

## Lessons Learned aus einem agilen Festpreisprojekt

Ein Festpreisprojekt mit Scrum umsetzen und dabei einen definierten Leistungsumfang bis zu einer fixen Deadline in höchster Qualität unter Einhaltung des geschätzten Aufwands abliefern? Wir haben diese Quadratur des Kreises versucht. Ist es uns gelungen? Fast. Haben wir fürs nächste Mal etwas gelernt? Ohja, jede Menge. Und gerne teilen wir diese Erkenntnisse hier mit Ihnen!

### 7 Personen. 10 Sprints. Ein Produkt.

Anfang 2017 beauftragte ein guter Kunde Software Quality Lab damit, ein Testframework für eines seiner Produkte zu entwickeln. Unser Projektteam sollte dabei eng mit dem Team des Kunden zusammenarbeiten, Produkt und Testframework zeitgleich hochgezogen werden. Als gemeinsame Vorgehensweise wurde Scrum ausgewählt, da die Anforderungen zu Beginn noch nicht völlig klar spezifiziert waren und Scrum durch die fixe Taktung in Sprints und die regelmäßigen Planning und Review Meetings die Synchronisation der Teams erleichtern würde.

Von Anfang an war Qualität ein wichtiger Grundpfeiler des Projektes. Entsprechend war von Anfang an der Tester bei allen Meetings dabei und bekam den besten Platz im Projektbüro. Qualität ist aber nicht Aufgabe des Testers alleine. Qualität kann man nicht am Ende reintesten, sie muss von Beginn an entwickelt werden. Entsprechend arbeiteten in unserem Projekt Entwickler und Tester eng zusammen und achteten gemeinsam auf kompromisslos hohe Qualität.

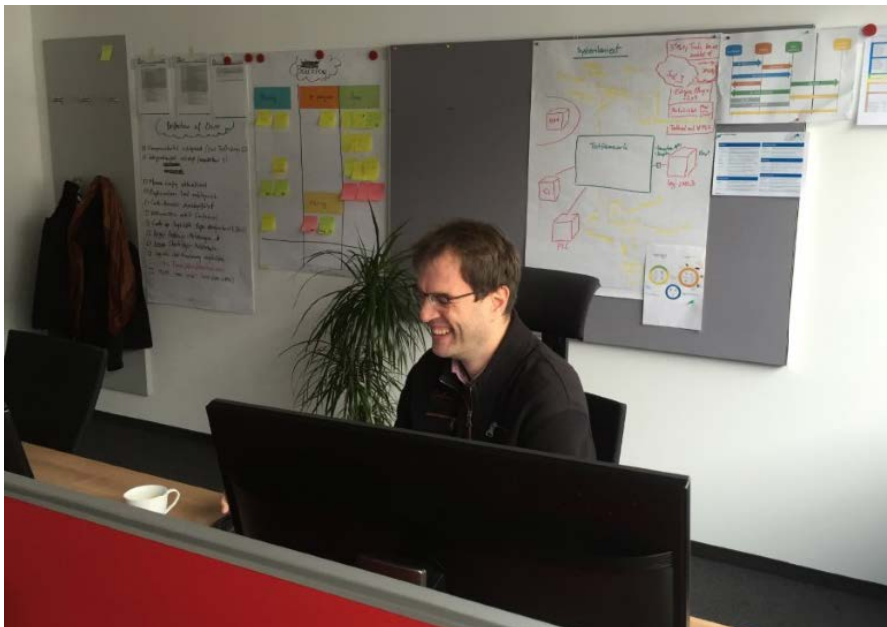


Abbildung 1 – Den besten Platz im Projektbüro bekam unser Tester. Höchste Qualität ist ein unverrückbares Ziel in agilen Projekten.

Was wir in diesem Projekt noch alles gelernt haben, was gut funktioniert hat und worauf wir stolz sind und was weniger gut funktioniert hat und wir das nächste Mal besser machen werden, lesen Sie auf den folgenden Seiten.



Agiler Festpreis –  
Ein Widerspruch?

Ein „Oxymoron“ ist ein Wort, das einen Widerspruch in sich darstellt. „Agiler Festpreis“ ist ein solches Oxymoron. Agil bedeutet beweglich und flexibel, ein Festpreis hingegen bezeichnet etwas Fixes, Festgeschriebenes.

Aus wirtschaftlicher Sicht ist die Forderung nach einem Festpreis verständlich. Budgets müssen geplant, beantragt und freigegeben werden. Abhängige Folgeaktivitäten müssen eingetaktet werden. Ein Festpreis gibt dabei das Gefühl der Kontrolle. Immer wieder erleben wir, dass beim Auftraggeber Erfahrung mit und Wissen über agile Vorgehensweisen und deren grundlegend andere Art an Projekte heranzugehen fehlt. Man kann sich einfach nicht vorstellen, wie es ohne fester Vereinbarung des Leistungsumfangs gehen kann. Das würde ja hohes Vertrauen des Auftraggebers in den Auftragnehmer voraussetzen, dass dieser auch ohne exakte Vorab-Definition etwas Wertvolles liefern wird.

Auch wenn manche so tun, meiner Meinung nach sind agile Methoden kein Zaubermittel für die Lösung des Festpreisdilemmas. Pflichtenforderungen (in agilen Vorgehensmodellen als MVP – Minimum Valuable Produkt, bezeichnet), gibt es in agilen Projekten nach wie vor. Technische Risiken sind darin ebenso vorhanden und müssen gelöst werden.

Der Vorteil, den agile Methoden bieten, liegt in einer höheren Flexibilität im Umgang mit Änderungen und einer früheren Sichtbarkeit von Problemen. Agile Projekte sind eher dem Time and Material Konzept verwandt. Aber auch da bleiben Pflichtenforderungen verpflichtend und nicht alles kann man rausnehmen oder anders machen. Agile Methoden sind eben leider auch kein Zaubermittel.

#### Markus Unterauer

Head of Consulting, Trainer

Der Quality Knowledge Letter ist ein periodisches Informationsmedium von Software Quality Lab und dessen Partnern mit dem Schwerpunkt SW-Qualität.

Weitere Infos zu diesem und anderen Themen finden Sie auf [www.software-quality-lab.com](http://www.software-quality-lab.com)

## Projekt-Setup

Die tatsächliche Arbeit am Projekt starteten wir für unseren Teil, das Testframework, im Mai 2017. Die Fertigstellung war für September geplant, was wir auch (fast) genau schafften. Der Gesamtaufwand war mit 200 Personentagen geschätzt worden.

Es gab im Gesamtprojekt zwei Teams, eines beim Kunden und eines bei uns. Unser Team bestand aus 7 Personen:

- ☑ 2 Entwickler
- ☑ 1 Tester
- ☑ 1 Architekt / Chef-Entwickler
- ☑ 1 Product Owner
- ☑ 1 Scrum Master (Teilzeit)
- ☑ 1 Controller (Teilzeit)

Der Best-Practice der Co-Location folgend, bezog unser Team ein Projektbüro, welches wir für die Dauer des Projektes nach unseren Bedürfnissen als „Visual Workspace“ einrichteten.



Abbildung 2 – Unser „Visual Workspace“ hält Definition of Done, Architekturübersicht und Taskboard jederzeit im Blick

An den Wänden hingen Architekturübersichten, die Definition of Done, ein Taskboard für Impediments und organisatorische Tasks, ein Flipchart mit den wichtigsten Infos zum aktuellen Sprint und auch jede Story war auf einem Flipchart beschrieben und jederzeit für alle sichtbar.

In unserem Scrum Projekt vereinbarten wir 2-Wochen-Sprints. Planning und Review wurden gemeinsam mit dem Kunden durchgeführt, Daily Standups und Retrospektiven machten wir in unserem Team. Da einige Teammitglieder in den ersten Monaten noch andere Projekte abzuwickeln hatten, legten wir in dieser Zeit Donnerstag und Freitag als primäre Projekttage fest, die sich alle für das Projekt freischaufeln sollten, damit das ganze Team konzentriert arbeiten kann.

## Projektverlauf

Die ersten Monate im Projekt waren stark geprägt von Lernen, Aufsetzen der Infrastruktur und Architekturentscheidungen. Wir

mussten bei uns die komplexe Infrastruktur des Gesamtproduktes und die Entwicklungsumgebung dafür nachbauen. Ebenso mussten die entsprechenden Tools eingerichtet und abgestimmt werden:

- ☑ Eclipse (Entwicklungsumgebung)
- ☑ Git (Sourcecontrol)
- ☑ Gerrit (Code Review)
- ☑ Maven (Build)
- ☑ Jenkins (Continuous Integration)
- ☑ JIRA (Backlog, Task und Bug Management)
- ☑ HipChat (Chat und Videokonferenzen)
- ☑ Confluence (Spezifikation und Konzepte)
- ☑ JIRA, RCPTT (Testmanagement)
- ☑ BCS (Zeiterfassung)

Für die Zusammenarbeit mit dem Kunden gab es gemeinsame Instanzen von JIRA und Git. Neben der Infrastruktur mussten wir uns auch in das mächtige und komplexe Produkt unseres Kunden einarbeiten. Nachdem diese Basisarbeit getan war, konnten wir beginnen, die Architektur des Testframeworks zu planen, offene Fragen mit Evaluierungen und Spikes zu klären und technische Risiken mit Prototypen abzusichern. Die gesamte Start- und Rampup-Phase zog sich bis in den August hinein.

Parallel dazu startete die Umsetzung der Anforderungen und deren Tests. Implementierung, Testautomatisierung und Test einer Story wurden soweit als möglich im selben Sprint durchgeführt, Entwickler und Tester arbeiteten dafür eng zusammen.

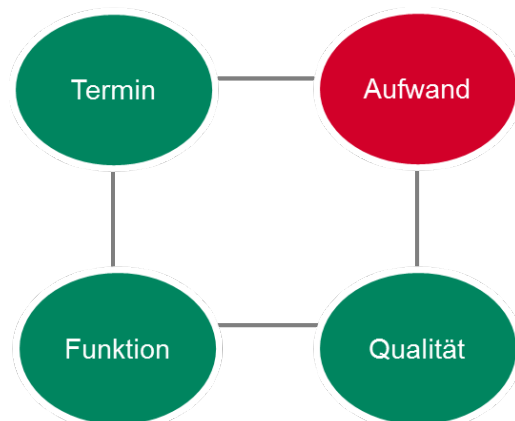


Abbildung 3 - Drei von vier der allgemeinen Projektziele des „Teufels-Quadrats“ konnten eingehalten werden

Bis Mitte September konnten wir alle Stories in sehr hoher Qualität abschließen. Lediglich drei Fehler waren uns durchgerutscht und auch die waren schnell behoben. Das Terminziel hatten wir also (fast) exakt einhalten können. Das Aufwandsziel mit 200 Personentagen hingegen schafften wir leider nicht. Für die Umsetzung der Pflichtenforderungen benötigten wir am Ende 240 Personentage.

Das Projekt wurde Ende September mit der Abnahme durch den Kunden abgeschlossen. Der Schlusspunkt auf unserer Seite war eine abschließende Projektretrospektive. Die folgenden Kapitel stellen die Ergebnisse dieser Retrospektive dar.

## Vollständiger Knowledge Letter Zugang

Wir freuen uns, dass Sie an diesem Thema Interesse haben und den Knowledge Letter von Software Quality Lab bis hierher gelesen haben.



**Dieser Knowledge Letter ist eine Vorschau (gekürzte Version des gesamten Artikels).**

Wenn Sie den ungekürzten Knowledge Letter lesen möchten, registrieren Sie sich bitte unter <http://www.software-quality-lab.com/download/knowledge-letter/anfrage-knowledge-letter/>

Sie erhalten nach der Registrierung vollen Zugang zu allen bisherigen Knowledge Letters von Software Quality Lab und erhalten automatisch künftige Knowledge Letter per E-Mail.

### Software Quality Days — Die größte Konferenz zum Thema „Software Qualität“ in Europa!



Besuchen Sie die Top-Konferenz mit allen Infos rund um Software Qualität.

Beste Qualität der Vorträge und Tutorials sowie eine Mischung aus praktischen und wissenschaftlichen Beiträgen machen die Software Quality Days zum Top-Event.

In den 3 praktischen Tracks werden anwendungsorientierte Vorträge präsentiert. Der wissenschaftliche Track zeigt Beiträge mit hohem Innovationsgrad und praktischer Anwendbarkeit, basierend auf Forschungsergebnissen. Im Solution Provider Forum präsentieren Aussteller ihre neuesten Tools mit Praxis-Beispielen.

Nähere Infos unter

[www.software-quality-days.com](http://www.software-quality-days.com)

