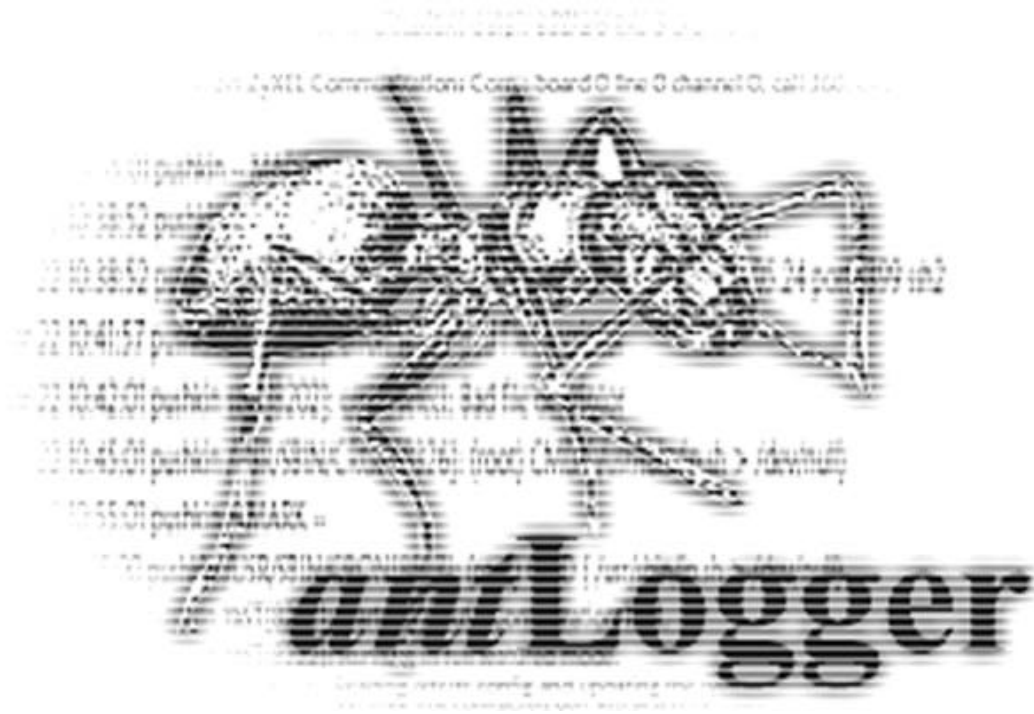


## Software Engineering Projekt



## Software Project Management Plan SPMP

Version 0.1

Patrick Bründler, Pascal Mengelt, Andy Wyss, Fabian Heusser  
Dozent: Jörg Hofstetter

---

## History

| Nr | Datum    | Name | Beschreibung                              |
|----|----------|------|-------------------------------------------|
| 1  | 24.03.02 | Bru  | Dokument erstellt, Mitglieder, Einteilung |
| 2  | 22.04.02 | Bru  | Details beigefügt                         |
| 3  | 06.05.02 | Wys  | Korrekturen                               |

---

# 1. Einführung

## 1.1 Projekt Überblick

Das Ziel dieses Projekts ist die Entwicklung eines Message-Logger. Dieser Message-Logger sollte sehr einfach in Java-Applicationen (im Folgenden „Client“ genannt) eingebunden werden können. Ein Client kann durch Methodenaufrufe Text-Meldungen in einen Message-Logger dauerhaft aufzeichnen und auch online anzeigen lassen (aktuell auftretende Meldungen und/oder alte Meldungen). Die Meldungen aller Client's eines Rechners werden gemeinsam gespeichert. Es ist möglich über ein Tool, das völlig unabhängig vom Client arbeitet, alle diese Messages nachträglich anzuzeigen.

Verwendungszwecke: Der Message-Logger kann z.B. zu Debug-Zwecken (Client meldet durch Meldungseinträge den „zurückgelegten“ Programm-Fluss und Zwischenresultate) oder für's Error-Logging im Felde (Client meldet wichtige Fehlersituationen und notwendige Information dazu) verwendet werden.

## 1.2 Projekt Auslieferung

Folgende Dokumente müssen am Ende des Projektes abgeliefert werden: SCMP, SPMP, SRS, SDD, Bedienungsanleitung, Reviewbericht. Teile aus diesen Dokumenten müssen auch schon früher fertiggestellt werden, siehe dazu 5.1 Meilensteine.

## 1.3 Abkürzungen

|       |                                                   |
|-------|---------------------------------------------------|
| IEEE: | Institute of Electrical and Electronics Engineers |
| SCMP: | Software Configuration Management Plan            |
| SDD:  | Software Design Document                          |
| SPMP: | Software Project Management Plan                  |
| SRS:  | Software Requirements Specification               |

# 2. Projekt Organisation

## 2.1 Organisationsstruktur

Unser Team besteht aus vier gleichberechtigten Mitglieder. Es gibt keine eigentliche Führungsperson. Alle wichtigen Entscheidungen werden im Team gefällt.

## 2.2 Verantwortlichkeiten

### Configuration Leader

Er ist verantwortlich für die korrekte Dokumente und Versionsverwaltung. Das SCMP wird durch ihn erstellt.

### Project Leader

Über ihn läuft die Kommunikation mit dem Dozenten. Er koordiniert auch die Zusammenarbeit mit den anderen Teams. In seinen Verantwortlichkeitsbereich fällt das SPMP.

### Requirements Leader

Der Requirements Leader ist für die Erstellung des SRS verantwortlich.

---

### **Design Leader**

Er erstellt das SDD.

### **QA Leader**

Seine primäre Aufgabe ist das Review, welches beim anderen Team durchgeführt werden muss.

### **Interfacekomiteemitglied**

Als Mitglied im Interfacekomitee muss man zu den Sitzungen dieses Komitees gehen. Und beteiligt sich an den Spezifikationen für die Interfaces.

### **Implementation Leader**

Der Implementation Leader ist für das korrekte Erstellen und zusammenfügen des Codes verantwortlich.

### **Documentation Leader**

Er ist für die korrekte Dokumentation des Codes verantwortlich. Er muss auch sicher stellen, dass eine gute Bedienungsanleitung erstellt wird.

## **3. Verwaltungsprozess**

### **3.1 Risikomanagement**

Das Risikomanagement machen wir im externen Dokument <http://www.w3p.ch/antLogger/xxxx.html>. Unserer Ansicht nach ist dies besser. So können wir zum Beispiel für dieses Dokument, welches viel häufiger ändert, eine eigene Änderungsliste führen.

### **3.2 Überwachungs- und Kontrollmechanismen**

Fertige Dokumente müssen von einem anderen Teammitglied überprüft und korrigiert werden. Schwerwiegende Entscheidungen werden im Team gefällt.

### **3.3 Mitglieder**

|                   |     |
|-------------------|-----|
| Patrick Bründler: | Bru |
| Fabian Häuser:    | Heu |
| Pascal Mengelt:   | Men |
| Andy Wyss:        | Wys |

### **3.4 Personalbesetzung**

|                           |     |
|---------------------------|-----|
| Configuration Leader:     | Heu |
| Project Leader:           | Bru |
| Requirements Leader:      | Men |
| Design Leader:            | Wys |
| QA Leader:                | Men |
| Interfacekomiteemitglied: | Wys |
| Implementation Leader:    | Heu |
| Documentation Leader:     | Bru |

---

## 4. Technischer Prozess

### 4.1 Methoden, Werkzeuge und Techniken

Jedes fertige Dokument wird von einem anderen Team-Mitglied durchgeschaut und auf Fehler überprüft.

Die Team-Mitglieder erstellen die Dokumentation mit Word. Aus fertigen Dokumenten werden PDF erstellt. Die Wahl der Entwicklungsumgebung ist jedem Team-Mitglied freigestellt, denn es ist kein Problem die Javafiles von einer Entwicklungsumgebung in eine andere einzubinden.

### 4.2 Softwaredokumentation

Die Softwaredokumentation wird direkt im Sourcecode eingebunden und mit JavaDoc erstellt.

## 5. Arbeitspakete, Zeitplan und Aufwand

### 5.1 Zeitraum

|               |          |
|---------------|----------|
| Projektstart: | 20.03.02 |
| Projektende:  | 08.05.02 |

### 5.2 Meilensteine

#### M1: 25.03.02

##### Statusbericht:

Organisation der Gruppe liegt vor

Interface Komitee definiert

Erste Risikoliste liegt vor

Zu entwickelndes Produkt für alle Team-Mitglieder im Wesentlichen klar

#### M2: 08.04.02

##### Statusbericht:

SRS bereits angefangen

Resultat des Interface-Komitees liegt vor

Zeitplan mit Meilensteinen liegt vor

Arbeitsplan für die einzelnen Mitarbeiter

#### M3: 15.04.02

Prototypen angefangen

Software Architektur fertig

SPMP fertig

#### M4: 22.04.02

##### Statusbericht:

SRS fertig

Architektur festgelegt

SDD fertig

Erste Software-Teile liegen vor

Peer Review ist organisiert

Betriebsanleitung angefangen

---

**M5: 29.04.02**

Grundsätzliche Funktionen der Software laufen

**M6: 06.05.02**

Komponente ist bereit für das Review  
Betriebsanleitung ist fertig

**M7: 08.05.02 12:00**

Review abgeschlossen

**M8: 08.05.02 Nachmittag**

Präsentationen

### 5.3 Projektphasen

|                         | 11   | 12 | 13 | 14    | 15 | 16 | 17 | 18  | 19 | 20 |
|-------------------------|------|----|----|-------|----|----|----|-----|----|----|
|                         | März |    |    | April |    |    |    | Mai |    |    |
|                         |      |    |    |       |    |    |    |     |    |    |
| Dokumentation erstellen |      |    |    |       |    |    |    |     |    |    |
| Programmieren           |      |    |    |       |    |    |    |     |    |    |
| Review / Tests          |      |    |    |       |    |    |    |     |    |    |
| Vortrag vorbereiten     |      |    |    |       |    |    |    |     |    |    |
|                         |      |    |    |       |    |    |    |     |    |    |
|                         |      |    |    |       |    |    |    |     |    |    |
| Meilenstein 1           |      |    |    |       |    |    |    |     |    |    |
| Meilenstein 2           |      |    |    |       |    |    |    |     |    |    |
| Meilenstein 3           |      |    |    |       |    |    |    |     |    |    |
| Meilenstein 4           |      |    |    |       |    |    |    |     |    |    |
| Meilenstein 5           |      |    |    |       |    |    |    |     |    |    |
| Meilenstein 6           |      |    |    |       |    |    |    |     |    |    |
| Meilenstein 7           |      |    |    |       |    |    |    |     |    |    |
| Meilenstein 8           |      |    |    |       |    |    |    |     |    |    |
|                         |      |    |    |       |    |    |    |     |    |    |

### 5.4 Arbeitsaufteilung

**Woche 13 (25.03.02):**

bru: SPMP, SCMP Review  
heu: SRS  
men: SRS  
wys: SDD, Interface

**Woche 15 (08.04.02):**

bru: SPMP, SDD  
heu: SRS, Prototyp  
men: SRS

---

wys: SDD, Prototyp

**Woche 16 (15.04.02):**

bru: SPMP, SRS Review

heu: SRS, Programmieren

men: SRS, Programmieren

wys: Prototyp, Programmieren

**Woche 17 (22.04.02):**

bru: Betriebsanleitung

heu: Programmieren

men: Programmieren

wys: Programmieren

**Woche 18 (29.04.02):**

bru: Betriebsanleitung

heu: Doku

men: Doku

wys: Programmieren

**Woche 19 (06.05.02):**

bru: Vortrag

heu: Review

men: Review

wys: Doku