

Anforderungen mit Hilfe von Prototypen erheben

Bereich
Anforderungen

Aktivität
(Kunden-)Anforderungen erheben

Ziele

- Anforderungen systematisch klären, verfeinern, kommunizieren und validieren
- Entdecken von Anforderungen, die (noch) nicht in Anforderungsspezifikationen enthalten sind
- Erzeugen eines gemeinsamen Verständnisses des zu entwickelnden Produkts

schnell ★★
durchführbar ★

einfach ★★
durchführbar ★

agil einsetzbar ★★
★

Motivation/Problemstellung

Oft sieht man sich der Herausforderung gegenüber, Anforderungen an ein Software-System erheben zu müssen, die jedoch von Stakeholdern nicht oder nicht detailliert genug geäußert werden. Um die Stakeholder zu stimulieren, sich konkretere Gedanken über ihre Anforderungen an das System zu machen, können Prototypen eingesetzt werden. Prototypen stellen eine visuelle Repräsentation bereits erhobener, interpretierter und implementierter Anforderungen dar, die häufig zu Äußerungen neuer Anforderungen führen, über die Stakeholder vorher nicht nachgedacht oder implizit als gegeben angenommen haben. Weiterhin wird durch den Einsatz von Prototypen ein gemeinsames Verständnis des sich in der Entwicklung befindenden Systems erzeugt und kostenintensive Beseitigungen zu spät entdeckter Konflikte können frühzeitig vermieden werden. Alternative Lösungen für entdeckte Konflikte können weiterhin prototypisch implementiert und basierend darauf diskutiert werden. Schließlich können die Machbarkeit von Lösungsideen geprüft und die Gestaltungsphase des Software Engineerings frühzeitig gestartet werden.

Kurzbeschreibung

Um die aufgeführten Ziele zu erreichen, kann ein Prototyping Prozess durchgeführt werden, der aus drei Phasen besteht: zunächst wird ein geeigneter Prototyp konzipiert, dieses Konzept wird in der zweiten Phase implementiert und schließlich in der dritten Phase durch Stakeholder evaluiert. Diese drei Phasen werden iterativ durchlaufen (vgl. Abbildung 1). Somit wird ermöglicht, dass kontinuierlich bestimmte Mengen von Anforderungen erhoben, diskutiert, verfeinert und evaluiert werden, bis schließlich das

Gesamtbild der Art und Weise, wie welche Anforderungen umgesetzt werden sollen, vervollständigt wird.

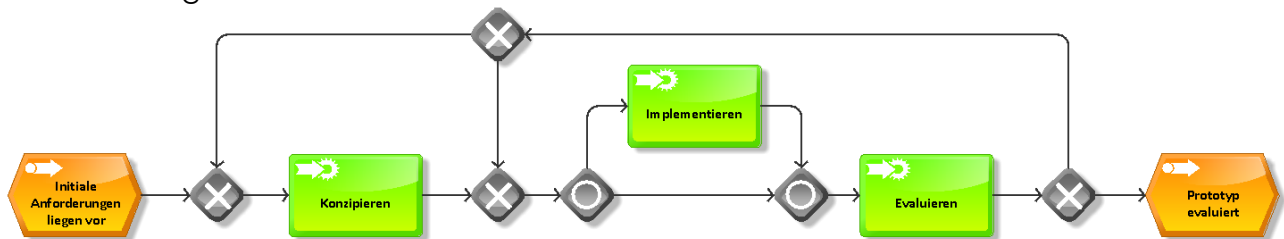


Abbildung 1: Prozess zur Erhebung von Anforderungen mit Hilfe von Prototypen

Input

- Initiale Menge von Anforderungen
- Personas

Output

- Validierte Prototypen
- Validierte Liste von Anforderungen
- Liste von Lösungsideen für bestimmte Probleme
- Nach Schweregraden sortierte Liste gefundener Probleme

Rahmenbedingungen

Ausführender

Anforderungsingenieur, Designer

Werkzeuge, Hilfsmittel

Stifte und Papier, Prototyping Tools (Antetype, Axure, Balsamiq Mockups, etc.)

Vorkenntnisse/Erfahrungen

-

Ort/Umgebung

Büro, Kundenworkshops

Weitere Teilnehmer

Domänenexperten, Endnutzer

Voraussichtliche Dauer

Fortlaufend während der Anforderungsphase, abhängig von der Anzahl zu evaluierender Anforderungen und zu erstellender Prototypen

Vorgehensweise

Vorbereitung

Um mit Hilfe von Prototypen Anforderungen zu erheben, zu klären, zu verfeinern und zu validieren, müssen bestimmte Informationen vorliegen, um die Aktivitäten in den einzelnen Phasen des zu durchlaufenden Prozesses erfolgreich durchführen zu können: Identifizieren Sie zunächst die Zielgruppe, die das finale System nutzen soll und daher auch die Prototypen evaluieren soll. Wählen Sie aus dieser Zielgruppe Evaluatoren aus und laden Sie sie zu einem gemeinsamen Workshop ein. Um diese Auswahl treffen zu können, ist es hilfreich, eine Persona eines Endnutzers zu erstellen, deren Eigenschaften dann die realen Evaluatoren nah kommen sollten. Weiterhin müssen Ihnen sowohl die Ist-Situation inklusive der zu lösenden Probleme, als auch die Soll-Situation zur

Durchführung bestimmter Aufgaben mit Hilfe des zu entwickelnden Systems bekannt sein. Schließlich muss Ihnen eine Liste von Anforderungen vorliegen, die von realen zukünftigen Nutzern des Systems erhoben wurden und geklärt, verfeinert oder evaluiert werden sollen. Hierzu müssen in jedem Fall die Anforderungen an kritische Bestandteile des Systems und die Hauptsystemfunktionen inklusive der Daten, die ausgetauscht werden sollen, und des Formats dieser Daten gehören. Der Entstehungsprozess der Anforderungen muss Ihnen in einer Weise dokumentiert vorliegen, dass alle Stakeholder die getroffenen Entscheidungen durch Angaben der entsprechenden Begründungen nachvollziehen und die erhobenen Anforderungen verstehen können.

Durchführung

Legen Sie pro Sprint fest, welche Anforderungen prototypisch umgesetzt werden sollen. Definieren Sie anschließend auf Basis dieser Liste von Anforderungen Hypothesen, die durch eine Evaluation verifiziert oder negiert werden sollen. Dazu gehört auch die Festlegung einer Zeit zur Erfüllung bestimmter Aufgaben mit Hilfe des Prototyps des Systems. Definieren Sie basierend auf den Hypothesen wiederum die Ziele des Prototyps. Bedenken Sie dabei, dass ein Prototyp immer nur ein Ziel oder wenige Ziele verfolgen kann, beispielsweise das Evaluieren der Benutzungsschnittstelle oder die Evaluation der technischen Umsetzbarkeit bestimmter Lösungsideen. Wie später die Evaluation der Prototypen im Detail stattfinden wird, hängt von der Definition der Ziele der Prototypen ab. Abhängig von den definierten Zielen können Sie nun aber vor allem eine verfeinerte Auswahl zu implementierender Anforderungen aus der zu Beginn festgelegten Liste von Anforderungen treffen. Achten Sie dabei darauf, dass der Umfang des Prototyps so weit eingeschränkt wird, dass Nutzer in der Lage sind, sich auf diejenigen Aspekte zu konzentrieren, die evaluiert werden sollen. Dadurch erzielen Sie nicht nur aussagekräftigere Evaluationsergebnisse, sondern reduzieren auch die Kosten und die benötigte Zeit zur Erstellung der Prototypen. Die Festlegung des Umfangs lässt sich durch den Einsatz von Szenarien erleichtern. Vor allem sollten Sie Anforderungen implementieren, die noch unklar sind, jedoch nicht solche, die bereits valide sind. Denken Sie auch immer daran, nicht-funktionale Anforderungen zu implementieren und berücksichtigen Sie existierende Style Guides und Interaction Guidelines.

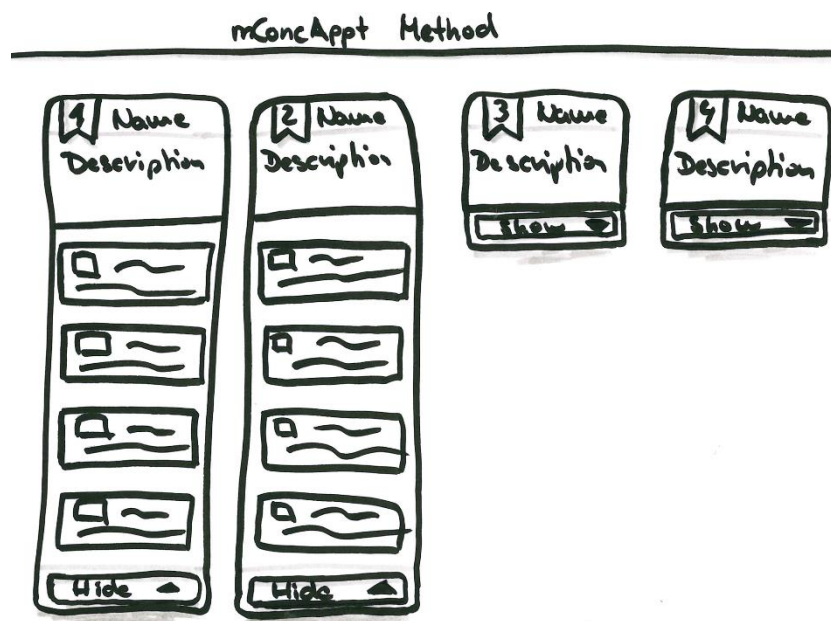


Abbildung 2: Low-Fidelity Prototyp

Nun kommt es darauf an, den richtigen Typ von Prototyp (Papier-Mockup, Simulation, vollfunktionaler Prototyp, etc.) abhängig von den spezifizierten Zielen zu wählen (vgl. Abbildung 2 Low-Fidelity Prototyp vs. Abbildung 3 High-Fidelity Prototyp). Diese Entscheidung hängt neben den Zielen auch von dem verfügbaren Budget, der verfügbaren Zeit, von der Komplexität des Systems, den Wünschen des Auftraggebers und den Vorlieben der Prototyp Ersteller ab. Generell sollten Mockups aber eher in frühen und vollfunktionale Prototypen eher in späteren Anforderungserhebungsphasen eingesetzt werden.

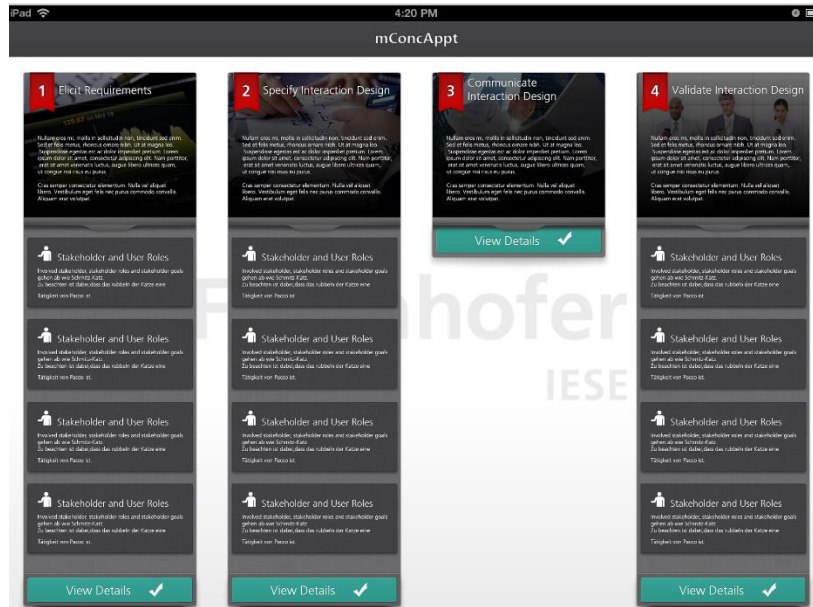


Abbildung 3: High-Fidelity Prototyp

Wurde der geeignete Typ von Prototyp gefunden, konzipieren Sie den Prototyp zunächst. Leiten Sie dafür die Hauptfunktionalität des Prototyps von der Soll-Situation ab. Erstellen Sie für diese Hauptfunktionalität einen Interaktionsfluss und Anwendungsfälle und verknüpfen Sie diese mit geeigneten Screen Flows, Mockups und Wireframes. Beginnen Sie stets mit dem Standard-Interaktionsfluss, der idealen Interaktion mit dem Prototyp und gestalten Sie dann Interaktionsflüsse in Fällen von Ausnahmen und Störungen, und erlauben Sie auch alternative Interaktionsflüsse.

Lassen Sie den Prototyp immer intern evaluieren (beispielsweise durch einen Experten Walkthrough), um Fehler innerhalb des Prototyps zu finden und zu lösen, bevor der Prototyp implementiert wird. Hierdurch können hohe Kosten gespart werden, die entstünden, wenn Fehler erst spät gefunden würden. Kommunizieren Sie alle Änderungen an dem Prototyp, die sich durch die interne Evaluation ergeben, an alle Stakeholder, die an dem Software-Entwicklungsprozess beteiligt sind. Lassen Sie den fehlerbereinigten Prototyp erneut intern evaluieren, bis keine Fehler mehr gefunden werden, die die externe Evaluation beeinträchtigen könnten. Anschließend kann entweder ein funktionaler, elektronischer Prototyp erstellt oder der vorliegende nicht-funktionale Prototyp extern evaluiert werden.

Entwickeln Sie in der zweiten Phase des Prozesses (Prototyp implementieren) einen funktionalen elektronischen Prototyp auf Basis der Anwendungsfälle und berücksichtigen Sie dabei nur diejenigen Attribute, die zur Implementierung der ausgewählten und durch den Prototyp zu prüfenden Anforderungen benötigt werden. Gehen Sie bei der Implementierung des Prototyps so vor, dass Sie zunächst eine Baseline implementieren, die auf möglichst sicheren Lösungen für ebenfalls sichere

Anforderungen basiert. Mit dieser Baseline als Grundlage können Sie leicht mehrere alternative Mockups (Konzepte) geeigneter Lösungsideen erstellen, um unsichere Anforderungen zu klären, zu verfeinern und allgemein zu diskutieren. Die Baseline wird lediglich um validierte Anforderungen bzw. deren Gestaltungslösungen erweitert, jedoch ansonsten nicht geändert.

Führen Sie auch für den implementierten Prototyp eine interne Evaluation durch, um etwaige Fehler zu finden und zu bereinigen, damit diese nicht bei der anschließenden Evaluation das Ergebnis negativ beeinflussen. Achten Sie darauf, dass die Antwortzeiten des Prototyps zwar realistisch, aber nicht zu lang sind, um Nutzer nicht zu langweilen oder zu verwirren und gleichzeitig eine Evaluation der Gebrauchstauglichkeit des Prototyps zu erlauben.

Lassen Sie den Prototyp (egal ob es sich um einen konzipierten oder implementierten Prototypen handelt) schließlich durch echte Endnutzer des späteren Systems an deren Arbeitsplätzen evaluieren. Lassen Sie die Evaluation stets von einem geübten Prototyp-Designer begleiten. Ziel dieser externen Evaluation ist das Bestätigen oder Negieren der Hypothesen, die Sie bereits zu Beginn der Überlegungen bezüglich des zu entwickelnden Prototyps formuliert haben. Lassen Sie die Endnutzer für die Evaluation möglichst realistische Aufgaben mit dem Prototyp durchführen und versetzen Sie sie in realistische Nutzungsszenarien, in denen wenn möglich auch echte Daten verwendet werden.

Achten Sie darauf, dass Prototypen für eine Evaluation vorbereitet werden müssen: nicht-funktionale Prototypen müssen sauber skizziert, gedruckt, sortiert oder in eine Anwendung überführt werden, in der sie klickbar werden (z.B. Microsoft PowerPoint); elektronische Prototypen müssen außerhalb der Entwicklungsumgebung ausgeführt und auf Zielgeräten installiert werden können.

Sorgen Sie für eine ungehemmte Atmosphäre während der Evaluation und führen Sie für die eigentliche Evaluation die folgenden vier Schritte durch:

1. Erläutern Sie den Nutzern zunächst die Gründe für die Evaluation (Evaluationsziele und Fokus) und erheben Sie persönliche Informationen der Nutzer durch den Einsatz eines Fragebogens, der individuell erstellt werden kann. Erläutern Sie den Nutzern auch, dass es sich um einen Prototyp handelt und nicht um ein finales System und ggf. warum der Prototyp nicht aussieht wie ein finales System.
2. Geben Sie den Nutzern etwas Zeit, um sich mit dem Prototyp und dessen Funktionalität vertraut zu machen, damit sie so mit dem Prototyp interagieren, wie sie es voraussichtlich auch mit dem finalen System tun würden.
3. Lassen Sie die Nutzer einzeln die Anwendungsfälle mit dem Prototyp durchspielen und den Nutzer laut äußern, was er gerade denkt und tut. Falls nötig, erinnern Sie ihn immer wieder daran, bitte zu kommentieren, was er tut, während er den Prototypen evaluiert. Vergleichen Sie währenddessen immer das erwartete mit dem beobachteten Verhalten und notieren Sie alle Abweichungen, ohne aber selbst zu kommentieren oder den Nutzer um weitere Erklärungen zu bitten. Sorgen Sie auch dafür, dass keine sonstigen Störungen der Interaktion des Nutzers mit dem Prototypen auftreten. Notieren Sie jedes beobachtete und geäußerte Problem des Nutzers sowie jede vermisste Funktion bzw. jeden geäußerten Wunsch. Notieren Sie zu jeder geäußerten oder beobachteten Auffälligkeit (sowohl negative als auch positive Auffälligkeiten), bei welchem Task und welchem Screen sie auftrat und beschreiben Sie, wie sie konkret entdeckt wurde. Achten Sie dabei besonders auf diejenigen Auffälligkeiten, die im Bezug zu den zu evaluierenden Anforderungen stehen.
Die Evaluation ist beendet, wenn alle geplanten Aufgaben zum Durchlaufen der Anwendungsfälle durchgeführt wurden.

4. Stellen Sie zum Abschluss der Evaluation allgemeine Fragen über die Meinung des Nutzers über den evaluierten Prototyp und diskutieren Sie die beobachteten und geäußerten Auffälligkeiten, insbesondere diejenigen mit Bezug zu den evaluierten Anforderungen. Erstellen Sie während dieser Diskussionen Mockups der gemeinsam mit dem Nutzer erarbeiteten Lösungen für notierte Probleme und klären und verfeinern Sie damit dessen Anforderungen. Achten Sie darauf, zu klären, ob es sich um Probleme handelt, die auch im späteren Betrieb des Systems auftreten würden oder ob das Problem nur aufgrund der Testsituation aufgefallen ist. Gehen Sie außerdem konkret auf Wünsche ein, die mit den evaluierten Anforderungen verwandt sind, von dem aktuellen Prototypen jedoch noch nicht erfüllt und somit als neue Anforderungen formuliert werden können. Die Diskussionen und damit die Evaluation enden mit der Erklärung des Nutzers, das der Prototyp unter Berücksichtigung der neu erstellten Mockups alle kritischen Aspekte der Anwendungsfälle erfolgreich abdeckt.

Nachbereitung/Auswertung

Bewerten Sie den Schweregrad jedes einzelnen gefundenen Problems nach der Durchführung der Evaluation. Berücksichtigen Sie dafür die Auswirkungen des Problems auf den Nutzer (z.B. Verwirrung, Verlorenheit), die Auftrittshäufigkeit des Problems sowie die Anzahl der Nutzer, die das Problem wahrnehmen. Strukturieren Sie neben diesen Schweregraden auch die übrigen Daten, die während der Evaluation erhoben wurden (Notizen, Aufzeichnungen, etc.). Prüfen Sie, ob zu jeder notierten Auffälligkeit auch die Aufgaben und Screens erfasst wurden, bei denen sie auftraten und ob zu jeder Auffälligkeit eine Beschreibung vorliegt, wie sie konkret auftrat. Führen Sie anschließend die Protokolle der Evaluationen durch einzelne Nutzer zu einer einzigen Liste zusammen. Sie erhalten somit eine Liste aller Auffälligkeiten mit Lösungsideen, die auf Basis ihrer Schweregrade sortiert werden kann. Mit Hilfe dieser Liste können Sie die Überarbeitungen der Prototypen planen. Führen Sie nach der Überarbeitung der Prototypen erneut eine Evaluation durch.

Gütekriterien/Empfehlungen

Es kann vorteilhaft sein, die eigentliche Evaluation des Prototyps durch einen realen Endnutzer durch Screen Capturing Tools aufzuzeichnen, zu filmen oder um Eye Tracking zu erweitern. Hierdurch erhalten Sie weitere Daten, deren Analyse aufschlussreich sein kann, wenn man sie mit den Notizen zu den eigenen Beobachtungen und Diskussionen mit dem Nutzer kombiniert.

Vermeiden Sie es, Mockups, die gemeinsam mit Endnutzern am Ende der Evaluation erstellt wurden, direkt in den Baseline-Prototypen zu übernehmen. Achten Sie vielmehr darauf, dass diese Mockups tatsächlich die Anforderungen erfüllen und denken Sie über weitere Lösungsmöglichkeiten nach, die zunächst wieder extern evaluiert werden müssen, bevor sie in den Baseline-Prototyp einfließen.

Risiken

Da es sich bei Prototypen um abstrakte Repräsentationen eines Systems handelt, besteht die Gefahr, dass Stakeholder Prototypen mit einer falschen Erwartungshaltung evaluieren. Stakeholder könnten beispielsweise denken, dass das finale System ebenso schnell wie der Prototyp entwickelt werden kann oder es sich bei dem Prototyp bereits um das finale System handelt. Weiterhin besteht die Gefahr, dass Stakeholder die limitierte Funktionalität eines Prototypen und dessen visuelle Repräsentation als gegeben und unveränderbar wahrnehmen. Dies führt dann zu Problemen, wenn Stakeholder mit

der Funktionalität und / oder der visuellen Gestaltung des Prototyps nicht zufrieden sind, aufgrund ihrer Ansicht, dass hier nichts mehr geändert werden kann, aber zu den Problemen schweigen.

Es sollte darauf geachtet werden, dass der Abstraktionsgrad eines Prototypen zu der Menge an Anforderungen passt, die er repräsentiert und dass klar kommuniziert wird, worum es bei der Evaluation eines bestimmten Prototyps geht: Evaluierende Stakeholder mit viel Expertise könnten sich ansonsten in Detaildiskussionen verwickeln, z.B. über das Aussehen von Schaltflächen, obwohl eigentlich das Bildschirm-Layout diskutiert werden sollte. Auf der anderen Seite können Prototypen ohne jegliches visuelles Design bei Stakeholdern mit geringem Abstraktionsvermögen dazu führen, dass sie nicht in Verbindung mit dem zu entwickelnden System gebracht werden können; die Evaluation und Erhebung bzw. Verfeinerung von Anforderung wird in diesem Fall schwierig. Für die Evaluation von Prototypen an den Arbeitsplätzen der Endnutzer muss die Genehmigung des Arbeitgebers eingeholt werden, eine nicht gegebene Genehmigung kann also die gesamte Evaluation gefährden oder zumindest beeinträchtigen, wenn die Evaluation an einem anderen Ort durchgeführt werden muss.

Einordnung in den agilen Referenzprozess

Mögliche Vorgänger

- Requirements Engineering\Funktionale Anforderungen erheben
- Requirements Engineering\Kundenanforderungen dokumentieren
- Requirements Engineering\Nichtfunktionale Anforderungen erheben
- Requirements Engineering\Produktvision erstellen
- Requirements Engineering\Systemkontext und -umfang festlegen
- User Experience\Kontextuelles Benutzerinterview durchführen
- User Experience\Produktphilosophie erstellen
- User Experience\Template-basierte UI-Konzeption

Mögliche Nachfolger

- Requirements Engineering\Anforderungen reviewen
- Requirements Engineering\Begründungen für Anforderungen dokumentieren
- Requirements Engineering\Kundenanforderungen in technische Anforderungen übertragen
- Requirements Engineering\Viele Augen sehen mehr
- User Experience\Informationsarchitektur erstellen
- User Experience\Template-basierte UI-Konzeption
- Architektur\Architekturlösungen im Team entwickeln
- Architektur\Grob- und Detailplanung bei der Implementierung nutzen

Mögliche Alternativen, verwandte Praktiken

- Requirements Engineering\Anforderungen reviewen
- Requirements Engineering\Anforderungen wiederverwenden
- Requirements Engineering\Experimentelles und exploratives Prototyping durchführen

- Requirements Engineering\Funktionale Anforderungen erheben
- Requirements Engineering\Kundenanforderungen dokumentieren
- Requirements Engineering\Nichtfunktionale Anforderungen erheben
- User Experience\Kontextuelles Benutzerinterview durchführen

Einordnung in das PQ4Agile-Qualitätsmodell

Funktionale Tauglichkeit, Effektivität, Effizienz, Nützlichkeit, kontextuelle Vollständigkeit

Schlagworte

Anforderungen, Prototypen, Design, Evaluation

Weiterführende Informationen

Informationen im Internet

-

Literatur

Typen von Prototypen: Lim, Y.-K., Pangam, A., Periyasami, S., & Aneja, S. (2006).

Comparative analysis of high- and low-fidelity prototypes for more valid usability evaluations of mobile devices. In: A. Mørch, K. Morgan, T. Bratteteig, G. Ghosh, & D. Svanaes (Eds.): *Proceedings of the 4th Nordic Conference on Human-Computer Interaction: Changing roles* (NordCHI '06), New York: ACM. Pp. 291–300.

Prototyping: Sommerville, I. (2011). *Software Engineering* (9th ed.). Addison-Wesley.

Best Practice „Anforderungen mit Hilfe von Prototypen erheben“

Version 1.0 – 19.05.2015 – Autor: Fraunhofer IESE

Das Projekt PQ4Agile wird vom Bundesministerium für Bildung und Forschung im Rahmen der Maßnahme KMU-innovativ: IKT (01 | S13032) gefördert.