

----- Boehm'sches Spiralmodell -----

Die Hauptidee des Boehm'sches Spiralmodells ist es das größte Risiko zu beseitigen. So wird jeder Zyklus des Wasserfall-Modells iterativ mehrfach durchlaufen. Dadurch nähert dich das Projekt über einen längeren Zeitraum den jeweiligen Zielen an. Hierbei finden vermehrt Risikoanalysen wie Verifikationen und Validationen statt.

Da die Phasen iterativ durchlaufen werden, kann man das Boehm'sches Spiralmodell als iterative Vorgehensweise beschreiben. Jedoch ist nach einem Zyklus die Phase abgeschlossen, also ein Inkrement, beispielsweise ein Prototyp, fertig. Dadurch vermischt es diese beiden Vorgehensweisen miteinander

Dokumentation meiner Nachforschung:
<!https://de.wikipedia.org/wiki/Spiralmodell#/> [15.10.2021]

----- Veränderte Anforderungen -----

a)

weinen?
Das Projekt als gescheitert ansehen / stornieren?
Den Kunden erschießen?

Bleibt dir ja nicht viel übrig, nicht wahr?

In der Realität würde man im "quick and dirty" style Funktionen rauslöschen, merken, dass man sie aus komplett nicht nachvollziehbaren Gründen noch benötigt, code Leichen in Kauf nehmen und die vorhandene Basis komplett zweckentfremden. Zusätzlich würde man komplett planlos, gar wahrlos, Funktionen hinzufügen und dem Kunden ein unsauberes unfertiges Projekt an den Kopf schmeißen. Nach diesem "Erfolg" sollte der Kunde das Unternehmen am besten meiden (spätesten, wenn er von der vorhanden-sein des Wasserfall-Modells hört).

b)

Man kann hoffen, dass das Projekt noch nicht weit vorangeschritten ist, und man die fehlerhafte Abzweigung rechtzeitig entdeckt und "geradebiegen" kann. Oh wait, "Einem Kunden fällt nach Zweidritteln erfolgreicher Projektlaufzeit ein, [...]" Na dann: siehe a)

c)

Man kann hoffen, dass die Funktion in einem eigenen Inkrement entwickelt wurde, welches komplett unabhängig von den anderen funktioniert. Ist unwahrscheinlich, aber hey, wenigstens ein bisschen Hoffnung bleibt. Dieses Inkrement kann gelöscht werden und weitere Inkremente werden "hinten" an die Ablaufstruktur angefügt. So kann man ein halbwegs sauberes Projekt erfolgreich beenden und den Kunden von fortan meiden. Sollte dies nicht so glücklich voranschreiten: siehe a)

d)

Im Zweifel lassen sich mehrere Sprint-Teams dafür missbrauchen die aktuell "fehlerhaften" Funktionalitäten zu beseitigen, sowie die neu gewünschten zu implementieren. Falls nicht: siehe c)

Auf jeden Fall werden in eigenen solcher Fälle viele Teamleiter schwitzen sowie viele viele Überstunden entstehen.

TL;DR:

@Kunden: Überlegt euch frühzeitig was ihr genau wollt,

@Teamleiter: Seid auf spontane und wechselfreudige Kunden vorbereitet
und @everyone: benutzt bitte nicht das Wasserfall-Modell!

Und meidet solch sprunghafte Kunden!

----- SCRUM -----

a)

Developer, Product Owner, Scrum Master

b)

<= 8 hours

c)

Daily Scrum)	= 15 min
Sprint)	<= 1 month
Sprint Retrospektive)	<= 3 hours
Sprint Review)	<= 4 hours

d)

Zu viel Vertrauen in- und Verantwortung für- die Mitarbeitenden.

Da man nur in einem Sprint gleichzeitig sein kann, wird Fachpersonal für eine spezifische Aufgabe "verschwendet".

Planung ist gut; zu viel Planung kann einem im Weg stehen -> SCRUM ist offensichtlich zu viel.