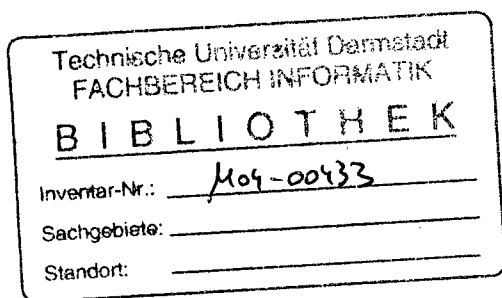


Friedrich Weltz, Rolf G. Ortman

Das Softwareprojekt

Projektmanagement in der Praxis



Campus Verlag
Frankfurt/New York

INHALTSVERZEICHNIS:

VORBEMERKUNG

11

SOFTWAREENTWICKLUNG ALS DOPPELPROZESS VON TECHNIKGESTALTUNG UND ARBEITSSTRUKTURIE- RUNG

13

Das Verständnis des besonderen Charakters von Softwareprojekten wie der Anforderungen, mit denen sich deren Gestaltung auseinanderzusetzen hat, erschließt sich nicht zuletzt aus dem Anwendungsbezug der entwickelten Produkte. Software liefert nicht nur technische Lösungen, sie strukturiert zugleich das Feld, in dem sie Anwendung findet. Dort, wo die Aufgaben von Menschen wahrgenommen werden - und dies ist ja noch immer zum weitaus größten Teil der Fall - heißt dies: Softwareentwicklung ist zugleich Technikgestaltung und Arbeitsstrukturierung.

I. ZIELDEFINITION

19

1. Die Projektentstehung - die Weichen werden gestellt

19

Wie Projekte entstehen, durch wen und wie präzise der Projektauftrag formuliert wird, ist für deren weiteren Ablauf und letztlich für ihr Ergebnis von großer Bedeutung. In vielen Projekten werden allerdings die Weichen von Anfang an falsch - oder schlimmer noch - gar nicht gestellt. Die Auswirkungen sind dann im weiteren immer wieder spürbar.

2. Das U-Boot-Verfahren - subversive Projektinitialisierung

26

Fast jedes dritte der untersuchten Projekte wurde ohne offiziellen "Segen" gestartet. Dem formalen Projektauftrag ging eine Phase voraus, in der bereits wesentliche Vorarbeiten geleistet und damit unter der Hand Fakten geschaffen wurden, die dann den weiteren Ablauf bestimmten.

3. **Die Anforderungsspezifikation - Reduzierung von Unbestimmtheit** 30
Die Bedeutung der frühen Phasen der Softwareentwicklung für deren weiteren Verlauf und Ergebnis wird zunehmend betont. Dies wird durch die Ergebnisse unserer Untersuchung sehr nachdrücklich bestätigt. Nicht nur, daß diese "frühen" Phasen in vielen Projekten einen beträchtlichen Anteil am gesamten Zeitaufwand erforderten, sie gestalteten sich zugleich auch unerwartet schwierig.
 4. **Projektkosten und Termine - nur selten richtig geschätzt** 35
Versuche, den Umfang von Softwareprojekten vorherzubestimmen, sind nur teilweise erfolgreich: Aussagen über die zu erwartenden Kosten liegen meist zu niedrig, werden häufig im Projektverlauf korrigiert - und dann doch nicht eingehalten.
 5. **Bestimmung des Projektumfangs - Unterschätzung als Methode** 40
In einem beträchtlichen Teil der Softwareprojekte wird der zu Beginn gesetzte Kosten- und Terminrahmen erheblich überzogen. Dies ist nur zum Teil auf fehlenden Einsatz oder mangelhafte Nutzung systematischer Verfahren zur Aufwandsermittlung zurückzuführen.
-
- ## II. ARBEITSTEILUNG 49
1. **Projektorganisation - normative Konzepte und situative Anpassung** 49
Die Gestaltung des organisatorischen Rahmens variiert von Projekt zu Projekt sehr stark. Sie wird weniger durch normative Konzepte als durch die Anpassung an die jeweiligen Rahmenbedingungen bestimmt.
 2. **Arbeitsteilung im Projektteam - Kooperation und Spezialisierung** 55
Wie die formale Projektorganisation variiert die Arbeitsteilung in den Projektteams von Projekt zu Projekt sehr stark. Sie ist häufig weniger Resultat formaler Aufgabenzuweisung, sondern eher das Ergebnis eines naturwüchsigen Prozesses.
 3. **Schlüsselpositionen - ein ausschlaggebender Leistungsfaktor** 61
In sehr vielen Projekten sind Positionen zu erkennen, die bei der Koordination und der gegenseitigen Information Schlüsselfunktionen einnehmen. Diese Positionen sind fast durchweg überfrachtet.

- 4. Fremdleistungen - Probleme werden nach außen "delegiert" 65**
- Die Möglichkeit, Entwicklungsarbeiten nach außen zu vergeben, eröffnet bei der Abwicklung von Softwareprojekten zusätzliche Gestaltungsmöglichkeiten, konfrontiert aber auch mit besonderen Anforderungen. Die Arbeitsteilung von externen und internen Leistungen ist damit wesentlicher Aspekt des Projektmanagements. In der Praxis wird dies allerdings vielfach kaum berücksichtigt.
- 5. Anwender-Entwickler-Beziehungen - Kooperation statt Beteiligung 72**
- Die Einbeziehung der Anwender in die Entwicklung von Software ist notwendig. Die Konzepte der Anwenderpartizipation allerdings reichen nicht mehr aus. Erforderlich ist eine kooperative Systementwicklung, an der beide Systeme gleichermaßen beteiligt und gefordert sind.
- III. PROZESSGESTALTUNG 81**
- 1. Ablaufsteuerung - zwischen Phasenschema und Selbststeuerung 81**
- Die Organisation des Projektablaufs orientiert sich überwiegend am Phasenschema, das dann allerdings in der Praxis häufig situativ an die jeweiligen Bedingungen angepaßt wird. Dabei spielen Prozesse der Selbststeuerung durch die Entwicklerteams eine große Rolle.
- 2. Personaleinsatz - eine vernachlässigte Gestaltungsdimension 88**
- Die Planung und Gestaltung des Personaleinsatzes ist für den Erfolg von Softwareprojekten von erheblicher Bedeutung, nicht nur unter Kosten- und Terminaspekten, auch für deren inhaltliches Ergebnis.
- 3. Qualitätssicherung - zwischen Planung und Ausführung 95**
- Die Bedeutung der Qualitätssicherung wird zwar durchweg betont, in der Projektpraxis wird sie aber häufig Opfer des Zeitdrucks oder anderer Umstände. Erforderlich sind Formen der Qualitätssicherung, die integraler Teil der Entwicklungsarbeiten sind.

**1. Qualifizierung -
Eigeninitiative und kollegiale Unterstützung** **105**

Die beträchtlichen Anforderungen an die Aneignung neuen Wissens, denen Softwareentwickler in fast jedem Projekt gegenüberstehen, werden bewältigt in einem - meist recht effektiven - Prozeß, in dem Wissensaneignung sowie die Übernahme und Bewältigung von Aufgaben eng verschränkt sind.

**2. Stille Leistungen -
schwer vorzuschreiben, aber unentbehrlich** **112**

Neben den Tätigkeiten, die offiziell als Aufgabe der Entwickler ausgewiesen sind, werden von diesen vielfach "stille" Leistungen erbracht. Solche stillen Leistungen sind für die Projekte und die Qualität der Software von großer Bedeutung, allerdings nur schwer zu fassen und kaum vorzuschreiben.

**3. Projektübergreifender Wissenstransfer -
dem Zufall überlassen** **118**

Der systematische projektübergreifende Transfer von Wissen und Erfahrungen wird in der Softwareentwicklung vielfach vernachlässigt. Dabei kommt ihm für die erfolgreiche Abwicklung von Projekten große Bedeutung zu.

V. KONSENSBILDUNG **125**

1. Konflikte - Verdrängung statt Verarbeitung **125**

Softwareentwicklung ist konfliktträchtig. Häufig werden Konflikte nicht offen, sondern verdeckt ausgetragen. Gerade dies kann die Entwicklungsarbeiten belasten. Die Verarbeitung von Konfliktstoff ist wesentliche Voraussetzung für eine erfolgreiche Abwicklung von Softwareprojekten.

2. Krisen - gefährlich, aber konstruktiv **132**

Krisen gehören zum normalen Schicksal von Softwareprojekten. Sie können Projekte gefährden, sie können aber auch Spannungen abbauen, Fehlentwicklungen korrigieren, die den Ablauf der Entwicklungsarbeiten belastet haben. Es gilt, Krisen konstruktiv zu nutzen.

**3. Synchronisierung der Entwicklungsprozesse -
eine Notwendigkeit** **138**

Die Definition der Vorgaben, an denen sich die Entwicklungsarbeiten orientieren, erfolgt in vielen Projekten im Vorlauf zur Etablierung des notwendigen Wissensstands und Konsenses unter den Beteiligten. Gefordert ist eine Synchronisierung der Entwicklungsprozesse.

VI. ERFOLGSVORAUSSETZUNGEN

147

1. Der Projekterfolg - eine Frage der Maßstäbe 147

Bei über einem Drittel der untersuchten Projekte wird der Erfolg kritisch oder kontrovers eingeschätzt. Dabei fehlen allerdings systematische Versuche zur Evaluierung fast völlig.

2. Die Situation der Entwickler - zwischen Identifikation und Überforderung 154

Die Tätigkeit vieler Softwareentwickler ist mit hohen Beanspruchungen verbunden. Daß man bereit ist, diese auf sich zu nehmen, ist nicht zuletzt auf den Gestaltungsspielraum, den man bei der Arbeit hat, zurückzuführen.

3. Die Softwarekrise - Krise des Projektmanagements 158

In den untersuchten Projekten stellt sich die "Softwarekrise" wesentlich als eine Krise des Projektmanagements dar, d.h. der Konzepte, Verfahren und Instrumente, die zur Planung, Steuerung und Kontrolle der Projektabwicklung herangezogen werden. Sie erweisen sich als weitgehend unzulänglich, die prozessuale wie die strukturelle Komplexität der Entwicklungsprozesse in den Griff zu bekommen.

4. Vom Projektmanagement zum Prozeßmanagement 164

Softwareentwicklung ist fast immer kein abgeschlossener Akt, sondern ein fortlaufender Prozeß technischer und sozialer Gestaltung. Das organisatorische Konstrukt "Projekt" erschwert eine fortlaufende Prozeßgestaltung, wird aber wegen seiner Vorteile unter Steuerungs- und Kontrollaspekten beibehalten.

5. Softwareentwicklung als Entscheidungsprozeß - zwischen Offenheit und Verbindlichkeit 168

Die inhärente Logik der Softwareentwicklungsprozesse legt zunehmend ein inkrementelles Vorgehen nahe. "Statische" Verfahren des Projektmanagements geraten in Widerspruch zu den Anforderungen solcher Entwicklungsprozesse. Neue, dynamische Konzepte des Projektmanagements werden erforderlich.

6. Zusammenfassung: Anforderungen an ein dynamisches Projektmanagement 176

VII. ANMERKUNGEN ZUR UNTERSUCHUNGSMETHODE

181