

Das Thema dieser Ausgabe:

# Grundlagen erfolgreicher Softwareprojekte

von Stefan Matt, Geschäftsleiter Creasoft AG

**Die Leistungsfähigkeit von Organisationen, die Software erstellen zeigt Unterschiede in der Grössenordnung von 10:1.**

## Woran liegt das?

Ich werde in diesem Dokument einen Überblick darüber geben, was nötig ist, um Softwareprojekte wiederholbar erfolgreich durchzuführen. Das soll Lesern, die selbst keine oder wenig Erfahrung mit Softwareprojekten haben, einen Eindruck über das Durchführen von Softwareprojekten vermitteln.

Ebenso soll es ein Einstiegspunkt in die Bewertung von Organisationen sein, die Softwareprojekte als Lieferanten durchführen.

Das Geheimnis erfolgreicher Softwarefirmen ist das Vorhandensein und Verwenden von beschriebenen **Prozessen**. Dadurch wird erreicht, dass Projekte mit **wiederholbarem Erfolg** auf **definierte Weise** durchgeführt werden können.

## Voraussetzungen für wiederholbaren Erfolg

Es gibt bestimmte Voraussetzungen, die eine Firma haben muss, um überhaupt in der Lage zu sein, Software-Projekte mit wiederholbarem Erfolg durchzuführen. Fehlen diese Voraussetzungen, so ist der Projekterfolg zufällig oder er hängt nur an bestimmten Mitarbeitern. Sobald diese Mitarbeiter nicht mehr zur Verfügung stehen, fehlt der Firma die Kompetenz.

## Anforderungsmanagement

Die Softwarefirma muss sicherstellen, dass am Beginn eines Projekts ein Einverständnis mit dem Kunden über sämtliche Anforderungen an ein Softwareprodukt erreicht wird. Diese Anforderungen müssen detailliert und schriftlich festgehalten werden. Sie sind die Grundlage für alle weiteren Planungen und Aktivitäten. Es muss auch festgelegt werden, wie bei Anforderungsänderungen vorgegangen wird. Es gibt Normen über den Aufbau von Anforderungsdokumenten.

## Projektplanung

Gemeinsam mit Kunden und Entwicklern müssen vernünftige und realistische Vorgehens-, Zeit- und Kostenpläne erarbeitet werden. Insbesondere bedeutet Planung nicht, einfach nur zu allen Terminwünschen ja zu sagen, um Auseinandersetzungen vorerst aus dem Weg zu gehen, oder um einen Auftrag zu erhalten.

## Projektverfolgung

Während des gesamten Projekts sollen alle am Projekt beteiligten und besonders der Kunde Einblick in den Stand des Projekts haben, damit nicht kurz vor dem Ende des Projekts böse Überraschungen auftreten. Dazu muss der aktuelle Stand des Projekts regelmässig mit dem geplanten Stand verglichen werden.

**creasoft ag**

Weiherring 115 . FL 9493 Mauren . T +423 375 8200 . F +423 375 8201 . [www.creasoft.li](http://www.creasoft.li)



### **Management von Unterlieferanten**

Es werden nur qualifizierte Unterlieferanten ausgewählt. Sie erhalten detaillierte Vorgaben und werden regelmässig kontrolliert. Es gibt Kriterien zur Auswahl von Unterlieferanten. Es werden langfristige, partnerschaftliche Beziehungen zu Unterlieferanten aufgebaut. Es gibt kurze Kommunikationswege.

### **Qualitäts-Management**

Das Qualitätsmanagement vergleicht das wirkliche Vorgehen mit dem geforderten. Es überprüft auch die Güte der erstellten Resultate (Dokumente und Programme). Es gibt dem Management Einblick in Projekte.

### **Konfigurationsmanagement**

Das Konfigurationsmanagement stellt die Integrität aller Produkte über den gesamten Lebenszyklus eines Systems sicher. Es wird beispielsweise erreicht, dass nur aktuelle Versionen ausgeliefert werden, dass alle Projektbeteiligten mit denselben aktuellen Dokumenten arbeiten und dass ein Produkt auch nach ein paar Jahren wieder weiterentwickelt werden kann.

### **Vorhandensein eines definierten Prozesses**

Das Vorgehen bei der Durchführung von Softwareprojekten muss definiert sein. Es gibt ein standardisiertes Vorgehensmodell mit festgelegten Dokumenten und Meilensteinen. Dieser Prozess basiert auf folgenden Fundamenten:

- Vermeiden von Fehlern (besonders von klassischen wohlbekannten Fehlern)
- Anwenden der bekannten Grundlagen des Software-Engineering (Best Practices)
- Aktives Risikomanagement

### **Vermeiden von Fehlern/ Umgang mit Risiken**

Dazu werden die Mitarbeiter anhand von geplanten Trainingsprogrammen geschult. Als Unterstützung gibt es Checklisten.

Es gibt bekannte Fehler und Hauptrisiken, die bei jedem Softwareprojekt zum Tragen kommen. Gleichzeitig müssen die speziellen Umstände jedes Projekts berücksichtigt werden.

### **Fundamentale Grundlagen**

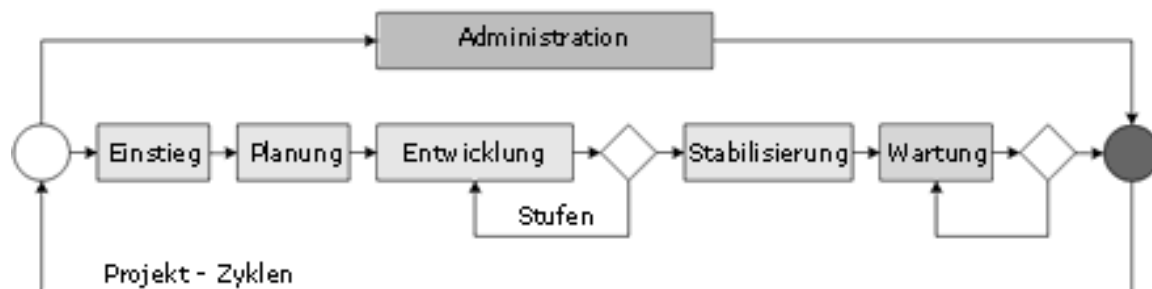
- **Es gibt Codierungsrichtlinien und Namensgebungsrichtlinien**  
Damit wird sichergestellt, dass Programmcode von einem Programmierer auch für andere Programmierer lesbar wird.
- **Richtlinien zur Datensicherung**  
Vorsorge gegen versehentliche Zerstörung, Viren, Hardwareausfälle, Elementarschäden.
- **Richtlinien zu freigegebenen Entwicklungswerkzeugen**  
Es ist sichergestellt, dass nur Werkzeuge und Komponenten verwendet werden, die getestet wurden, Alle verwenden die selben Versionen.
- **Aufzeichnung der im Projekt verwendeten Werkzeuge, Programme und Hardware**  
Damit das Projekt weiterentwickelt werden kann.
- **Installationsanleitungen für die Weiterentwicklung**  
Damit das Projekt weiterentwickelt werden kann.



- **Automatisiertes Versionskontrollsystem**  
Verwalten von Versionen, Änderungen, etc. Schutz vor versehentlichem gegenseitigem Verändern, wenn mehrere Entwickler an einem System arbeiten.
- **Identifizierbarkeit von Produkten anhand eindeutiger Versionsnummern bei jeder geringsten Änderung.**
- **Automatisiertes Verwalten von Fehlermeldungen, Wünschen und Änderungen.**  
Damit nichts vergessen geht.

## Der Software Prozess

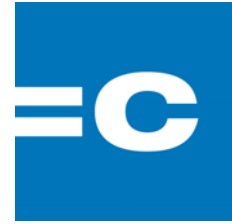
Es gibt verschiedene Vorgehensmodelle. Fast genauso viele sind akademisch und für Projekte der echten Welt nicht anwendbar. Heute wird ein anwendungsfallorientiertes, zyklisches Vorgehen als erfolgreichstes Modell eingesetzt.



Das zyklische Vorgehen bedeutet auch, dass vorteilhaft die Abschnitte Einstieg und Planung als Vorprojekt durchgeführt werden. Danach folgt dann die Entscheidung ob weiter vorgegangen wird. Die Entwicklungsphase muss in überschaubaren Stufen mit überprüfbaren Ergebnissen erfolgen. In den verschiedenen Abschnitten werden folgende Resultate erzeugt.

### Abschnitt Einstieg

- **Vorgehensmodell festlegen**  
Der standardisierte Entwicklungsprozess der Firma wird auf das aktuelle Projekt angepasst.
- **Projektleitbild erstellen / Grenzen festlegen**  
Zusammen mit dem Kunden wird eine Aussage im Umfang von etwa einer Seite erarbeitet, was die Ziele des Projekts sind, und wo die Grenzen sind.
- **Alle Rollen im Projekt besetzen**  
Es werden Verantwortliche für Produktmanagement, Projektleitung, Entwicklung, Test, Schulung und Betrieb des Systems festgelegt. Das Vorgehen bei Anforderungsänderungen wird definiert.
- **Erste Risikoanalyse**  
Zusätzlich zu den klassischen Risiken der Softwareentwicklung werden projektspezifische Risiken festgelegt.
- **Wunschtermine festhalten**  
Die Wunschtermine werden schriftlich festgehalten.
- **Projektinfrastruktur einrichten**  
Kommunikation der Projektmitglieder, Sitzungsturnus, Versionskontrolle, VPN, Projekt-Homepage, Projektordner, etc.



## Abschnitt Planung

- **Detaillierter Prototyp der Benutzeroberfläche**  
Der Prototyp muss alle Dialoge mit allen Feldern enthalten.
- **Anforderungen detailliert definieren** (Pflichtenheft, Anwendungsfälle)  
Es wird beschrieben, was die Benutzer mit dem Programm machen können. Das normale Verhalten, und das Verhalten in Ausnahmesituationen wird beschrieben. Der Kunde hat wegen seiner Kenntnisse des Fachbereichs meist unausgesprochene Anforderungen an ein System. Die Entwickler müssen sich in den Fachbereich einarbeiten.
- **Analysemodelle** (Datenbanken, Systemstruktur)  
Um die Komplexität des Systems zu verstehen müssen Modelle erstellt werden. Das dient zur Grössenschätzung der Aufgabe.
- **Beginn mit Systemdesign**  
Es müssen erste Konzepte zur Umsetzung gemacht werden, weil es hier Rückkopplungen zu den Anforderungen gibt. Zum erstellen eines Zeitplans muss ein grobes Umsetzungskonzept vorhanden sein.
- **Beginn mit Benutzerdokumentation**  
Die Benutzerdokumentation wird als Beschreibung zum Prototypen erstellt.
- **Detaillierter Projekt-, Zeit- und Kostenplan**  
Nun kann eine erste verbindliche Grössenschätzung des Projekt vorgenommen werden, die aber immer noch mit Unsicherheit behaftet ist. Die Genauigkeit der Schätzung hängt stark mit der Genauigkeit der Anforderungsdefinition zusammen.
- **Risiken verfeinert analysieren**  
Neu erkannte Risiken müssen bearbeitet werden. Einige bisherige Risiken werden bereits wegfallen.

## Abschnitt Entwicklung

Hier wird zuerst das Design - der Bauplan der zukünftigen Software - erstellt. Es ist wichtig, die Projektleitung auch in diesem Abschnitt nicht zu vernachlässigen.

- **Design Datenhaltung**  
Datenbankmodelle, Dateiformate etc.
- **Design Systemstruktur**  
Aufteilung des Systems in Komponenten, interne Struktur der Komponenten, etc.
- **Benutzerdokumentation**  
Anhand der nun feststehenden Oberfläche kann das Benutzerhandbuch fertiggestellt werden.
- **Testvorschrift / Abnahmekriterien**  
Aus den Anforderungen entstehen die Testfälle. Ev. muss mit einem Werkzeug ein Testprogramm für automatische Regressionstests erstellt werden, damit bei jeder neuen Version die bestehenden Funktionen wieder getestet werden können.
- **Installationsprogramm und Anleitung**  
Möglichst früh muss ein Installationsprogramm erstellt werden, das auf allen Zielplattformen getestet werden kann. Dazu ist es nötig Installationssysteme zu haben, wo rasch verschiedene Versionen der Zielplattformen bereitgestellt werden können.
- **Versionsmanagement**  
Es muss immer nachvollziehbar sein, was in welcher Version von wem wieso gemacht wurde. Alte Versionen müssen zu jeder Zeit wiederhergestellt werden können.



- **Verwaltung von 'Anomalitäten'**  
Alle Wünsche, Fehler, Ideen, Änderungen, Bemerkungen, offenen Punkte etc. müssen festgehalten und verwaltet werden.
- **Periodische Projektkontrolle**  
Projektfortschritt, Risiken, Kosten müssen regelmässig (2-3 Wochen) in Projektsitzungen überprüft werden.

### **Abschnitt Stabilisierung**

- **Testprotokolle**  
Die Resultate von Tests müssen schriftlich festgehalten werden.
- **Projektarchivierung für spätere Weiterentwicklung**  
Es muss sichergestellt sein, dass ein Programmierer zu einem späteren Zeitpunkt das Projekt weiterentwickeln kann. Dazu müssen alle Entwicklungswerkzeuge, alle zugekauften Komponenten und die Sourcen archiviert sein. Es muss Anleitungen zur Neuinstallation der Entwicklungsumgebung geben. Diese Anleitungen müssen auch verifiziert werden.
- **Übergabe des Quellcodes an den Kunden / Kompilationstest**  
Es muss überprüft werden, ob der Kunde eine lauffähige Version des Programms erstellen kann.
- **Auslieferung des fertigen Produkts**  
Die Endversion des Produkt muss zur Verfügung gestellt werden.

© Creasoft AG, 2002 - Alle Rechte vorbehalten