



Was ist der **deepseed**[®]-Ansatz?

Das Ziel des **deepseed**-Ansatzes besteht darin, ein Projekt in agiler Manier vorbildlich umzusetzen und dabei Mitarbeiter der Kundenorganisation auszubilden (durch gleichzeitiges Coaching und Schulung „on the job“).

Dabei ist das primäre Ziel nicht eine „Reinraum“-Implementierung agiler Prinzipien und Grundsätze, sondern ein Leben dieser Werte in der jeweiligen komplexen Situation und im normalen Projektumfeld des Kunden.

Dieser Ansatz verspricht, eine tiefgreifende Erfahrung mit einer agilen Vorgehensweise zu sammeln, die einerseits funktioniert, andererseits aber auch hilft Fehler, die normalerweise in einer erstmaligen Anwendung agiler Grundsätze begangen werden, zu vermeiden.

Inhalt

1 Ziele	2
2 Rahmenbedingungen	3
2.1 Projektwahl	3
2.2 Technische Voraussetzungen	4
2.3 Organisatorische Voraussetzungen	5
3 Projektmarketing	5
4 Vorgehen	6
4.1 Vorbereitungen	6
4.2 Durchführung	6
4.3 Monitoring	6
4.4 Abschluss	7

1 Ziele

Wir sehen die folgenden Ziele, die mit dem agilen Vorzeigeprojekt, erreicht werden sollten:

- „Learning on the job“
- „Spüren“ wie sich ein agiles Projekt anfühlt
- Trotzdem fertiges Produkt (keine Trockenübung) liefern
- Reale Probleme/Hindernisse identifizieren und angehen
- Direktes Verbessern der technischen Infrastruktur und des Toolings
- Beweisen, dass agiles Vorgehen auch im komplexen Umfeld des Kunden funktionieren kann
- Grundlagen für eigenes Leben agiler Prinzipien geschaffen (Kunden-Mitarbeiter sollen das Wissen selbstständig anderen weitergeben)
- Fachwissen aktiv beim Kunden verankern
- Aufzeigen funktionierender Möglichkeiten agile Projekte in ein noch nicht agiles Umfeld einzubetten
- Hinweise liefern, in welche Richtung sich die Organisation des Kunden noch entwickeln müsste, um als Ganzes agiler zu werden

2 Rahmenbedingungen

Die folgenden technischen und organisatorischen Voraussetzung müssen aus unserer Sicht erfüllt sein, um einen möglichst guten Start zu haben.

2.1 Projektwahl

Jedes Projekt ist anders gelagert und die richtige Wahl des Projektes ist wichtig, damit die Aussagekraft der beispielhaften Umsetzung gewahrt bleibt. Folgende Faktoren haben diesbezüglich eine Auswirkung:

- **Projektumfang:** Der Projektumfang sollte nicht zu klein und auch nicht zu gross sein. Eine Laufzeit von mindestens einem halben Jahr und weniger als einem Jahr ist empfehlenswert. Ist das Projekt zu gross ist die Chance auf einen erfolgreichen Abschluss gering (egal mit welcher Software-Engineering Methodik gearbeitet wird). Ist es zu klein, wird es nicht als repräsentativ wahrgenommen.
- **Projektart:** *Wartungsprojekte* sind einfacher planbar, aber dafür etwas weniger repräsentativ, weil nicht alle Aspekte eines neuen Projektes beleuchtet werden können. Auch besteht oft eine IT-Infrastruktur, die eventuell nicht den agilen Bedürfnissen gerecht wird oder in kurzer Zeit gerecht werden kann. Die Herausforderung liegt in diesem Fall, die bestehenden technischen und organisatorischen Gegebenheiten langsam in Richtung agile Vorgehensweise umzubauen.
Neue Projekte haben den Vorteil auch mit der benötigten Infrastruktur von Grund auf neu zu beginnen. Ein radikalerer Schnitt ist möglich. Ausserdem kann mit solch einem Projekt auch illustriert werden, wie man ein neues Projekt in agiler Manier aufsetzt. Ausserdem hat man die zusätzliche Herausforderung der klassischen Ungewissheit bezüglich Anforderungen und Wissen über den Lösungsraum und kann anschaulich zeigen, wie man damit umgehen kann.
- **Abhängigkeiten:** Es ist von Vorteil, wenn nicht zu viele Abhängigkeiten funktionaler und organisatorischer Natur zu anderen Projekten und Organisationseinheiten bestehen. Ansonsten läuft man Gefahr, dass man versucht die gesamte Organisation zu verändern. Ein Abgrenzung ist schwerer möglich.
Andererseits braucht es einige Abhängigkeiten zu Fremdsystem und unterschiedlichen Organisationseinheiten, damit das Projekt nicht als zu trivial und unrealistisch bezüglich der Aussagekraft eingeschätzt wird. Es soll ja schliesslich ein möglichst wirklichkeitsnahes Bild der Herausforderungen und der wirksamen Massnahmen zeichnen.

2.2 Technische Voraussetzungen

- Freier Zugang zu Entwicklungs-/Test-Umgebungen bis und mit Stufe Integration
- Erlaubnis Deployments automatisch durchzuführen (im Falle eines neuen Projektes bis und mit Produktion)
- Entweder wird die bestehende Softwareverteilungs-Struktur genutzt
 -und Schwachstellen werden aufgezeigt
 -und wir ändern langsam die Softwareverteilungs-Struktur (was auch andere Projekte tangieren kann)
- ...oder wir bauen eine eigene schlanke und agile Softwareverteilungs-Struktur auf
 - Dies ist möglich für ein komplett neues Projekt
 - Wir halten selbstverständlich bestehende Stufenmodelle (Entwicklung, Systemtest, Integration, Produktion) und andere allgemeine Vorgaben, wenn nötig ein.
 - ...und wir automatisieren Tests auf allen Stufen
- Einsatz eines einfachen Issue-Tracking-Tools
 - das Tool (bevorzugt JIRA/Confluence) wird durch **RED**pill aufgesetzt/konfiguriert
 - Vorteile:
 - Die Konfiguration bleibt so einfach und universell wie möglich
 - Unser „**deepseed** insight“ Dashboard kann eingesetzt werden, wenn JIRA zum Einsatz kommt (Anbindung anderer Tools, wie z.B. YouTrack kann implementiert werden)
 - Mitarbeiter des Kunden sammeln erste Erfahrungen mit dem neuen Tool
- Wir bringen bewährte Entwicklungs- und Testing-Tools mit (falls nicht bereits im Einsatz) und integrieren sie in die bestehende Tool-Landschaft (bei einem neuen Projekt, unter Berücksichtigung von Unternehmensrichtlinien)
- Kunden-weit zugängliche Dokumentation in einer zentralen Wiki-Instanz
- Java-basierter Software-Stack

2.3 Organisatorische Voraussetzungen

- Das Projekt muss mindestens eine Laufzeit von 6 Monaten haben, etwas länger (8 Monate) wäre besser
- Kunden-Entwickler müssen "mit ins Boot geholt" werden, nicht nur externe Entwickler
- Alle Rollen werden auch durch Kunden-Mitarbeiter besetzt (SM, PO, Team)
 - der ScrumMaster hat die Kompetenz Veränderungen durchzusetzen
- Alle Rollen werden durch **RED**pill-Mitarbeiter gecoacht
 - 2-3 **RED**pill-Mitarbeiter arbeiten als Entwickler im Team mit (bis max. 50% des Projektteams)
- Die Ausbildung der Kunden-Mitarbeiter kann ebenfalls durch **RED**pill erfolgen
- Co-Lokation des Teams (inkl. PO), idealerweise im gleichen Raum

3 Projektmarketing

- Das Projekt muss genügend Aufmerksamkeit haben
- Das Projekt muss genügend komplex, schwierig und mit Akzeptanzproblemen behaftet sein
- Auftritt im Intranet des Kunden (wie andere Projekte)
- Offenes Dashboard (an einer hoch frequentierten Stelle installiert)

4 Vorgehen

4.1 Vorbereitungen

- 360°-Analyse (Fachlich, technisch, organisatorisch, kulturell)
- Führung abholen - Vorstellung Projekt, Ziele (Erfahrung, Diagnose, Ausbildung, Produkt), etc.
- Auswahl der Mitarbeiter des Kunden (Bereitschaft mitzuwirken, technisch erfahren, breites Fachwissen der Kunden-Systeme, 100% verfügbar)
- Einführungs-Workshop für Team-Mitglieder des Kunden (Wie werden wir arbeiten?)
- Beteiligte Fachabteilungen einbeziehen
- Beteiligte Teams einbeziehen
- Setup der technischen Infrastruktur

4.2 Durchführung

- Festlegen der Team-Charter
- Festlegen der Definition of Done
- Cross-funktionales Team (gemäss Scrum), mit Einbezug des Fachs
- Ausbildung "on the job" der Team-Mitglieder des Kunden (Zeitbudget Reservation, ca. 10% der Projektzeit)
- Initiales Befüllen des Backlogs (erster Anforderungsworkshop)
- Erzeugte Releases (wenn möglich bei jedem Sprint) werden produktiv geschaltet (Agiler Grundsatz)
- Showcase der produktiven Releases (einmal pro Sprint/Release oder nach vorher vereinbarten Intervallen)

4.3 Monitoring

- Kennzahlen (automatisch) → Tool gestellt von REDpill („deepseed insight“-Dashboard)
- Resultate aus der Retrospektive werden publiziert (Hindernisse, Massnahmen)
- In regelmässigen Abständen: „Meet-the-Team“ (alle interessierten Mitarbeiter können das Entwicklungsteam interviewen bezüglich ihrer Erfahrungen und Herausforderungen)
- Blogs: PO, SM, Entwickler
- Regelmässige Peer-Review (innerhalb des Teams) → Tool-unterstützt (peerUp!)
- Regelmässige Projekt-Review (durch Stakeholder und beteiligte Abteilungen/Teams)

4.4 Abschluss

- Abschluss-Event
 - Rückblick
 - Lessons Learned
 - Ausblick, wie Agilität weiter ausgebaut werden könnte (teamübergreifend und eventuell auch unternehmensweit)

deepseed is powered by



Für weitere Informationen wenden Sie sich an:

REDpill GmbH
Eichenspesstrasse 5
6010 Kriens

Phone: 078 / 9232053
E-Mail: info@red-pill.ch