

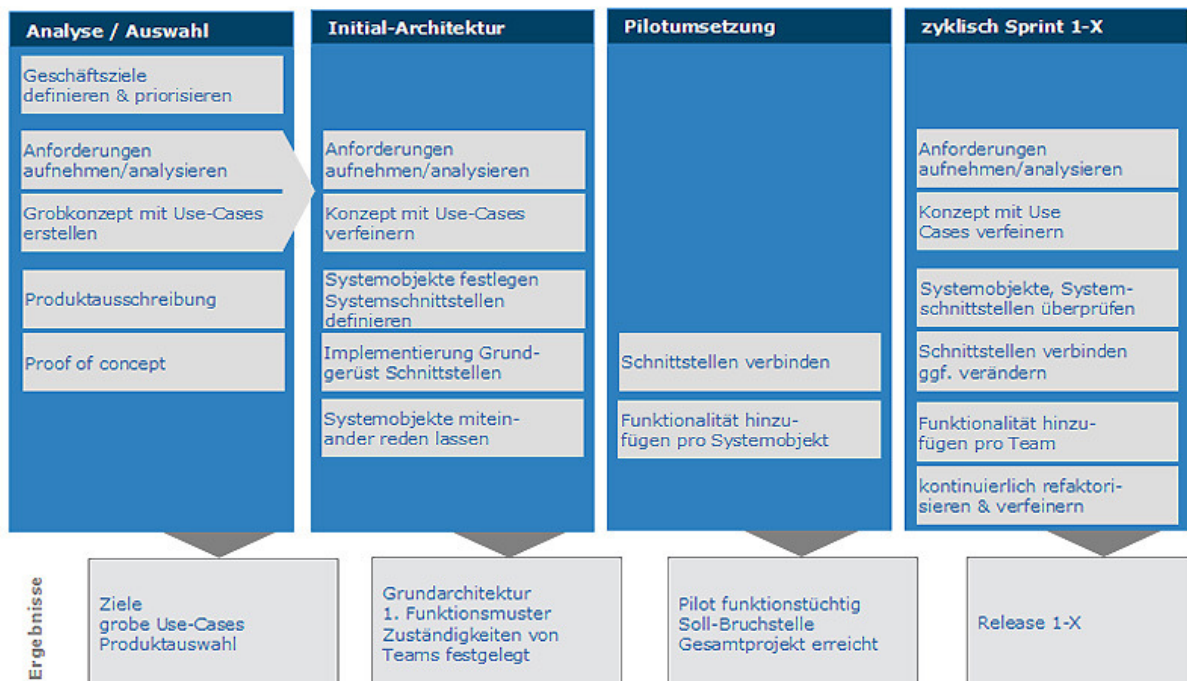
Das Nionex Projektvorgehen

Nionex realisiert IT-Lösungen, mit denen Unternehmen ihre eigene Wettbewerbsfähigkeit steigern. So, wie sich die Märkte verändern, in denen die Unternehmen agieren, ändern sich auch die Anforderungen an die IT-Lösungen, die zur Steigerung der Wettbewerbsfähigkeit benötigt werden. Verstärkt wird dieser Effekt noch dadurch, dass Nionex, als Experte für High Performance E-Business, IT-Lösungen im stetigen Wandel des Internet realisiert. Die Veränderung von Anforderungen in Unternehmen werden also neben Markt- und Organisationsveränderungen auch durch neue technische Möglichkeiten getrieben.

Klassische Vorgehensmodelle, die meist auf einer vollständigen Vorabspezifikation der Anforderungen beruhen, berücksichtigen aus der Erfahrung von Nionex die Tatsache, dass Softwareentwicklung stetiger Veränderungen unterliegt, nicht. Unternehmen erhalten nach den klassischen Vorgehensmodellen im günstigen Fall das Ergebnis, das zu Anfang der Projektlaufzeit geplant wurde, jedoch meist nicht das, was man zum Zeitpunkt der Einführung eigentlich benötigt. Flexibel auf Änderungen zu reagieren, ist jedoch in diesen Vorgehen nicht vorgesehen.

Nionex setzt in Softwareprojekten auf so genannte agile Methoden. Aus zahlreichen Softwareprojekten gereifte und als Best Practices bezeichnete Verfahren werden bei Nionex in den unterschiedlichen Phasen eines Projekts eingesetzt, um Softwareprojekte erfolgreich und mit dem vom beauftragenden Unternehmen zum Zeitpunkt der Einführung benötigten Ergebnis umzusetzen. Ein wesentliches Merkmal der Umsetzung ist, dass bereits zu einem frühen Zeitpunkt im Projekt lauffähige Ergebnisse entstehen, die mit dem beauftragenden Unternehmen für die weitere Umsetzung abgestimmt werden.

In der nachstehenden Ansicht werden die wesentlichen Projektphasen dargestellt. Nionex strebt dort, wo es sinnvoll möglich ist, an, Standardsoftwareprodukte in der Umsetzung einzusetzen. Je nach Projektausprägung kann dies Content Management, Portal sowie erforderliche Middleware Software sein, die von Nionex für die zu entwickelnde Lösung miteinander integriert werden. Um frühzeitig relevante Ergebnisse zu produzieren, findet daher gegebenenfalls eine Auswahl infrage kommender Softwareprodukte statt.



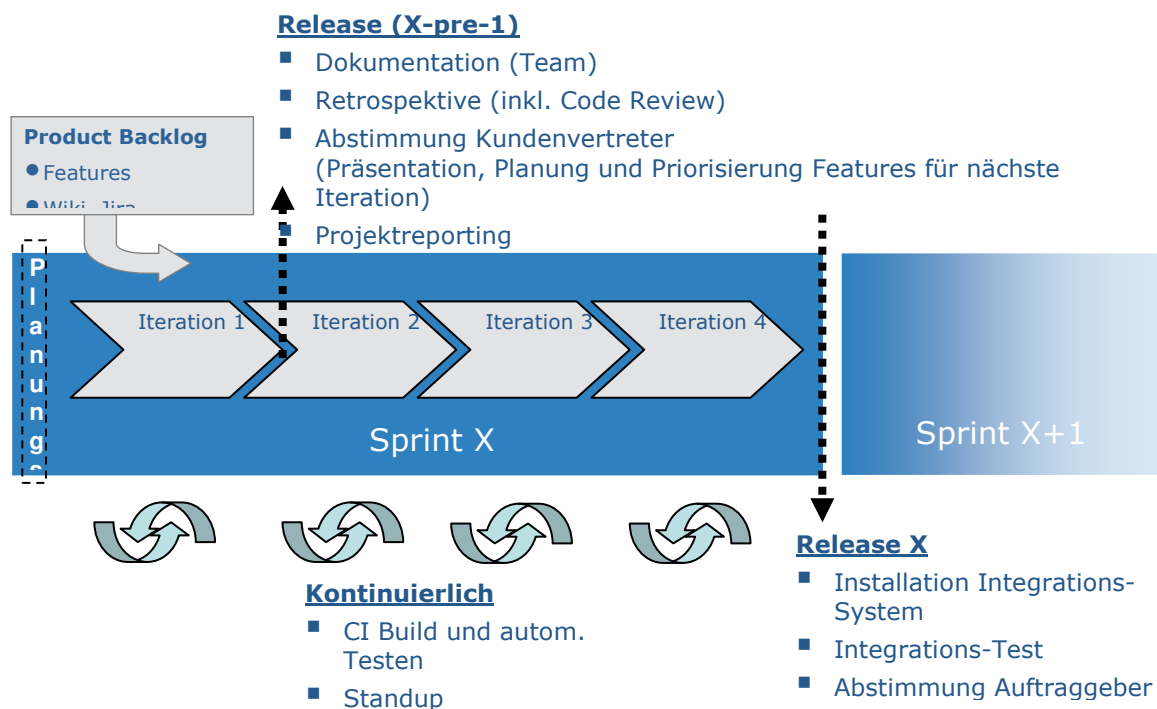
Die Anfangsphase des Projekts über die oben dargestellten Phasen Analyse, Initial-Architektur und Pilotumsetzung finden mit definierter Konzeption statt. Bereits in diesen Phasen werden aber agile Methoden eingesetzt, mit denen insbesondere das Zusammenspiel des Entwicklungsteams optimiert

wird. So finden tägliche 15 minütige Standup Meetings statt, um in kurzer Form das Team auf den gleichen Informationsstand über den Projektfortschritt zu bringen. Die Analyse wird außerdem durch System Architekten und Entwickler durchgeführt, die im späteren Projektverlauf für die Umsetzung verantwortlich sind. Durch den im Vergleich zu klassischen Entwicklungsprozessen frühen Start von Entwicklungstätigkeit wird bereits hier die Entwicklungsumgebung aufgebaut mit dem Ziel, kontinuierlich eine in sich konsistente und korrekte Version der in der Entwicklung befindlichen Software zu haben. Wir sprechen hier von Continuous Integration und automatisiertem Testen. Projektbegleitende Dokumentation findet bei Nionex auf Basis moderner Wiki Technologie in dafür eigens geschaffenen virtuellen Projekträumen statt.

Auf diese Weise entsteht ein erster funktionstüchtiger Pilot, der für das Unternehmen eine aussagekräftige Entscheidungsgrundlage für den Erfolg des Projekts bietet. Gleichzeitig wird an dem Pilot das Zusammenspiel erforderlicher Softwarekomponenten verprobt. Hier können zu einem frühen Zeitpunkt und vor Beschaffung der Softwarekomponenten noch wesentliche Projektentscheidungen getroffen werden.

Der Umfang des Projekts wird im Product Backlog beschrieben. Dieses besteht aus den umzusetzenden Arbeitspaketen in Form von Features, Aufgaben und Fehlermeldungen. Die Verwaltung des Product Backlog erfolgt über ein webbasierten Tool in Kombination mit einer Wiki-Dokumentation.

Die Umsetzung erfolgt in mehreren Iterationen. Vier bis sechs Iterationen werden zu einem Sprint zusammengefasst. Zu Beginn eines Sprint findet eine kurze Planungsphase statt, in der die umzusetzenden Arbeitspakete aus dem sog. Product Backlog entnommen und für die Umsetzung eingeplant werden. Ein regelmässiges Projektcontrolling unterstützt dabei den Projektleiter bei der objektiven Bewertung des Ressourceneinsatzes. Ergebnis eines Sprint ist ein lauffähiges Softwarerelease, das mit den gleichen Qualitätsanforderungen wie das endgültige Release erstellt wurde. Je nach Projektart kann dieses Release mit den bis dahin umgesetzten Funktionen bereits eingesetzt werden. Im Regelfall dient es der Abstimmung mit dem Auftraggeber um die Planung für den darauffolgenden Sprint durchzuführen.



Die wichtigsten Tools und Best Practices bei Nionex:

Product Backlog

- Aufgabenplanung über webbasierte Tools (JIRA)
- Wiki-Dokumentation der Arbeitspakete, Systemarchitektur, Systemkomponenten (Confluence)
- Ressourcen- und Terminplanung auf Basis der erzielten Ergebnisse

Teamkommunikation

- Tägliche Standup Meetings (ein Teammeeting, das im stehen abgehalten wird)
- Wöchentliche Retrospektive mit offener Feedbackrunde
- Regelmäßige Code Reviews

Qualitätssicherung und Dokumentation

- Automatisiertes Testen – von Anfang an (JUnit und Webtest)
- Wöchentliche Dokumentation der erzielten Ergebnisse (Confluence Wiki)
- Kontinuierliche (mehrmals tägliche) Integrationstests mit automatisiertem Durchlauf aller Testfälle (Cruisecontrol Continuous Integration)
- Objektive Bewertung des Quellcodes anhand von Metriken (Checkstyle, Cobertura),

Kunden Feedback

- Regelmäßige Bereitstellung eines lauffähigen Softwarerelease (pro Sprint)
- Gemeinsame Bewertung des Zwischenstands und Priorisierung der Aufgabenpakete für den nächsten Sprint
- Direkter Zugriff auf das Product Backlog für Anmerkungen, Planänderungen oder Fehlerreports (JIRA)