Einleitung

Wenn ein neues Produkt entwickelt werden soll, so sollte sich der Auftraggeber zunächst überlegen, "`warum"' dieses für Kunden interessant wäre.

Darauffolgend ist zu entscheiden, "`wie"' es entwickelt werden soll. Zuletzt muss exakt beschrieben werden, "`was"' für ein Produkt entstehen soll und welche Funktionalitäten es aufzuweisen hat.

Dies ist eine kurz gefasste Beschreibung, aber ein Unternehmer sollte im Optimalfall mittels dieses Paradigmas ein Produkt entwickeln. Letzteres Schema sei angelehnt an den \textit{Golden Circle} des Requirements Engineering\footnote{Vgl. Schenkel, Michael (2015) 3. Abschnitt im Hauptabschnitt (siehe Literaturverzeichnis \cite{GOLDENCIRCLE}).} von Innen nach außen.

Fälschlicherweise wird häufig mit dem "`was"' begonnen. Dadurch es kann vorkommen, dass teilweise unrealistische Anforderungen an das Produkt übereilt und unprofessionell vertraglich festgehalten werden. Dies sollte nicht geschehen, da gute Anforderungen ein Teil des Garant für einen erfolgreichen Projektabschluss sind. Wenn sie nicht gewissenhaft und strukturiert er- sowie bearbeitet werden, können daraus viele schwerwiegende Probleme und Folgefehler entstehen, welche bis zur Produktverfehlung und / oder Fehlkonstruktion führen können.

Im schlimmsten Fall entsteht solch eine Produktverfehlung. Das Produkt ist ein anderes. Als von den Stakeholdern gefordert. Möglicherweise wird das Projekt abgebrochen. Auch kann es vorkommen, dass das Produkt zum Zeitpunkt der Fertigstellung bereits veraltet ist. Dabei kann schon zu Beginn ein  
Fehler bei der Anforderungsaufnahme genügen, um das gesamte Projekt nachhaltig zu gefährden, oder scheitern zu lassen. Dieser Fehler kann seinen Ursprung allerdings an verschiedenen Stellen haben. Mögliche Verfehlungen in der Aufnahme sind z.B. unvollständige, falsch interpretierte oder unrealistische Anforderungen, sodass die nächsthöhere Prozessinstanz diese nicht exakt deuten, oder durchführen kann.

Eine weitere Möglichkeit um den Entwicklungsprozess zu verbessern und Fehlern entgegen zu wirken ist die agile Softwareentwicklung. Bei dieser müssen nicht direkt am Anfang alle Anforderungen in einem Pflichtenheft festgehalten werden, sondern sind dynamisch änderbar in einem Backlog festgehalten. Je nach genutztem System kann beispielsweise monatlich ein Review der bisherigen Arbeit geschehen und anschließend eine neue Liste von Anforderungen für die folgende Entwicklungsphase erstellt werden (Beispiel Scrum [5][9]). Ein solches System erhöht außerdem die Chance, dass mögliche Fehler frühzeitig erkannt und behoben werden. Diese Methode ist absolut effizient, da die Kosten um einen Fehler zu beheben von der Analyse und Design Phase (ca. 400 €), über die Test- und Abnahmephase (ca. 4800 €), bis hin zur Veränderung einer bereits in Betrieb befindlichen Software (ca. 9000 €) exponentiell zunehmen. Letzteres kann aus dem Blickwinkel der Stakeholder und Konsumenten eine Abnahme der Produkt- und der Firmenakzeptanz bewirken („Imageschäden“) [4]. Es ist also zu sehen, dass sich die Wahl eines Entwicklungsprozesses in Zusammenhang mit dem Umgang mit den Anforderungen entscheidend auf das Projekt auswirken kann. Um den Umgang mit Anforderungen effizienter zu gestalten, sowie das Risiko eines Projektabbruchs zu minimieren, soll eine Software entwickelt werden, welche sowohl bei der Aufnahme, als auch der Verwaltung von Anforderungen Unterstützung bietet.

Motivation

Der Leitgedanke, die Formulierung und Verwaltung von Anforderungen zu vereinfachen und effizienter zu gestalten, bietet die Motivation zur Entwicklung einer Software. Diese Software muss zwangsweise einen gleichwertigen, oder höheren qualitativen Output innerhalb einer Planungsphase generieren (in Relation zu Tabellen-, oder Textsoftware), um als effizient bezeichnet zu werden. Wenn Verantwortliche unachtsam oder ungeübt mit Anforderungen agieren, führt dieser suboptimale Umgang zu Qualitätsverlusten. Es erfordert explizit aufgeklärte, oder geschulte Mitarbeiter, um Anforderungen manuell ohne Softwareunterstützung abzulegen. Hierbei wird weder die Formulierung kontrolliert, noch eine konkrete Struktur gewährleistet. Infolgedessen entstehen Redundanzen und Uneindeutigkeit. Dies bedeutet einen zusätzlichen Verwaltungsaufwand, welcher die Kosten und Geschwindigkeit der eigentlichen Produktentwicklung negativ beeinflusst. Um jenen Problemen entgegenzuwirken, wird eine Onlineapplikation konzipiert, welche weniger risikoanfälliges Anforderungsmanagement ermöglicht. Während des Entwicklungsprozesses existiert die Möglichkeit, individuelle Stakeholderwünsche als Zusatzfunktionen zu implementieren, sollten genügend zeitliche Ressourcen verbleiben.

Infolge des Aufbaus als Webanwendung können Anwender jederzeit über einen gängigen Webbrowser auf ihre Anforderungen zugreifen. Aufgrund der steigenden Anzahl von Anwendern, welche über mobile Endgeräte auf die Applikation zugreifen, soll es zudem ermöglicht werden, unterwegs komfortabel über das Endgerät zu arbeiten. Dabei kann die Nutzererfahrung durch ein responsives (dynamisches) Design weiter verbessert werden. Parallel kann die Software neben dem Einsatz in Projekten als Lernsoftware für die korrekte Anforderungsformulierung fungieren. Als Grundbaustein existiert hierbei eine integrierte Formulierungsschablone nach Chris Rupp. Jene soll dem Anwender die Formulierung durch einen vorgegebenen Satzbau erleichtern. Außerdem soll es möglich werden, mit mehreren Anwendern zeitgleich an derselben Anforderungsliste zu arbeiten. Durch die resultierende Teamarbeit wird die Effizienz gesteigert und das Arbeitsklima verbessert.

Nach Abschluss des Entwicklungsprozesses besteht die Möglichkeit, dieses Projekt unter einer Open Source – Lizenz zu veröffentlichen, um individuelle Weiterarbeit Dritter zu ermöglichen.

KONZEPT – NACH DEN BILDERN

Im Bereich der Vernetzung, speziell im Web-Bereich, gibt es stetige Weiterentwicklungen. Folgend an das Moor’sche Gesetz angelehnt existieren immer kürzere Technologie-Lebenszyklen. Dieser Fakt sowie ein verändertes Nutzungsverhalten verringern die Attraktivität statischer Websites. Aktuell spielt die Verbreitung von mobilen Endgeräten und steigende mobile Datennutzung eine größer werdende Rolle. Für Suchanfragen über die Internet-Suchmaschine „Google“, die von mobilen Endgeräten durchgeführt werden, wird bereits seit Juni 2013 eine Anpassung des Ranking-Algorithmus bekannt gegeben [1]. Am 26.02.2015 kündigte der Konzern an, das Ranking für „mobile-friendly“ Websites mit in die Gesamtbewertung dieser Suchanfragen einfließen zu lassen[2]. Diese Anpassung ist planmäßig seit dem 21.04.2015 implementiert und der Öffentlichkeit zugänglich[3]. „Google“ unterstützt mittels dieses Schrittes den technischen Wandel, welcher die Denkweise der Webentwickler nachhaltig beeinflusst. Letzteres sowie das veränderte Konsumverhalten der Technik-Anwender verdeutlichen, dass derzeit eine hohe technische und konsumangepasste Versiertheit der Entwickler nötig ist. Infolgedessen werden bei der Entwicklung der webbasierten Software in diesem Projekt verschiedene Technologien miteinander verbunden. Das spiegelt die tiefergehende Evaluation der Anwenderbedürfnisse wider und fördert sowie verstärkt das Nutzungserlebnis. Eine Auflistung der genutzten Tools, Sprachen sowie Frameworks finden sich unter Punkt 4.2 („Realisierung“). Nach der Überprüfung unterschiedlicher Konzeptionen wird nachfolgendes Konzept festgehalten:

Die Anwendereingaben werden vom Browser erfasst. Zunächst schickt dieser einen standard HTTP-Request an den Server. In der Antwort sind anschließend HTML-, CSS- und Scriptdateien enthalten. Diese benutzt der Browser, um den Inhalt aufzubereiten und die Darstellung anzupassen. Somit werden wichtige Funktionen und auch die Darstellung auf mobilen Endgeräten ermöglicht. Weiterhin kann der Funktionsumfang der Website mittels jQuery und Bootstrap nach Belieben erweitert werden. Dem Anwender werden mehrere Funktionalitäten bereitgestellt. Einen Überblick über diese gibt die Abbildung 4.4. Neben dem Registrieren und Einloggen, wird sich der Anwender nach Beendigung seiner Arbeits-Session vom System abmelden können.

Nach der Anmeldung befindet sich der Anwender auf einer Übersichtsseite, genannt „Dashboard“, von welcher aus ihm weitere Interaktionsmöglichkeiten mit dem System bereitgestellt werden. Als Interaktionen existieren das Verfassen sowie Bearbeiten von Anforderungen. Zudem lässt sich die Ansicht individuell an die Anwenderwünsche mittels Zeilensortierungen anpassen. Entsprechende Aktionen sind in Abbildung 4.4 grün markiert.

Alle Anforderungen werden dem Team des Verfassers zugewiesen. Sie sind somit an dieses gebunden und durch alle Mitglieder einsehbar.

Alle Mitglieder eines Teams besitzen das Recht, dieses zu bearbeiten. Entsprechende Interaktionen diesbezüglich sind in Abbildung 4.4 blau markiert. Ein Anwender kann zudem seine Nutzerdaten ändern. Dies bezieht sich auf das Passwort und die eingetragene E-Mail Adresse. Diese Interaktionen sind in der Abbildung 4.4 gelb markiert. Um das System zu Warten, besitzt ein Administrator die Möglichkeiten, Nutzer zu entfernen sowie Teams anderen Nutzern zu überschreiben.

Folgende Meilensteine sind für den Entwicklungszeitraum geplant: das Aufsetzen eines Datenbanksystems sowie aller nötigen Abhängigkeiten. Folgend muss ein funktionsfähiges Authentifizierungssystem für die Anwender implementiert werden. Dies ist notwendig, da die Software direkt auf Basis dieses Multi-User-Systems weiter entwickelt werden soll. Um dieses Grundgerüst zu schaffen und möglichst viele Fehlerquellen schon während der Entwicklungsphase ausfindig zu machen und zu beheben, wird zunächst mit einem Zeitrahmen von etwa sechs Wochen eingeplant. Diese Zeitspanne dient als Orientierungspunkt und wurde durch etwaige Aufwandschätzungen der einzelnen Aufgaben festgelegt. Ebenso ist dieser Schritt als Meilenstein anzusehen. Nachdem der Test des Benutzersystems auf seine Funktionalitäten hin überprüft wurde, erfolgt die Implementierung der Anforderungsschablone sowie dessen notwendige Funktionen. Es muss gewährleistet sein, dass Anforderungen sowohl in die Datenbank geschrieben als auch ausgelesen werden. Dazu ist ein einzigartiger Separator notwendig, damit die Satzglieder der Anforderungen später rekonstruiert werden können. Auch muss es ermöglicht werden, dass Metainformationen wie z.B. die Identifikationsnummer des Erstellers, oder ein Zeitstempel für die Anforderungssortierung gespeichert werden können. Ist dieses Ziel erreicht muss ein geeignetes visuelles Ausgabeformat gewählt und die korrekte Darstellung der Anforderungen überprüft werden. Zusätzlich muss die Übersichtlichkeit sowie Individualisierbarkeit der Anforderungen gewährleistet sein. Um dies zu realisieren, sind Sortierfunktionen erforderlich. Nach Verifizierung der ordnungsgemäßen Funktion soll der zweite Meilenstein nach weiteren sieben bis acht Wochen erreicht sein. Der dritte Meilenstein besteht aus dem Teamsystem. Jenes soll gleichzeitiges Arbeiten an derselben Anforderungsliste ermöglichen. Ein Team muss erstellt werden können, damit alle weiteren Implementierungen darauf aufbauen können. Das Verwalten von Teams, insbesondere das Hinzufügen und Löschen anderer Nutzer ist wird im Funktionalitätenumfang der Software enthalten sein, da ein funktionierendes Teamsystem anders nicht möglich ist. Besonders bei diesem Teil des Systems muss kontrolliert werden, dass der Anwender keine Fehler erzeugen kann. Vor allem betrifft dies das Hinzufügen von Nutzern und die Namenswahl der Teams. Um Komplikationen bei den Datenbankanfragen zu vermeiden, sollte sich jeder Nutzer nur in einem Team befinden. Die Teamnamen müssen zur korrekten Identifikation und zur Vermeidung von Redundanzen einzigartig sein. Weiterhin beinhaltet die Erfüllung des dritten Meilensteins auch die Anpassung der Darstellung, bzw. Optik der Software. Dazu gehören beispielsweise Tooltipps, um eine verständlichere Kommunikation und Interaktion zwischen Mensch und Maschine herbeizuführen und zu ermöglichen. Zu diesem Zweck sollen auch Dialoge und Komplexität der Menüführung kontrolliert werden. Zur Überprüfung der Intuitivität kann die Software von Dritten getestet werden. Aufgrund des steigenden Testumfanges, wird der Zeitaufwand hier höher geschätzt, als bei den vorherigen Meilensteinen. Sobald alle Tests diesbezüglich erfolgreich verlaufen sind, wird dieser Meilenstein abgeschlossen sein. Allerdings werden hier weiterhin Anpassungen und eventuell Fehlerbehebungen notwendig sein, da in diesem zeitlichen Rahmen nicht alle Fälle der Benutzung simuliert werden können. Verbleibende Zeit wird genutzt, um Fehler zu beheben und die Software zu verbessern. Außerdem können weitere Ideen und Features evaluiert und implementiert werden, da die Entwicklung nicht auf einem Wasserfallmodell beruht, sondern an agile Softwareentwicklung angelehnt ist. Dies gibt die Möglichkeit, zusätzliche Anwenderwünsche während jeder Entwicklungsphase berücksichtigen zu können.