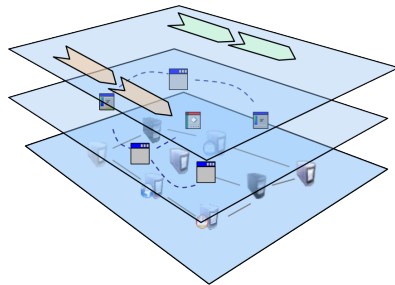


Projektmanagement

2. September 2008
Projektgruppe Modulares EAM System

David Heimann, Roland Koppe



Carl von Ossietzky Universität Oldenburg
Fakultät II - Department für Informatik
Abteilung Informationssysteme - Prof. Dr. Dr. h.c. H.-J. Appelrath

OFFIS - Institut für Informatik
Bereich Betriebliches Informationsmanagement

Inhaltsverzeichnis

1	Einleitung	3
2	Projektziele	3
3	Vorgehen	3
4	Zeitplanung	5
5	Aufgabenverteilung	7
6	Dokumentation	7
7	Dokumente	8
7.1	Projektmanagement und Projektdokumentation	8
7.2	Anforderungsdefinition	8
7.3	Entwurf des Systems	8
7.4	Quellcode und Klassenbeschreibungen	8
7.5	Handbuch	9
7.6	Schnittstellendokumentation	9
7.7	Festlegung eines Änderungsverfahrens	9
8	detaillierte Anforderungen	9
8.1	svn	9
8.2	Protokolle	10
8.3	Internetseite	11
8.4	Konventionen	11

1 Einleitung

Zur erfolgreichen Durchführung eines Projektes benötigt dieses ein Projektmanagement, welches den Planungsprozess umfasst, der notwendig ist, um ein Projekt erfolgreich abzuschließen. Deswegen werden im Folgenden die Projektziele des Projektmanagements näher erläutert. Anschließend wird das Vorgehen in der Entwicklung näher beschrieben, wobei insbesondere auf das verwendete Vorgehensmodell eingegangen wird.

Weiterhin werden die Aufgaben- und Zeitplanung näher beschrieben, was beinhaltet, dass die Entwicklung des Prototypens in verschiedene Arbeitspakete aufgeteilt wird.

Zur Protokollierung von wichtigen Ereignissen, beispielsweise das Erfüllen von Meilensteinen, wird ein Projekttagebuch geführt, welches in schriftlicher Form bei Roland nachzusehen und nicht digitalisiert ist.

2 Projektziele

Folgende Ziele wurden von uns zu Beginn des Projektes formuliert:

- Projektteilnehmer sollen Erfahrungen im Programmierbereich und Softskills sammeln
- Qualität des Produktes entspricht den Erwartungen des Kunden
- Erfahrung im Projektmanagement
- Einhaltung der Terminplanung
- Auseinandersetzung mit der EAM-Thematik sowie verwendeten Technologien
- Das Projektteam soll als solches auftreten und zusammenwachsen
- Motivation der Mitglieder im Projektteam
- Erhaltung einer guten Stimmung innerhalb des Teams
- Möglichst ausgewogene Lastenverteilung unter den Projektteilnehmern

3 Vorgehen

Dieser Abschnitt soll einen Eindruck des in diesem Projekt verwendeten Vorgehensmodells liefern.

Im ersten Abschnitt des Projekts wurden die Anforderungen des Kunden an das zu entwickelnde EAM-System analysiert, spezifiziert und festgelegt. Die Ergebnisse dieses Schrittes finden sich in der Anforderungsdefinition.

Erst nach Abschluss der Anforderungsdefinition ging das Projekt in die nächste Phase über. Dadurch sollte sichergestellt werden, dass die in die Anforderungsdefinition

aufgenommenen Anforderungen vollständig und korrekt waren, bevor weitere Schritte eingeleitet wurden.

Die nächste Phase des Projekts zeichnete sich durch eine „agile“ Vorgehensweise aus. Agil meint in diesem Zusammenhang die positiv wirkende Vermischung von Entwurf und Implementierung. So startete der Entwurf mit der Konzipierung und Definition von Grundbausteinen, die durch die Analyse der Anforderungen bereits bekannt waren. Auf der Basis dieses Grundentwurfs wurde die Architektur des Systems, also die verschiedenen Schichten und Schnittstellen, definiert.

Der so entstandene Entwurf wurde dann durch die Implementierung vervollständigt. Mögliche Probleme oder Schwierigkeiten während der Implementierung wirkten somit wieder zurück auf den Entwurf, wodurch flexiblere und problemadequate Lösungen entwickelt werden konnten.

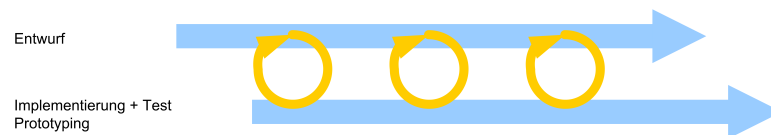


Abbildung 1: Zusammenwirken Entwurf Implementierung

Die Phase der Implementierung wurde durch die Entwicklung und Fertigung von Prototypen erweitert. Diese Prototypen stellten den aktuellen Stand der Entwicklung dar. Die Präsentation, der so entstehenden Prototypen an den Kunden, halfen dem Entwicklungsteam beim Verständnis der korrekten Umsetzung. Weiterhin erhielt der Kunde einen Einblick in den aktuellen Stand der Entwicklung und konnte gegebenenfalls weitere Wünsche äußern. Die Frage, ob diese Wünsche des Kunden während des aktuellen Projekts umgesetzt werden konnte oder ein Folgeprojekt einzuplanen war, erforderte eine Analyse des zu erwartenden Aufwands und einen Abgleich mit der Anforderungsdefinition.

Die Implementierung schloss mit der Ausgabe eines „final release“ ab. Dieses Release stellte also den programmiertechnischen Teil des zu entwickelnden Produkts dar.

Während der gesamten Laufzeit des Projekts wurden verschiedene Dokumente für den Projektbericht gesammelt. Diese Dokumente, welche neben der Anforderungsdefinition, dem Entwurf und dem Handbuch auch die Dokumente des Projektmanagements enthalten, gehören dem Produkt an.

Weiterhin wurden im gesamten Projektverlauf mindestens einmal wöchentliche Treffen wahrgenommen. Diese Treffen dienten zur Diskussion des aktuellen Stands sowie dem Feedback zu einzelnen Aufgabenpaketen. Außerdem hatten diese Treffen den Zweck, den Kunden aktuelle Ergebnisse zu präsentieren.

4 Zeitplanung

Eine Übersicht der Termine die im Projekt auftraten, liefert Abbildung 2 sowie die grafische Übersicht in Abbildung 3.






























Nr.	Vorgangsname	Dauer	Anfang	Ende	Vorgänger
1	 Vorab Prototyp	13 Tage	Fr 09.11.07	Mi 28.11.07	
2	 Datenmodell, Implementierung	13 Tage	Fr 09.11.07	Di 27.11.07	
3	 Vorstellung	0 Tage	Mi 28.11.07	Mi 28.11.07	
4					
5	Anforderungsdefinition	40 Tage	Mi 21.11.07	Mi 30.01.08	
6	 Aufteilung, Einleitung, Themen	11 Tage	Mi 21.11.07	Mi 05.12.07	
7	 Use Cases, Ausformulierung, Rechtschreibung, PM, Doku	11 Tage	Do 06.12.07	Do 20.12.07	6
8	 Fertigstellung Teil 1	5 Tage	Mo 17.12.07	Fr 21.12.07	
9	 Fertigstellung Teil 2	6 Tage	Mo 07.01.08	Mo 14.01.08	8
10	 Abgabe Anforderungsdefinition	0 Tage	Di 15.01.08	Di 15.01.08	9
11	 Überarbeitung Anforderungsdefinition	10 Tage	Mi 16.01.08	Di 29.01.08	10
12	 Abgabe Anforderungsdefinition, überarbeitet	0 Tage	Mi 30.01.08	Mi 30.01.08	11
13					
14	Dokumentation	200 Tage	Mi 07.11.07	Mi 03.09.08	
15	 Modulkonzept/Modularisierung, Datenmodell und Datenbank, Rollenkonzept detailliert ausarbeiten	15 Tage	Do 31.01.08	Mi 20.02.08	
16	 Architektur: Ebenen und Schnittstellen definieren	5 Tage	Do 07.02.08	Mi 13.02.08	
17	 Festlegung von Aufgabenpaketen zur Implementierung	0 Tage	Mi 13.02.08	Mi 13.02.08	16
18	 Abgabe und Vorstellung vorläufiger Entwurf	0 Tage	Mi 20.02.08	Mi 20.02.08	17
19	 Erweiterung Entwurf, Kreislauf mit Implementierung	122 Tage	Mi 20.02.08	Fr 15.08.08	18
20	 Fertigstellung Entwurf	13 Tage	Mo 18.08.08	Mi 03.09.08	19
21	 Dokumentation (Handbuch, Klassenbeschreibung, etc.)	23 Tage	Mo 04.08.08	Mi 03.09.08	
22	 Projekttagbuch	200 Tage	Mi 07.11.07	Mi 03.09.08	
23					
24	Implementierung	123 Tage	Do 21.02.08	Mi 20.08.08	
25	 Implementierung	121 Tage	Do 21.02.08	Fr 15.08.08	
26	 Tests und Entwicklung	121 Tage	Do 21.02.08	Fr 15.08.08	
27	 Prototyp 1, verikal, Grundfunktionalität	0 Tage	Mi 26.03.08	Mi 26.03.08	
28	 Prototyp 2, verikal, erweiterte Funktionalität	0 Tage	Mi 23.04.08	Mi 23.04.08	
29	 Vorstellung Prototyp	0 Tage	Mi 30.04.08	Mi 30.04.08	
30	 pre Alpha	0 Tage	Mi 28.05.08	Mi 28.05.08	
31	 Alpha	0 Tage	Mi 25.06.08	Mi 25.06.08	
32	 Beta	0 Tage	Mi 30.07.08	Mi 30.07.08	
33	 Final Release	0 Tage	Fr 15.08.08	Fr 15.08.08	
34	 Final Release Präsentation	0 Tage	Mi 20.08.08	Mi 20.08.08	
35					
36	 Abgabe Projekt	0 Tage	Mi 03.09.08	Mi 03.09.08	
37	 Abschlusspräsentation	1 Tag	Mo 06.10.08	Mo 06.10.08	

Abbildung 2: Übersichte über Termine im Projekt

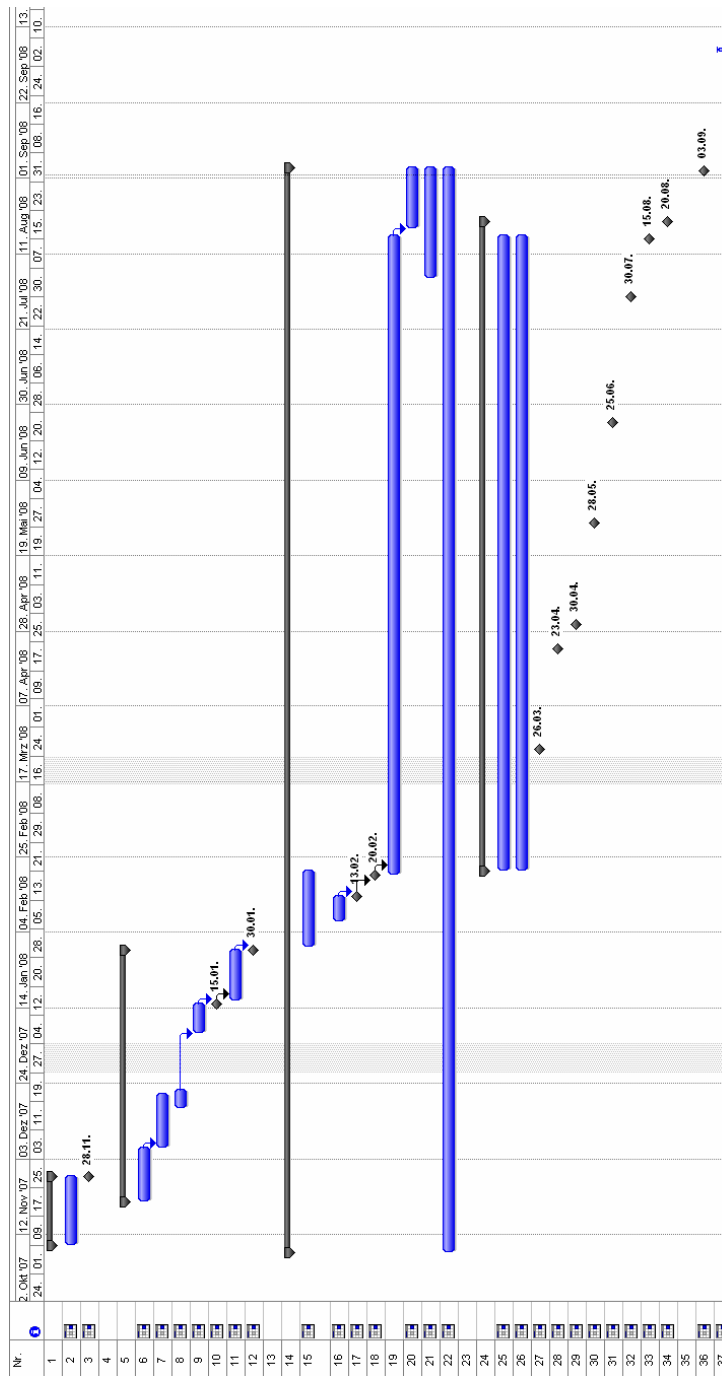


Abbildung 3: Übersicht über Projektzeitpunkte

5 Aufgabenverteilung

Die Aufgabenbereiche wurden, wie in der folgenden Tabelle ersichtlich, auf einzelne Teilnehmer verteilt. In der Tabelle 1 ist zu sehen, wer für welchen Aufgabenbereich bzw. Modul zuständig war.

Aufgabenbereich	Name
Projektmanagement	David, Roland
Core	Roland, Christian Z., Christian R., Jens, Jörn, David, Yu, Mart
BundleManager	Yu, Jens, Christian Z.
DataManager	David, Bin
UserManagement	Christian R., Igor, Jens
Erweiterte Datenerfassung	Christian Z.
Export	Mart
Analyse	Mart
Visualisierung I	Philipp
Visualisierung II	Philipp
Visualisierung III	Philipp
QueryBrowser	Roland

Tabelle 1: Aufgabenbereiche

6 Dokumentation

Die Dokumentation des Projektes ist eine wichtige Aufgabe, deren Ergebnis zu dem zu entwickelnden Produkt gehört. Grundsätzlich gilt für alle Dokumente die Einhaltung des einheitlichen Layouts. Dieses Layout kann dem bereits zur Verfügung stehenden Dokuments „Anforderungsdefinition“ entnommen werden.

7 Dokumente

kurz	Bezeichnung
PM	Dokumentation des Projektmanagements
AFD	Anforderungsdefinition
ENT	Entwurf des Systems
IMPL	Quellcode und JavaDoc
TEST	Testdokumentation und Testfälle
HAND	Handbuch zur Installation und Bedienung
SCHN	Schnittstellendokumentation

Tabelle 2: Übersicht der Dokumente

7.1 Projektmanagement und Projektdokumentation

Die Dokumentation des Projektmanagements umfasst einen Überblick des verwendeten Vorgehensmodells für das Projekt, welche in diesem Dokument zur Verfügung steht. Außerdem wurden die Projektplanung, Zeitplanung und die Aufgabenplanung dargestellt und durch die Projektleitung bei Bedarf laufend aktualisiert. Neben diesen Punkten werden die konkreten Anforderungen an die Dokumentation des Projekts benannt.

7.2 Anforderungsdefinition

Bei der Anforderungsdefinition handelte es sich um das Dokument zur Spezifikation der Anforderungen an das zu entwickelnde Softwaresystem. Neben den funktionalen und nicht-funktionalen Anforderungen enthält dieses Dokument auch die Anforderungen zur Dokumentation, Anwendungsoberfläche, Anwendungsfällen und eine Systemabgrenzung.

7.3 Entwurf des Systems

Der Entwurf ist ein eigenständiges Dokument, welches die Entwurfsentscheidungen und Technologien zur Umsetzung des Softwaresystems enthält. Weiterhin gibt der Entwurf einen Einblick in die Datenstrukturen und Architektur des Systems.

7.4 Quellcode und Klassenbeschreibungen

Zur Dokumentation zählen auch der Quellcode mit entsprechenden Kommentaren und Beschreibungen, die hier in Form von JavaDoc vorgenommen werden und in einem Dokument Klassenbeschreibungen zusammengefasst werden.

7.5 Handbuch

Das Handbuch des Systems unterteilt sich in zwei Bereiche. Zum einen in die Installationsanleitung des Softwaresystems und zum anderen in die Bedienungsanleitung des Systems. Dabei werden Unterscheidungen zwischen dem administrativen und normalen Benutzer vorgenommen, da letzterer kein Wissen über die Administrationsbereich des Systems benötigt beziehungsweise besitzen darf.

7.6 Schnittstellendokumentation

Der Dokumentation gehört bei dem zu entwickelnden modularen EAM-System auch eine Dokumentation der Schnittstellen des Kernsystems an, welche Teil des Entwurfs ist. Diese Schnittstellen können und werden von Modulen genutzt, daher müssen diese Schnittstellen definiert werden.

Diese Dokumentation der Schnittstellen soll es damit Entwicklern erlauben, Module für das EAM-Tool geeignet zu entwickeln und einzusetzen. Entwickler von Modulen benötigen nicht zwingend ein Wissen über die tatsächliche Umsetzung des Kernsystems, Entwurfentscheidungen werden somit durch dieses Dokument gekapselt. So besteht bspw. die Möglichkeit die Entwicklung von Modulen an außenstehende Entwickler auf einfache Weise zu vergeben.

Dazu macht dieses Dokument die Schnittstellen bekannt und beschreibt diese textuell. Erweitert wird diese Beschreibung durch das Bereitstellen einer beispielhaften Implementierung, die das leichtere beziehungsweise schnelle Verständnis der Schnittstellen fördern soll.

7.7 Festlegung eines Änderungsverfahrens

Als Änderungsverfahren wurde der kontinuierliche Änderungsprozess ausgewählt. Dies bedeutete, dass Änderungen in der Anforderungsdefinition oder dem Entwurf laufend in den Entwicklungsprozess einfließen. Vor allem im Hinblick auf die agile Implementierung erwies sich dieses Verfahren als das geeignetste.

8 detaillierte Anforderungen

Im Folgenden werden die detaillierteren Anforderungen an die Dokumentation beschrieben.

8.1 svn

Das svn des Projektes dient neben der Funktion des Repository der Dokumentation. So befinden sich verschiedenste Dokumente im svn. Im Folgenden wird die grundlegende Ordnerstruktur angegeben.

- build
- doc

- afd
 - entwurf
 - handbuch
 - jsf_dokumente
 - klassenbeschreibung
 - latex_dokumente
 - praesentation
 - projektmanagement
 - protokolle
 - seminar
 - sitzungsdokumente
- installer
 - lib
 - release
 - src

Neue Dokumente, das heißt Text, Quellcode oder auch alles andere sind ausreichend mit Kommentaren zu versehen, um den Zweck der Änderung nachvollziehen zu können.

Der Ordner „src“ enthält den Quellcode des Produktes. Eine genauere Übersicht des Aufbaus ist dem Entwurf zu entnehmen. Weiterhin gibt es verschiedene Ordner wie „build“, in den zwischenzeitliche Serverversionen ausgelagert wurden, „installer“, welcher die Installationsroutine enthält, „lib“, die aus Libarys für die Datenbank sowie Equinox besteht, und „release“ welche die abschließenden Dokumente, Sourcecode und das Produkt selber besitzt.

8.2 Protokolle

Protokolle wurden für die wöchentlichen Treffen angefertigt. Spätestens zwei Tage nach dem Treffen sollte das Protokoll im svn zur Verfügung gestellt werden. Außerdem waren die Protokolle per Email-Verteiler zu versenden. Die Protokolle wurden automatisiert in die Internetseite des Projektes eingebunden.

Der Ordner der Protokolle im svn lauteten `doc/protokolle`. Für jedes Protokoll wurde ein Ordner der Form `YYYY_MM_DD` angelegt, in dem sich das Protokoll als TEX-Dokument befindet. Es wurde ein Template bereitgestellt, welches als Muster für die Protokolle verwendet wurde.

Jedes Protokoll sollte so ausführlich sein, dass der Verlauf einer Sitzung nachvollzogen werden konnte. Des Weiteren waren die Ergebnisse der Sitzung und die Aufgaben (TODO) für die kommende Woche mit dem jeweiligen Beauftragten festzuhalten. Die

Tagesordnung für die nächste Sitzung, sowie der nächste Protokollant waren zu bestimmen. Ein Protokollant wurde jeweils beim nächsten Treffen Moderator der Sitzung.

In die Tagesordnung gehörten mindestens die folgenden Punkte

1. Begrüßung und Anwesenheit
2. Protokollant bestimmen
3. Protokoll des letzten Treffens absegnen
4. Feedback zu letzten Woche
 - a) aktueller Stand im Zeitplan
 - b) Ergebnisse kurz präsentieren
5. - aktuelle Punkte -
6. Aufgaben für die kommende Woche und Beauftragte, neues TODO

8.3 Internetseite

Die Internetseite des Projektes ist unter der Adresse www.pg-eam.de zu finden. Sie dient Kunden und Interessierten als Informationsquelle. Auf ihr sind die verschiedenen Projektdokumente downloadbar. Weiterhin werden die Ziele als auch das Projekt vorgestellt.

8.4 Konventionen

Grundlegend gelten die folgenden Konventionen:

- alle Dokumente sind mit kleinbuchstaben bezeichnet,
- Ausnahme bilden Quellcode-Dokumente, hier gilt:
 - Paketbezeichner: `packageOne`
 - Klassenbezeichner: `MyClass`
 - Methodenbezeichner: `myMethod`
 - Variablenbezeichner: `myVariable`
 - Konstantenbezeichner: `MY_CONSTANT`
- als Sprachen für Programmcode werden gewählt:
 - Kommentare und Dokumentation: deutsch
 - Pakete, Klassen, Methoden und co: englisch
- alle Dokumente enthalten im Quelltext Kommentare zum besseren Verständnis,
- nach Möglichkeit ist bei Dateinamen auf Umlaute zu verzichten sonst ä->ae,

- Leerzeichen sind durch Unterstriche `_` zu ersetzen,
- der Programmcode ist ausreichend zu dokumentieren, aber ausdrücklich nicht jede Zeile!
- zur Dokumentation von Programmcode ist JavaDoc zu verwenden,

Listing 1: Code Beispiel

```

1  /*
2   *
3   * Klasse.java
4   *
5   * YYYY-MM-DD
6   *
7   * Copyrighthinweise
8   */
9
10 package ehne.mehne.muh;
11
12 import ehne.mehne.Bla;
13
14
15 /**
16  * Klassenbeschreibung als JavaDoc
17  *
18  * @version x.y.z
19  * @author      Vorname Nachname
20  */
21 public class Klasse extends IrgendeineAndereKlasse {
22
23     /** wenn erforderlich, Info zur Variablen hier */
24     private int ehneCount;
25
26
27     /**
28      * Beschreibung des Konstruktors
29      */
30     public Klasse() {
31         // gebe was in der Konsole aus...
32         // Kommentare nur, wenn es notwendig ist!
33         System.out.println("test");
34     }
35
36
37     /**

```

```

38     * Beschreibung mit optionaler Angabe,
39     * wer das hier verbraucht hat, wenn noch nicht in
40     * der Klassenbeschreibung angegeben
41     *
42     * @author Hans Meier
43     */
44     public void machNix() {
45         if (1 == 1) {
46             // hier passiert nix
47         }
48
49
50         boolean b = false;
51         if (b) {
52             // hier sollte man nicht hinkommen
53         } else {
54             // hier schon eher
55         }
56
57         if (!b) {
58             System.out.println("test_bestanden");
59         }
60     }
61
62
63     /**
64     * Beschreibung
65     *
66     * @author Hans Meier
67     * @param newCount die neue Generation des Zählers
68     */
69     public boolean setEhneCount(int newCount) {
70         ehneCount = newCount;
71         return false;
72     }
73 }

```