die Entwickler können aus diesen Projekten auswählen und Ihre Zeiten zuweisen.

Als Basis soll das bisherige Zeiterfassungsblatt (Siehe Abbildung Kapitel 2.1.3 (Ist-Zustand) auf Seite 6) dienen. Entsprechend sollten auch "unproduktive" Arbeiten (untere Hälfte des Zeiterfassungsblattes) erfasst werden können. Dies ist erforderlich, um sicherstellen zu können, dass alle Beteiligten ihre Projektarbeit wirklich eintragen.

2.1.5 Umfeld

Die Entwickler arbeiten hauptsächlich auf UNIX-Maschinen (Sun Solaris), vereinzelt aber auch auf Wintel PCs. Die Administration arbeitet ausschliesslich auf Wintel-PCs. Alle Maschinen befinden sich in einem Netzwerk. Alle Rechner im Netzwerk sind mit dem Netscape Communicator 4.5 ausgerüstet und haben Zugriff auf den Intranet-Webserver (Apache auf Solaris).

2.2 Beziehung zu anderen Systemen

Es existiert ein in der Einführung befindliches Finanzbuchhaltungs-System. Es soll in einer späteren Phase möglich sein, diesem System Daten zur Weiterverarbeitung zu übergeben (nicht Bestandteil dieser Diplomarbeit).

2.3 Hardware

Wie in Kapitel 2.1.5 (Umfeld) auf Seite 6 beschrieben, sind die Arbeitsplätze mit UNIX- oder Wintel-Maschinen ausgerüstet. Damit Die Applikation auf allen verfügbaren Computern ausgeführt werden kann, soll Java [2] zum Einsatz kommen (plattformunabhängig).

Als Plattform für die zentrale Datenbank stehen verschiedene bestehende Solaris-Server zur Verfügung, welche diese Aufgabe (zumindest in der Anfangsphase) übernehmen können. Aus Kapitel 2.1.2 (Größenordnung) auf Seite 5 kann abgeleitet werden, dass die Belastung des Datenbankservers nicht sehr gross sein wird.

2.4 Software

Grundsätzlich soll die Verwendung von möglichst 'offener' und freier Software angestrebt werden, damit eine gewisse Herstellerunabhängigkeit erreicht werden kann.

2.4.1 Laufzeitumgebung

Die Applikation (Java Frontend) soll auf Java 1.1 basieren und kann JFC (Swing) benutzen. In einem ersten Ansatz soll die Applikation als Applet im netzwerkweit vorhandenen Netscape Communicator 4.5 laufen. Dies ermöglicht das System mit minimalstem Installationsaufwand einzuführen.

Sollten sich Java-Inkompatibilitäten mit dem verwendeten Browser zeigen, kann auf Sun Microsystems "Java Plugin Software" [4] zurückgegriffen werden, welche solche Probleme nicht kennen sollte.

Von Java 1.2 wird aus folgenden Gründen (noch) nicht Gebrauch gemacht:

- JDK 1.2 wurde erst gerade offiziell freigegeben (am 8. Dezember 1998 in New York)
- JDK 1.2 wird von den verfügbaren Browsern (noch) nicht unterstützt

Es soll aber bei der Implementation darauf geachtet werden, dass in einem späteren Zeitpunkt die Migration auf Java 1.2 möglichst einfach durchgeführt werden kann (siehe hierzu [11] und [12]).

Als Datenbank soll eine relationale SQL-Datenbank (RDBMS) zum Einsatz kommen. Es wird die Freeware-Datenbank PostgreSQL [6] vorgeschlagen, da sie auch das nötige JDBC-Interface unterstützt.

2.4.2 Entwicklungsumgebung

Entsprechend der Laufzeitumgebung kommt JDK 1.1 (in einer aktuellen Version, zur Zeit 1.1.7B (reference implementation) / 1.1.6 (production release)) zum Einsatz. Dem Autor steht ausserdem Java Workshop 2.0 von Sun Microsystems zur Verfügung.

2.4.3 Programmierrichtlinien

Da der Arbeitgeber des Autors keine eigene Software entwickelt und deshalb nicht auf bestehende Programmierrichtlinien zurückgegriffen werden kann, wird auf [7] verwiesen (ergänzend kann auch [8] und [9] konsultiert werden). Um die Plattformunabhängigkeit zu erfüllen sollte auch [10] berücksichtigt werden.

2.5 Benutzereigenschaften

2.5.1 Eingabe

Die Anwender ("Entwickler", siehe Kapitel 1.4.3 (Begriffe) auf Seite 3) der Projektzeiterfassungs-Applikation sind ausschliesslich hochqualifizierte Ingenieure mit Erfah-

rung in der Bedienung komplexer Software-Tools und hohem technischen Verständnis. Da die Zeiterfassung eher etwas unbeliebtes ist, soll die Applikation dies nicht unnötig erschweren.

2.5.2 Ausgabe

Die an den Resultaten interessierten Personen (in der Administration und Buchhaltung) weisen mindestens PC- und WWW-Kenntnisse auf.

3 • Technische Anforderungen

3.1 Allgemeines

3.1.1 Installation

Die Installation des Java Frontends (Java Applet) beschränkt sich darauf, eine HTML-Startseite und die Java Classfiles in die Dateistruktur der Intranetdaten einzufügen und in anderen Seiten Links darauf zu setzen.

Zur Installation der Datenbank wird auf die Dokumentation des entsprechenden Produkts [6] verwiesen. Für die Datenstruktur-Initialisierung der Datenbank soll ein Script erstellt werden.

Aufgrund der Sicherheitsrestriktionen von Java Applets ist zu beachten, dass der Datenbank-Server auf derselben Maschine (selbe IP-Adresse) laufen muss wie der Webserver, welche die Applet-Classfiles geliefert hat. Diese Bedingung kann in der Zielinstallation erfüllt werden. Alternativ liesse sich auf der Webserver-Maschine ein Datenbank-Gateway installieren.

3.1.2 Sprache

Die Applikation soll aus folgenden Gründen eine englischsprachige Benutzerführung aufweisen:

- Es werden verschiedensprachige Anwender damit arbeiten
- Die bisherige Arbeitsumgebung ist (fast) ausschliesslich in Englisch gehalten
- Englisch ist in der Mikroelektronikbranche üblich und kann von den Anwendern vorausgesetzt werden
- Es existiert eine Zweigniederlassung in den USA (diese ist jedoch nicht involviert)

3.1.3 Eingabe

Alle Eingaben sollen über ein GUI (Java Frontend) erfolgen. Eingaben werden ausschliesslich durch die Entwickler vorgenommen.

3.1.4 Ausgaben

Alle Ausgaben sollten als Reports oder Listen in HTML erzeugt werden und über das Intranet zugänglich sein. Einsicht sollten alle Mitarbeiter der drei Firmen haben, also auch die Leute aus der Buchhaltung, dem Verkauf und der Administration.

3.1.5 Unterhalt

Die Datenbank soll kompakt gehalten werden und nicht mehr benötigte Detailinformationen sollten nicht unnötig darin verbleiben. Dies soll erreicht werden, indem beendete Projekte abgeschlossen werden können, d.h. es wird aus der Datenbank eine Zusammenfassung erzeugt und die Detaildaten danach entfernt. Für nichtproduktive Projekte kann dies eine Art Jahresabschluss sein.

3.2 Detaillierte Funktionsbeschreibung

Die folgend in den Tabellen angegebenen Prioritäten (P) bedeuten:

- 1: Muss für ein sinnvolles Arbeiten vorhanden sein, zwingend
- 2: Sollte wenn möglich realisiert werden, ist aber nicht zwingend
- 3: Optionale Zusatzfunktionen ("Nice to have")

Die Funktionen lassen sich in 3 Gruppen einteilen:

3.2.1 Erfassen

Dies sind die Funktionen, mit welchen der Entwickler in Berührung kommt und welche durch das Java Frontend abgedeckt werden.

Funktion	P	Beschreibung
User selektieren	1	Damit die einzugebende Zeit einem Entwickler zugeordnet werden kann (Selektion aus einer Liste). Hier ist darauf zu achten, dass Verwechslungen vermieden werden (Auswahl über den ausgeschriebenen Namen anstelle eines Usernamens).
Projekt selektieren ⁽¹⁾	1	Damit die einzugebende Zeit einem Projekt zugeordnet werden kann (Selektion aus einer Liste vorgegebener Projekte, siehe auch Kapitel 4.2.3 (Projekt) auf Seite 12).
Tag selektieren ⁽¹⁾	1	Damit die einzugebende Zeit einem Tag der aktuellen Woche zugeordnet werden kann.
Zeit erfassen	1	Erfassen der Zeit, der Hauptgrund der Applikation. Es soll für die aktive Woche pro Entwickler / pro Projekt / pro Tag ein Wert möglich sein.
Woche abschliessen	1	Eine Woche muss abgeschlossen werden, um die nächste in Angriff nehmen zu können. Dies ist eine bewusste Einschränkung um den Entwickler zum regelmässigen Ausfüllen zu zwingen. Eine abgeschlossene Woche soll angeschaut, nicht mehr aber verändert werden können (dies sollte nur durch eine privilegierte Person möglich sein, siehe Kapitel 3.2.3 (Administration) auf Seite 11). Bei Abschluss der Woche sollte eine Gesamtarbeitszeitkontrolle stattfinden, d.h. die gesamte Arbeitszeit (inklusive Ferien, Krankheit, usw.) der aktuellen Woche darf 80% der nominalen Wochenarbeitszeit nicht unterschreiten. Ist diese Bedingung nicht erfüllt, kann die Woche nicht abgeschlossen werden und es soll eine entsprechende Meldung ausgegeben werden.
Projekt in persönliche Liste ⁽²⁾ aufnehmen	2	Es soll möglich sein, die "Hauptprojekte" so darzustellen, dass die Daten sofort (und ohne Projektselektierung) eingegeben werden können.
Projekt aus persönlicher Liste entfernen	2	Das entsprechende Gegenstück.

1. Eventuell in einem Arbeitsgang (persönliche Liste)
2. Eine persönliche konfigurierbare Liste sollte ähnlich einfach auszufüllen sein wie das Papier-Zeitblatt (siehe Bild in Kapitel 2.1.3 (Ist-Zustand) auf Seite 6).

3.2.2 Auswerten

Diese Funktionen erzeugen die gewünschten Ausgaben. Die Reports sollten im HTML-Format erzeugt werden, damit diese mit einem Web-Browser betrachtet werden können.

Funktion	P	Beschreibung
Gesamtaufwand auf einem Projekt	1	Angabe des gesamten (internen) Entwicklungsaufwandes [CHF] eines spezifischen Projekts. Dies soll dabei nach den Firmen aufgeschlüsselt und als Gesamtsumme erscheinen. Wird für die Nachkalkulation verwendet. Auch der Projektverantwortliche und die Projekt-Informationen sollten ersichtlich sein. Dieses Resultat ist bei Designabschluss von grossem Interesse.
Arbeitszeit der Entwickler	1	Durch hinzufügen von "allgemeinen Arbeiten" wie Ausbildung, Administration, Ferien, Meeting usw. (Siehe auch das Zeitblatt in Kapitel 2.1.3 (Ist-Zustand) auf Seite 6) lässt sich die Arbeitszeit erfassen. Dies dient zur Kontrolle ob jeder Entwickler seine Daten wirklich eingegeben hat. Ausgabe: 1 Wert pro Woche und Entwickler.
Status-Report	1	Listet auf, wer welche (vergangenen) Wochen abgeschlossen hat und wer nicht.
Arbeitsprofil des Engineering	2	Ermitteln der Produktivität der gesamten Entwicklungsabteilung. Darzustellen sind pro Entwickler / Total: Aufwand aller (produktiven) Projekte (1 Zahl), Aufwand jedes unproduktiven Projektes (z.B. 12 Zahlen), eine "Effizienzkennziffer" (Verhältnis von produktiver zu totaler Arbeitszeit) in %. Diese statistischen Werte erlauben eine bessere Zeitabschätzung bei der Ressourcenplanung.
Budget überprüfen	2	Es soll überprüft werden, ob das dem Projekt zugewiesene (interne) Budget nicht schon aufgebraucht ist und ggf. deutlich darauf aufmerksam machen. Die Darstellung soll wie folgt aufgebaut sein: Aufwand in Stunden (und Kosten [CHF]) pro Woche (Zeilen) und pro Entwickler (Spalten), Total Stunden (und Kosten [CHF]) pro Entwickler über alle Wochen (Totalzeile) und Projekt-Gesamtstunden (und Kosten [CHF]) (Gesamttotal).

Funktion	P	Beschreibung
Wochenauslastung eines Entwicklers	3	Anzahl bearbeiteter Projekte und daran gearbeitete Stunden eines Entwicklers (siehe Abbildung Kapitel 2.1.3 (Ist-Zustand) auf Seite 6, letzte Spalte).
Bezogene Ferienstunden	3	Pro Mitarbeiter im laufenden Jahr bezogene Ferien [Std].

3.2.3 Administration

Diese Funktionen sind nur für einen privilegierten Anwender (Administrator) über ein Passwort zugänglich.

Funktion	P	Beschreibung
Authentifizierung	1	Zugriffsberechtigung nur durch ein Passwort.
Passwort ändern	1	Zugangspasswort ändern.
Werte korrigieren	1	Möglichkeit, einen falschen Wert in einer bereits abgeschlossenen Woche eines beliebigen Projekts / Entwicklers zu ändern (braucht kein Luxus-Interface). (Diese Funktion kann als erstes implementiert und als Datenbankzugriffs-Test benutzt werden).
Neues Projekt eröffnen	1	Eröffnen eines neuen Projektes und setzen von Namen, Bezeichnungen, Budget usw. (siehe Kapitel 4.2.3 (Projekt) auf Seite 12).
Projekt löschen	1	Projekt entfernen. Löscht das Projekt und alle dazugehörenden Jobs.
Projekt ändern	1	Um z.B. ein Projekt umzubenennen, da z.T. zu Projektbeginn (Abklärungsphase) nicht alle Angaben vorliegen.
Mitarbeiter erzeugen	1	Erstellen eines Mitarbeiters, damit dieser seine Zeiten erfassen kann.
Mitarbeiter löschen	1	Löschen eines Mitarbeiters. Dies sollte nur möglich sein, wenn alle Projekte, an denen er beteiligt war, abgeschlossen sind.
Mitarbeiter ändern	2	Da Namen (bei Heirat), Telefonnummer, Email, Ansatz ändern können. Die Firma darf NICHT geändert werden, da sonst bereits verbuchte Arbeiten fälschlicherweise der neuen Firma belastet werden. ⁽¹⁾
Mitarbeiter (de)aktivieren	2	Sperrt den Mitarbeiter für die Eingabe (Siehe auch Kapitel 4.2.2 (Mitarbeiter) auf Seite 12).
Firma erzeugen ⁽²⁾	3	Eine neue Firma erzeugen (sehr selten benutzte Funktion).
Firma löschen	3	Eine Firma löschen (sehr selten benutzte Funktion, sollte nur möglich sein, wenn diese keine laufenden Projekte und Mitarbeiter mehr hat)

1. Um trotzdem einen Firmenwechsel innerhalb der Firmengruppe abbilden zu können, kann ein neuer (gleichnamiger) Mitarbeiter mit demselben Namen und der neuen Firmenzugehörigkeit erzeugt und der ursprüngliche Eintrag deaktiviert (siehe "Mitarbeiter (de)aktivieren") werden.
2. Bei der Einführung des Systems werden die Firmen direkt auf der Datenbank mit SQL-Befehlen erzeugt. Deshalb kann das System sehr gut ohne diese Funktion (und dessen Gegenstück) betrieben werden.

4 • Datenbasis

4.1 Zugriff

Für den Zugriff auf die Datenbank soll in einer ersten Phase kein Passwort erforderlich sein. Um trotzdem einen minimalen Schutz gegenüber nicht-Entwickler zu erreichen, kann die Intranet-Zugriffsseite mit einem Passwort gesichert werden (Authentifizierung durch den Webserver).

4.2 Struktur

Basierend auf der Aufstellung in Kapitel 2.1.1 (Situation) auf Seite 5 gibt es 4 Objekttypen. Diese und deren Informationsgehalt sind folgend aufgelistet:

4.2.1 Firma

Eine Firma ist erforderlich, da Projekte und Mitarbeiter dieser zugeordnet werden müssen um den Aufwand nach Firma aufschlüsseln zu können.

Name	Typ	Beschreibung
Name	String	Firmenname.
LogoURL	String	URL auf das Firmenlogo (Bitmap), kann zur besseren Visualisierung der Firmenzugehörigkeit verwendet werden.

4.2.2 Mitarbeiter

Name	Typ	Beschreibung
Name	String	Voller Name des Mitarbeiters.
Kürzel ⁽¹⁾	String	Das firmenintern verwendete Kürzel (2-3 Zeichen, z.B. "PM").
Email ⁽²⁾	String	Email-Adresse.
Telefon ⁽²⁾	String	Interne Telefonnummer.
Firma	Ref	Referenz auf seine Arbeitgeberfirma.
Aktiv	Boolean	Ermöglicht, einen Entwickler als passiv zu kennzeichnen. Dies ist notwendig, da Mitarbeiter, die in einer laufenden Projektphase die Projektarbeit beenden, bis zum Projektabschluss weitergeführt werden müssen. Mit diesem Flag können diese für die Eingabe gesperrt werden.
Ansatz	Float [CHF]	Interner Stundenansatz (ermöglicht eine Aufwandrechnung eines Projektes).
Sollzeit	Float [Std]	Vertraglich vereinbarte Wochenarbeitszeit (z.B. 40 Std). Dies ermöglicht das Überprüfen, ob eine Woche komplett erfasst wurde (siehe Kapitel 3.2.1 (Erfassen) auf Seite 10).

1. Für die Darstellung von kompakten Listen
2. Rein informative Angabe

Bei der Verwendung des Stundenansatzes ist zu bemerken, dass es möglich ist, aus den generierten Reports darauf zurückzuschliessen. Sollte dies mit der Firmenphilosophie kollidieren, kann dieser Wert einfach auf 1 gesetzt werden.

4.2.3 Projekt

Es gibt verschiedene Typen von Projekten: Produktive (Kundenprojekte) und nicht produktive (administrative Arbeiten, usw.). Bei den produktiven Projekten soll noch zwi-

schen Abklärung (Auftragshoffnung) und Design (Auftrag erhalten) unterschieden werden.

Name	Typ	Beschreibung
Code	String	Projektcode, z.B. L5A8260, TH4011.1C Dieser wird z.T. von den Chipherstellern zugeteilt
Name ⁽¹⁾	String	Projektname, z.B. PLIO Dieser Name wird üblicherweise vom Kunden benutzt (und auch von ihm festgelegt)
Kunde ⁽¹⁾	String	Name des Kunden, für welchen das Projekt realisiert wird (Auftraggeber) Ist dies ein nicht produktives Projekt, ist der Wert NULL
Typ	Int	Typ des Projekts ("Abklärung" oder "Design")
Budget	Float [CHF]	Budgetierter Aufwand für die Entwicklung (ermöglicht eine Aufwandüberschreitung zu erkennen)
Firma	Ref	Referenz auf die Firma (Auftragnehmer) Ist dies ein nicht produktives Projekt, ist der Wert NULL
Betreuer	Ref	Referenz auf den Projektverantwortlichen Ist dies ein nicht produktives Projekt, ist der Wert NULL
Bemerkung	String	Ermöglicht, einem Projekt erweiterte Zusatzinformationen mitzugeben (maximal 100 Zeichen).

1. Diese Angaben erleichtern die eindeutige Identifizierung des Projektes und sollten beim Selektieren sichtbar sein

Es ist sicherzustellen, dass die Felder jederzeit geändert werden können. Dies ist nötig, da in der Abklärungsphase kein definitiver Code vorliegt. Dieser wird erst für die Realisierung festgelegt (definitive Bestellung des Kunden). Trotzdem soll eine Vorabklärung möglich sein. Die dabei erfassten Zeiten sollen dann auch später im laufenden Projekt zur Verfügung stehen.

Ein typischer Lebenszyklus eines Projekts läuft etwa so ab:

- Ein Projekt wird zwecks Abklärung (z.B. Machbarkeitsstudie) eröffnet und als produktives Design vom Typ "Abklärung" definiert. Nicht bekannte Felder werden mit Namen versehen (z.B. Code mit "Abkl_ECT")
- Die Entwickler arbeiten an der Abklärung und verbuchen die aufgewendete Zeit auf diesem Projekt. Diese Zeiten werden in den Jobs im Feld "Zeit Abklärung" festgehalten.
- Der Kunde hat bestellt, ein Projektcode wurde zugeteilt. Entsprechend wird jetzt das Projekt angepasst (umbenannt) und auf den Typ "Design" geändert.
- Die Entwickler arbeiten am Projekt und verbuchen die aufgewendete Zeit auf diesem Projekt. Diese Zeiten werden in den Jobs im Feld "Zeit Design" festgehalten.
- Ist das Projekt abgeschlossen, wird ein Report erzeugt und das Projekt kann gelöscht werden.

4.2.4 Job

Hier werden die aufgewendeten Stunden pro Projekt/Mitarbeiter/Tag festgehalten. Es soll gemäss Kapitel 4.2.3 (Projekt) auf Seite 12) zwischen Abklärung und Design unterschieden werden.

Name	Typ	Beschreibung
Projekt	Ref	Referenz auf das bearbeitete Projekt
Mitarbeiter	Ref	Referenz auf den Mitarbeiter
Tag	Date	Datum
Zeit Abklärung	Float [Std]	Geleistete Arbeit in Stunden für Abklärungen
Zeit Design	Float [Std]	Geleistete Arbeit in Stunden für effektive Entwicklungsarbeit

5 • Externe Schnittstellen

5.1 Benutzeroberfläche

Das Java Frontend soll ein GUI aufweisen, welches auf Javas AWT (ev. Swing) aufbaut. Es werden 2 Funktionsgruppen unterschieden:

5.1.1 Eingabe

Dem Entwickler soll eine Wochenübersicht mit "seinen" Projekten präsentiert werden, (vergleichbar mit dem Zeitblatt in Kapitel 2.1.3 (Ist-Zustand) auf Seite 6), in welcher er seine Zeiten eintragen kann. Die Bedienung sollte der eines Spreadsheets entsprechen.

Das Portfolio soll dem Mitarbeiter ermöglichen, die von ihm bearbeiteten Projekte festzufalten, damit diese für spätere weitere Eingaben sofort zur Verfügung stehen. Hierzu stehen ihm die Funktionen "Projekt in persönliche Liste aufnehmen" und "Projekt aus persönlicher Liste löschen" (siehe Kapitel 3.2.1 (Erfassen) auf Seite 10) zur Verfügung.

Ist seine Woche ausgefüllt, kann er diese abschliessen und erhält damit Zugriff auf die folgende Woche. Dabei ist sicherzustellen, dass eine Woche nicht unbeabsichtigt abgeschlossen werden kann, da diese danach nicht mehr änderbar ist.

5.1.2 Administrator

Die in Kapitel 3.2.3 (Administration) auf Seite 11 beschriebenen Funktionen stehen dem Anwender nach der Authentifizierung (Passwort) zur Verfügung. Dies kann mittels einfacher Maske realisiert werden. Es ist zu gewährleisten, dass der verwendete Projekt-Code eindeutig ist, um das Projekt richtig identifizieren zu können.

Für die Editor zur Korrektur falscher Eingaben in schon abgeschlossenen Wochen reicht eine sehr einfache Eingabemaske aus.

5.2 Softwareschnittstellen

5.2.1 Datenbank

Als Schnittstelle zur Datenbank kommt JDBC [5] zur Anwendung. Dieses API ist Bestandteil der Java-Umgebung und wird von PostgreSQL unterstützt.

5.2.2 Webserver

Die Anbindung an den Webserver zwecks dynamischer Generierung von Reports erfolgt über das CGI [13], mittels Servlet [14] oder ähnlich.

6 • Zeitplan

Das Projekt sollte einen Umfang von 300 Stunden (38 Tage) aufweisen, wovon die Erstellung des Pflichtenheftes mit ca. 6 Tagen zu veranschlagen ist.

Folgend eine Zeitabschätzung für die Realisierung der beschriebenen Projekt-Zeiterfassung.

• Total	30 Tage
6.1 Design	
• Total	15 Tage
6.1.1 <u>Architekturüberlegungen</u>	
• Total	3 Tage
• Zusammenspiel Browser / Applet / RDBMS / Webserver / Generatoren	
• Evaluation Webserver-Einbindung (CGI, Servlet, ...)	
6.1.2 <u>Datenstruktur</u>	
• Total	2 Tage
6.1.3 <u>Applet</u>	
• Total	5 Tage
• GUI	
• Logik	
• Ablauf	
6.1.4 <u>Reportgeneratoren</u>	
• Total	5 Tage
• Design allgemeine (wiederverwendbare) Teile	
• Einzelne Generatoren	
6.2 Implementation	
• Total	8 Tage
6.2.1 <u>Aufsetzen der Datenbank</u>	
• Total	1 Tag
• Installation	
• Script Datenstrukturdefinition	
6.2.2 <u>Implementation Applet</u>	
• Total	4 Tage
• Administratoren-Teil	
• Eingabeteil	
• Webseiten	
6.2.3 <u>Implementation Reportgeneratoren</u>	
• Total	3 Tage
• Webserver konfigurieren (Anbindung)	
• Generatoren	
6.3 Test	
• Total	4 Tage
• Testdaten erzeugen	
• Eingabe	
• Administrator	
• Reports / Abschlüsse	

6.4 Sonstiges

- Total
- Optische Gestaltung der Reports (HTML)
- Reserve

3 Tage

6.4.1 Dokumentation

Die Dokumentation soll parallel zur Arbeit geschrieben werden. Ein Benutzerhandbuch ist nicht vorgesehen, eine Kurzübersicht auf dem Intranet (HTML) wäre jedoch sicher hilfreich.