

Leitfaden zur

Sabotage von IT-Projekten

105 taffe Taktiken für Teamleiter

Autor: Bernd Schulte Osthoff
complexity coach
IT's people! Soft Skills für
Führungskräfte in der IT
www.itspeople.de

Kontakt: E-Mail: itsabotage@itspeople.de
Fon&Fax: 0700.1618.0000

Sie dürfen diesen Text weitergeben, wenn Sie ihn nicht verändern. Sie dürfen auf ihn einen Link setzen: www.itspeople.de/skurriil

Sie dürfen diesen Text ganz oder in Teilen für Ihre Präsentationen benutzen, wenn Sie dabei das Logo „IT's people“ und die Domainangabe „itspeople.de“ unverändert lassen.

Wählen Sie Projekte mit unmöglichen Terminvorgaben.

- 1) Bei Projektausschreibungen ist es üblich, unerreichbare und unrealistische Termine zu verlangen, wenn diese überhaupt angegeben werden. Das weiß auch Ihre Konkurrenz. Also bewerben Sie sich. Sagen Sie On-Time&On-Budget zu. Das machen alle so. Wenn Sie den Auftrag erst einmal haben, wird Ihre Firma schon irgendwie Geld verdienen.
- 2) Entwickler lieben Herausforderungen, also erzählen Sie ihnen, dass die Termine mit sportlichem Ehrgeiz locker zu schaffen sind.
- 3) Dann ziehen Sie sich in Ihr Büro zurück und lachen herzlich.

Übernehmen Sie nur Festpreis-Projekte!

- 4) Projekte nach Aufwand sind für Zögerer und Zauderer. Nur bei einem Festpreis-Projekt können Sie sicher sein, dass alle Vorgaben, Vereinbarungen und Aufwandsschätzungen auf jeden Fall weitab von jeder Realität liegen. Nur hier haben Sie eine sichere Aussicht auf ein vollständig gesprengtes Budget..
- 5) Bei einem bereits laufenden Festpreis-Projekt ist das gesamte Budget mit hoher Wahrscheinlichkeit schon längst aufgebraucht. Nichts entfaltet solche Wirkungen wie unbezahlte Überstunden.
- 6) Aus Sicht des Kunden übernehmen nur absolute Profis ein Festpreis-Projekt. Das bedeutet für Ihr Team den ständigen Drang zu kreativen Kosteneinsparungen und zusätzlicher Produktivität.

Arbeiten Sie nur für weit abgelegene Kunden.

- 7) Sie brauchen Ihrem Team keine Unterbringungs-alternativen bieten; nehmen Sie, was der Kunde hat: ausgediente Lagerräume, fensterlose Keller oder unbelüftete Abstellräume.
- 8) Vermeiden Sie Kunden im Innenstadtbereich; Ihr Team hätte zuviel Ablenkung und Ausweichmöglichkeiten für Essen und Einkaufen.
- 9) Ziehen Sie Kundenlocations in Industriegebieten oder sozialen Randbereichen vor; damit vermeiden Sie überlange Spaziergänge Ihrer Mitarbeiter in den Pausen.

Weniger ist mehr bei der Projektbesetzung.

- 10) Besetzen Sie Ihr Projektteam niemals nach den angebotenen oder beauftragten Qualifikationen
- 11) Ein Jahr Berufserfahrung genügt für die Position eines Gruppenleiters. Setzen Sie das Team unter ständigen Druck.
- 12) Zwei Jahre Berufserfahrung genügen für die Position eines Projektleiters. Auch ohne vorherige Führungserfahrung.
- 13) Drei oder mehr Jahre Berufserfahrung sind zu teuer. Hier besteht außerdem die Gefahr selbständigen Denkens und Handelns, unter Missachtung Ihrer Autorität.

Berufen Sie nur Externe ins Team.

- 14) Ignorieren Sie die erfahrenen Mitarbeiter in Ihrer Firma, die gerne mit den neuesten Technologien dieses neuen Projektes arbeiten und lernen würden. Viele dieser Entwickler sind ohnehin an die oder über 40.
- 15) Unvorbelastete Junginformatiker, frisch von der Uni, sind ideal für die Anforderungs- und Designphasen im Projekt.
- 16) Verschaffen Sie unbedingt nur den Externen den Zugang zu den neuesten Technologien. Die Internen kommen mit den Schwächen der alten Systemen Ihrer Firma besser klar.
- 17) Zahlen Sie den Externen einen deutlich höheren Anteil am Budget und lassen Sie dies inoffiziell alle wissen. Das spornt alle gleichermaßen an.

Aufwand ist **relativ**.

- 18) Der Projektaufwand folgt den Regeln der Quantenmechanik, ist also abhängig vom Beobachter und von niemandem wirklich genau zu bestimmen. Sie haben die freie Auswahl.
- 19) Für eine erste Schätzung empfiehlt sich die DB-Methode: nehmen Sie ein DartBoard, werfen Sie drei mal, und errechnen Sie den Mittelwert. Informieren Sie NICHT Ihr Team.
- 20) Lassen Sie Ihr Team den Aufwand selbst ermitteln. Weisen Sie die Ergebnisse mindestens $n-1$ Mal als zu hoch zurück (n = Anzahl der verbleibenden Projektphasen).
- 21) Der Kunde weiß, dass er keinen Einfluss auf den Aufwand hat. Lassen Sie ihn nicht merken, dass Sie wissen, dass er das weiß.

Nutzen Sie das Sparta-Prinzip: Engpaß Hardware.

- 22) Maximal ein Festnetz-Telefon pro Arbeitsgruppe. Nur für den Ernstfall bekommt der Kunde zu Projektbeginn die Handy-nummern des gesamten Teams.
- 23) Remote-Zugänge dürfen nur nach Genehmigung durch das Controlling genutzt werden. Die Genehmigung muss wöchentlich erneuert werden.
- 24) Das Alter der Entwicklungssysteme sollte proportional zur Betriebszugehörigkeit der Mitarbeiter sein. Aufrüstungen oder Updates vor und während des Projekts sind abzulehnen.
- 25) Sind Aufrüstungen oder Updates nicht zu vermeiden, maximieren Sie den Spieltrieb-Faktor.

Jedem das Seine.

- 26) Der Kunde möchte, dass Sie so bald wie möglich Ergebnisse liefern. Also weisen Sie Ihr Team an, nicht auf die Designdokumente oder Methodikstandards zu warten und sofort anzufangen. Spaghetticode ist hier akzeptabel.
- 27) Formale Methodiken sind unausweichlich. Der Kunde wird danach fragen. Schicken Sie zwei aus Ihrem Team dann auf einen Crashkurs, als Multiplikatoren für die anderen. Wählen Sie später eine völlig andere Methodik. Sie sind der Boss.
- 28) Ihr Team wird eigene Vorstellungen haben. Wählen Sie eine Methodik aus und zwingen Sie alles in dieses Vorgehen. Auch wenn die Methodik ungeeignet oder veraltet ist.

Statusmeetings für alle!

- 29) Verlangen Sie die Anwesenheit des gesamten Teams zu allen Statusmeetings. Das erzeugt Gemeinschaftssinn und ist eine willkommene Abwechslung. Auch wenn der Einzelne kein Interesse oder keinen Nutzen von dem hat, was besprochen wird.
- 30) Wenn konkrete Probleme anstehen, fragen Sie nicht Ihr Team nach Vorschlägen. Laden Sie zwei Experten aus anderen Projekten ein. Die entstehenden Fachdiskussionen sind für alle eine Bereicherung.
- 31) Vermeiden Sie es, über Anforderungsspezifikationen zu diskutieren. Der Kunde weiß nicht, was er will. Das Team weiß nicht, was der Kunde will. Und Sie wollen flexibel bleiben, bis zum Projektende.

Mischen Sie sich unter's Volk.

- 32) Als Manager sind Sie nicht unbedingt bei den laufenden Meetings erwünscht. Übernehmen Sie deshalb kleine Entwicklungsaufgaben pro forma. So kann Sie niemand mehr ausschließen.
- 33) Werden bei einem Meeting Design- oder Codier-Fehler erwähnt, unterbrechen Sie das Meeting und verlangen Sie die sofortige Beseitigung. So halten Sie das Team auf Trab.
- 34) Schlagen Sie laufend Änderungen oder „interessante“ Erweiterungen vor, die der Kunde lieben wird. Keiner wird wagen, Ihnen zu widersprechen.
- 35) Ändern Sie regelmäßig nach Feierabend die Passwörter der Server und Entwicklersysteme. Sicherheit geht vor.

Geschwindigkeit ist keine Hexerei.

- 36) Ihr Kunde liebt schnelle Systeme. Setzen Sie daher von Anfang an die Geschwindigkeit der Software auf die oberste Priorität. Hängen Sie überall Formel-1-Poster auf.
- 37) Lassen Sie es nicht zu, dass für Teilsysteme externe Softwarebausteine eingekauft werden. Hier haben Sie keine Kontrolle über die Geschwindigkeit.
- 38) Wiederverwendbare Module, Objekte oder Klassen sind zu langsam. Plan B: gründen Sie ein Design-Komitee für Wiederverwendbarkeit. Berufen Sie Ihr ganzes Team. Das sorgt für Zeitplan sprengende Diskussionen.

Unit-Testing geht vor.

- 39) Jede Unit ist vor jedem Build eingehend zu testen. Alle Schnittstellen müssen exklusiv für den Test programmiert werden. Alle Testergebnisse müssen vom Teamleiter geprüft und abgezeichnet werden. Verlangen Sie ein professionelles Dokumentations- und Ablagesystem.
- 40) Treffen Sie keinerlei Vorbereitungen für Integrationstests. Schließlich ist nicht abzusehen, wann alle Unit-Tests abgeschlossen sind. Außerdem können Sie sich ja jederzeit Testdaten aus anderen Projekten ausleihen, zusätzlich benötigte Hardware beim Kunden nutzen, und jederzeit auf die Unit- und Subsystem-Leute zurückgreifen.

Alte Hasen in Reserve.

- 41) Setzen Sie für die Integrationstests unbedingt jemanden ein, der keine Ahnung von den im Projekt verwendeten Systemen hat. Für ein Java- oder Linux-Projekt ist ein MVS-Veteran ideal.
- 42) Auch jemand aus dem mittleren Management ist ideal. Oder in kleinen Firmen der Chef selbst. Diese Leute haben jahrelang nicht mehr programmiert, erzählen aber immer gern von den glorreichen Anfangsjahren als Programmierer.
- 43) Folgende Verhaltensweisen identifizieren den idealen Integrationstester: ist viel in der Firma unterwegs, spricht ständig mit Kreti und Pleti, liest selten E-Mails, mag den Kunden nicht, kritisiert ständig die Firmenpolitik.

Profis sind **anpassungsfähig.**

- 44) Lassen Sie jederzeit Änderungen zu. Change Requests sind in Ordnung, solange sie für das Projektmanagement, das Prozessmanagement, das Qualitätsmanagement und für den Vorgehensmodell-Standard dokumentiert werden.
- 45) Falls die Arbeit am Projekt einmal zu glatt läuft, nutzen Sie die Zeit und schalten Sie bei ein oder zwei Änderungen die Verantwortlichen für Projektmanagement, Prozessmanagement, Qualitätsmanagement und Vorgehensmodell-Standard miteinander kurz.
- 46) Meilensteine sind die Ausnahme. Sie müssen eingehalten werden. Koste es, was es wolle. Wenn zu diesem Zweck jede Menge Provisorien eingebaut werden, um so besser.

Man muss es nur begründen können.

- 47) „Das haben wir doch schon längst freigegeben“
(übersetzt: „Jetzt tanz‘ Du nicht schon wieder aus der Reihe“).
- 48) „Diese Änderungen sind fixiert, bekannt und dokumentiert“
(übersetzt: „Lass bloß die Finger davon“)
- 49) „Das ist nur eine kleine Änderung, das kriegt ihr zwischendurch hin“ (übersetzt: eine Woche Mehrarbeit)
- 50) „Das ist der Abschluss von Change Request Nr....“
(übersetzt: „Keiner weiß, wieso – warum – woher“)
- 51) „Das sind Fehlerkorrekturen, die erledigt ihr mit Links“
(übersetzt: „WER hat denn wohl den Murks produziert, ha?“)

Zeigen Sie, was Sie können!

- 52) Wenn die Arbeit zu glatt vonstatten geht, beraumen Sie eine Präsentation an (wichtig: alter Beamer, lüfterloser Raum)
- 53) Nutzen Sie die Projektionsfläche. Full-Size-Screenshots aus Excel sind optimal.
- 54) Zeigen Sie alle Lösungsvarianten, die in Arbeit sind (schließlich weiß keiner genau, wohin die Reise geht)
- 55) Erinnern Sie an alle Anforderungen an die fertige Lösung (damit lösen Sie in der Regel absolutes Stillschweigen aus)
- 56) Zeigen Sie alle Fehlerprotokolle der letzten Tests (das motiviert ungemein für den unmittelbar anstehenden Test)

Meetings machen munter.

- 57) Wenn Ihr Team überarbeitet erscheint oder die Fehlerraten durch Übermüdung ansteigen, beraumen Sie ein Meeting an. Optimal ist früh morgens nach einem Integrationstest oder Build, oder abends nach 18:30 Uhr.
- 58) Nehmen Sie immer ein oder zwei Entwickler zu den Kundenmeetings mit. Erlauben Sie technische Erläuterungen.
- 59) Bringen Sie Ihr Team alle zwei Wochen mit anderen Teams zusammen, um die Fortschritte beim Qualitätsmanagement und der Prozess-Standardisierung zu feiern.
- 60) Rufen Sie das gesamte Team zusammen, um die besuchenden Entwickler über die Erlebnisse beim letzten Kundenmeeting berichten zu lassen.

Streber bremsen.

- 61) Wenn eines Ihrer Teams besonders gut und schnell ist, übertragen Sie ihm zusätzliche Aufgaben für Dokumentation und Test.
- 62) Stellen Sie einzelne Mitglieder des guten Teams vorübergehend in schlechten Teams ab. Die sollen sich daran ein Beispiel nehmen.
- 63) Machen Sie aus einem guten Team zwei, die Sie dann mit Under-Performern auffüllen. So erhalten Sie ein gesundes Gleichgewicht.
- 64) Wenn Sie das gute Team nicht ohne Aufsicht teilen können, weisen Sie alle Neueinstellungen diesem Team zu. Für Einweisung und betriebsinterne Ausbildung geht eine Menge Zeit drauf, die auf die Projekte gebucht werden kann.

Bleiben Sie sauber.

- 65) Wenn Ihre Testprotokolle zu viele Fehler ausweisen, sind Ihre Entwickler nicht gut genug. Machen Sie entsprechend Druck.
- 66) Wenn Ihre Testprotokolle zu wenige Fehler ausweisen, sind Ihre Tester nicht gut genug. Machen Sie entsprechend Druck.
- 67) Alle Tests sind bei jeder Änderung zu wiederholen, auch wenn nur Kommentare oder Prüfroutinen entfernt wurden. Wenn sich jemand beschwert, verweisen Sie auf das Qualitätsmanagement, die internen Firmenstandards oder ISO oder CMM oder ITIL oder ...
- 68) Wenn trotz Allem zu wenig Fehler gefunden werden, verändern Sie „probehalber“ die Laufzeitparameter der Compiler, Generatoren oder Testumgebung. Diese Änderungen sind irreversibel und erhöhen die Fehlerzahl garantiert. Ideal ist z.B. ein höherer Optimierungslevel.

Bleiben Sie **sachlich!**

- 69) Entwickler in der IT lieben es, technische Probleme zu lösen. Sorgen Sie für die nötige Dosis „touchy-feely“, politisch-psychologischer Herausforderungen. Wechseln Sie die Teamleiter. Einfach so.
- 70) Sie wissen, dass TQM die Abkürzung für „Torquemada“ ist, den spanischen Großinquisitor? Nur durch körperliche Grenzerfahrungen kann man wachsen: „Was uns nicht umbringt, macht uns hart“!
- 71) Entwickler lieben das Rationale. Halten Sie sich also von ihnen fern. Kein persönlicher Small-Talk. Sprechen Sie immer zuerst von den Ergebnissen der letzten Tests oder über das Budget.

Demonstrieren Sie Souveränität!

- 72) Niemand darf sich länger als sechs Minuten in Ihrem Büro aufhalten.
- 73) Jeder hat sich an den Dress-Code zu halten, auch am Wochenende.
- 74) Verlassen Sie Ihr Büro nur, um Ihrem Team eine dringende Reklamation des Kunden vorzulegen.
- 75) Tragen Sie Ihre Anzugjacke wie einen königlichen Umhang, um die Schultern gelegt.
- 76) Gewähren Sie dem Kunden direkten Zugriff auf die Entwickler.
- 77) Widerrufen Sie die Vorgaben und Anweisungen des Kunden mit gebührendem zeitlichen Abstand.
- 78) Machen Sie Urlaub, wenn das Projekt in die Endphase geht.

Sorgen Sie für Zuversicht!

- 79) Bitten Sie einen Bekannten aus einem anderen Unternehmensbereich, ein Verbesserungsmeeting mit Ihrem Team zu leiten.
- 80) Bitten Sie Ihren Bekannten, auf alle Verbesserungsvorschläge begeistert und empathisch zu reagieren: „Das kann ja nicht wahr sein!“, „Da muss was passieren!“, „Wenn erst die oberste Leitung davon erfährt!“.
- 81) Notieren Sie die Namen der Vorschlagenden. Nur damit Sie Ihnen später persönlich Rückmeldung geben können. Und für Ihre persönliche Liste der Störenfriede, natürlich.
- 82) Unmittelbar nach dem Meeting informieren Sie Ihr Team in einem Memo, dass das „Management“ sich eingehend mit den Vorschlägen befasst und in Kürze erste Änderungen vornehmen wird.

Überstunden sind kein Grund zur Freude.

- 83) Aus Stolz setzen Entwickler alles daran, ihre Aufgaben abzuarbeiten. Das ist eine ernsthafte Persönlichkeitsstörung.
- 84) Wenn Ihre Entwickler motiviert bis in die Nacht arbeiten, führen Sie reguläre Schichtarbeit ein. Mit unregelmäßigen Wechseln und ohne Zuschläge. Das ist eine der besten Therapien.
- 85) Stellen Sie jeden vor versammelter Mannschaft zur Rede, der versucht, Überstunden auf das Projekt zu buchen. Er sabotiert schließlich die unternehmensinterne Leistungsstatistik.
- 86) Engagieren Sie unverheiratete Freelancer, die nach Stunden bezahlt werden. Die machen mit Freuden 100 Stunden die Woche. Wer immer mit familiären, gesundheitlichen oder anderen fadenscheinigen Gründen früher gehen will, wird das bald lassen.

Kaufen Sie qualifizierte Unterstützung von außen zu.

- 87) Engagieren Sie im letzten Drittel des Projekts freie Mitarbeiter von Zeitarbeitsfirmen oder Vermittlungsagenturen. Deren Einweisung bringt das Projekt, wenn auch nur vorübergehend, zum Stillstand.
- 88) Zahlen Sie ihnen das Doppelte der internen Stundensätze. Die Arbeitsergebnisse mögen es nicht wert sein, aber die Wirkung auf die Moral der angestellten Mitarbeiter ist unbezahlbar.
- 89) Setzen Sie die Leiharbeiter an die Stellen des Projekts, wo es viel zu lernen und viele Überstunden zu leisten gibt. Fordern Sie von Ihren Mitarbeitern mindestens den gleichen Einsatz. Natürlich ohne Bezahlung der Überstunden.

Sondern Sie Theoretiker aus.

- 90) In jedem Team gibt es Mitarbeiter, die mit ihren fachlichen Kenntnissen glänzen und diese immer wieder den Kollegen zu Gute kommen lassen wollen.
- 91) Nehmen Sie diese Gutmenschen aus dem Team und beauftragen Sie sie, Artikel und Beiträge für die Firmenhomepage zu schreiben.
- 92) Loben Sie Arbeit der Theoretiker. Reichen Sie deren Beiträge zu Kongressen ein. Schicken Sie sie regelmäßig zu Weiterbildungs-Seminaren. Das deprimiert die praktisch arbeitenden Kollegen.
- 93) Nach kurzer Zeit will niemand diese Theoretiker zurück im Team haben. Befördern Sie sie zum Projektleiter, damit sie weiterhin Artikel schreiben können.
- 94) Wenn ein Theoretiker kündigt, lamentieren Sie vor versammelter Mannschaft über die verlorenen Investitionen in Weiterbildung.

Optimieren Sie **Mitarbeitergespräche.**

- 95) Setzen Sie ein Mitarbeitergespräch (Jahres- oder Feedback-Gespräch) immer überraschend für die nächste Woche an.
- 96) Verschieben Sie es dann maximal 12 Stunden vorher um mindestens zwei Monate.
- 97) Führen Sie das Gespräch an Ihrem Schreibtisch. Damit die Hierarchie klar ist. Lassen Sie die Tür offen. Denn an den Kritikpunkten können sich die Kollegen ein Beispiel nehmen.
- 98) Erlauben Sie Telefonanrufe und andere Störungen. Schließlich hätten Sie Wichtigeres zu tun!
- 99) Geben Sie Ihre Einschätzung der Leistung und die Kritikpunkte erst unmittelbar im Gespräch bekannt. Keine Vorwarnungen!
- 100) Noch einfacher ist es, das Gespräch kurzfristig abzusagen und durch eine E-Mail zu ersetzen.

Achten Sie auf Ihre **Aussprache**.

- 101) Wann immer Sie über anstehende Projekte sprechen, erwähnen Sie die Worte „Stellen“, „Stellenplan“, „Produktivität“, „Auslastung“, „Fehlerraten“ und „Budget“ möglichst häufig.
- 102) Teilen Sie regelmäßig allen mit, wenn irgendwo von „Stellenabbau“ zu lesen war.
- 103) Verlangen Sie für jede „Stelle“ eine adäquate Qualifikation entsprechend der „Stellenbeschreibung“ für Neueinstellungen. Damit haben Sie auch das Problem der Über-40-jährigen gelöst.
- 104) Sprechen Sie nicht von „Personal“, „Personalentwicklung“, „Personentagen“, sondern nur noch von „HR“, „Ressourcen“ und „Leistungseinheiten“.
- 105) „Soziale Kompetenz“ ist irrelevant, solange es jeden Monat Fortschritte in der technologischen Entwicklung gibt.