

# Kritische Reflexion

Dies ist eine Reflexion über das Entwicklungsprojekt interaktiver System vom Sommersemester 2019 an der TH Köln. Im folgenden wird beschrieben warum das Projekt, in seiner aktuellen Form nicht marktfähig wäre.

Es soll ein System entwickelt werden, dass es Kleingärtnern erlaubt ihre Übererzeugnisse schnell und einfach direkt zu vermarkten, um Personen eine weitere Möglichkeit zu bieten regional und saisonal einzukaufen.

Zu Beginn der Projektdefinition stand das Problem, dass die hohen CO2 Werte durch Importe zu hoch sind zwar im Fokus, aber mit fortschreitender Bearbeitung des Projekts rückte dieser Gedankengang mehr in den Hintergrund. Das Projekt entwickelte sich langsam in die Richtung eines E-Commerce Systems für Kleingärtner. Dieses Problem war während der Erarbeitung von Artefakten über die Domäne leicht erkennbar. Der Handel und die Produktion von Obst und Gemüse wurden mehr beleuchtet, als schon existierende alternativen Verteilungswege der Erzeugnisse von Kleingärtnern.

Die Marktrecherche erwies sich als schwierig. Ein direktes Konkurrenzprodukt existiert nicht in dieser Form. Daher konnten nur schwierig Konkurrenten herangezogen werden. Rückblickend betrachtet wurde festgestellt, dass solche Systeme, zumindest in Deutschland, nicht existieren, weil die Gesetzeslage eindeutig definiert, dass Verkäufe bis zu drei mal pro Jahr vom Grundstück aus erlaubt sind, aber keine Kontrollen oder Nachweise darüber vorzeigbar wären. Somit könnte sich das System in eine rechtliche Grauzone begeben. Zudem wurden in der Marktrecherche Vereine von Kleingärtnern, die Erzeugnisse untereinander verteilen, nicht in Betracht gezogen. In Anbetracht der Vereine als Konkurrenz, hätte das System sich wohlmöglich nicht in Richtung E-Commerce entwickelt.

Die Alleinstellungsmerkmale waren einseitig beschrieben und platzierten sich eher als besseres Verkaufsargument für regionales Obst und Gemüse vom Kleingärtner. Ergänzend wurde gegen Ende des Projekts das Gesamtkonzept, um die Möglichkeit der Rezeptvorschläge durch einen externen Webservice, erweitert.

Die Zielhierarchie wurde gut strukturiert gelistet, jedoch war die Zielsetzung für das Projekt zu hoch angesetzt, da die Umsetzung innerhalb eines Semesters nicht möglich war.

Die Risiken waren zahlreich und projektspezifisch. Der Vorteil war das in der Konzeptionsphase schon PoCs adressiert werden konnten. Leider waren diese nicht durchgängig technisch und daher auch teilweise fehlerhaft in dieser Form.

Zu Beginn der Definition von Anforderungen, waren die funktionalen gut definiert, jedoch die qualitativen und organisationalen nicht. Daher musste dieses Artefakt überarbeitet und die einzelnen Anforderungen messbar werden. Mit stetigen Fortschritt im Projekt wurde die Liste der Anforderungen ergänzt und mit der Erstellung des UI Prototypen gegen geprüft.

Die Kommunikationsmodelle waren syntaktisch fehlerhaft. Der nummerierte Ablauf der deskriptiven und präskriptiven Kommunikationsmodelle hat komplett gefehlt. Zudem wurde in den Modellen erstmals die Datenübertragung mit Zahlungsdiensten aufgegriffen, jedoch im restlichen Konzept nicht spezifiziert. Hier hätten Alternativen noch erwähnt werden können.

Die PoCs waren nicht technisch umgesetzt. Sie beschrieben keinerlei Kommunikation zwischen Client - Dienstnutzer und Dienstgeber. Dies lässt sich, zu diesem Zeitpunkt, auf mangelndes Fachwissen zurückführen. Dieses Artefakt wurde überarbeitet und die einzelnen PoCs messbar gemacht. Leider waren es auch zu wenige. Das hatte direkten Einfluss auf die spätere Codierung.

Zum ersten Meilenstein waren in der Softwarearchitektur Mängel vorhanden. Es wurde die clientseitige und serverseitige Anwendungslogik überarbeitet. Zudem wurde das Modell um eine Legende erweitert. Und kleinere Logikfehler, wie die graphische Darstellung der Einbindung von externen Webservices, wurden behoben.

Der Methodische Rahmen war zum ersten Meilenstein nicht gut begründet und abgewägt. Zum zweiten Meilenstein wurde das Vorgehensmodell stark überarbeitet und für die Domäne passend ausgewählt. Begründet wird das durch das steigende Verständnis über das System das umgesetzt werden soll. Durch den hinzugewinn eines neuen eigenständigen Alleinstellungsmerkmals wurde die Komplexität der Funktionsweise des Systems erhöht. Das "usage centered design" war optimal für die Umsetzung des funktional ausgeprägten Systems.

Inbegriffen in dem "usage centered design" ist das Vorgehensmodell von Larry Constantine. Der erste Schritt war eine Stakeholderanalyse zu erstellen. Die Stakeholder waren mit den Objekten aus der Domänenrecherche vergleichbar. Leider war auch hier der Fokus zu sehr auf E-Commerce gerichtet. Zwar wurde das Problem um das CO2 behandelt, nur wurden beispielsweise auch Vereine vernachlässigt. Die Einbeziehung dieser Vereine hätte das System zu diesem Zeitpunkt noch beeinflussen können, sodass der Handel und Verkauf von Obst und Gemüse nicht hauptsächlich im Fokus steht.

Die "User Profiles" und die daraus resultierende "User Role Map" waren nicht eindeutig genug. Sie waren sehr grob und klein gehalten. Die "User Role Map" hätte nicht erstellt werden müssen. Vielleicht hätte wie bei der Stakeholderanalyse die zusätzliche Betrachtung von Vereinen eine entscheidende Rolle spielen können.

Nach der "User Role Map" wurden projektspezifische "Use Cases" und ihre essentiellen Schritte erstellt. Da die Erstellung von jeglichen "Use Cases" die nicht projektspezifisch sind sehr zeitaufwändig ist wurde auf diese verzichtet. Zu bedauern ist das nur zwei projektspezifische "Use Cases" zu identifizieren waren. Zudem wirkt deren Erstellung rückblickend gesehen überhastet und nicht detailliert genug. Durch die Erweiterung des Gesamtprojekts um ein weiteres Alleinstellungsmerkmal wurde die zugehörige Erstellung eines weiteren "Use Cases" schlicht vergessen. Das Alleinstellungsmerkmal zur Berechnung des CO2 Wertes fand leider gar keine Berücksichtigung, da die Annahme, dass der Ablauf nur serverseitig stattfindet und keine "Intention des Nutzers" voraussetzt falsch ist. Die "Use Case Map" beinhaltet nur wenige Teilfunktionen des Gesamtsystems und hätte weiter ausgearbeitet werden müssen, um spätere Codierungsansätze ableiten zu können.

Für die "User Profiles" und die "Use Cases" hätte wesentlich mehr Zeit aufgewendet werden müssen.

Darauf folgend wurde eine "Navigation Map" erstellt. Durch diese Karte konnte man anschließend den GUI Prototypen erstellen. Die "Navigation Map" war schlicht, aber stellte alle notwendigen graphischen Abläufe des Systems dar. Es hätte noch eine Legende erstellt werden müssen, um die einzelnen Elemente besser nachvollziehen zu können.

Der GUI Prototyp war zu Beginn nur papierbasierend und recht einfach gehalten. Das wirkte sich aber nicht nachteilig auf die weitere Evaluation aus. Die Evaluation erfolgte mittels "Cognitive Walkthrough", da dieser Schritt "usage" basierend ist. Während der Durchführung viel auf das einige Buttons fehlen. Zum Beispiel auf dem Produktescreen fehlte ein Button um einzelne Produkte hinzuzufügen, zudem war dem Nutzer unklar was nach der zu erfüllenden Aufgabe passiert. Anschließend wurde die Evaluation zu einem "screen based" Prototypen erarbeitet.

Die Datenstrukturen waren in Ordnung.

In den REST Modellierung gab es Diskrepanzen bei der Unterscheidung zwischen Benutzer und Gärtner. Die Gärtner waren ursprünglich als Subressouce eines Benutzers gedacht, aber diese Modellierung wurde wieder verworfen.

Die Pub & Sub Kommunikation war in Ordnung.

Die Codierung war der schwierigste Teil des Projekts. Leider wurde viel zu spät damit begonnen und mangels Wissen in JavaScript und Java konnte kein lauffähiger Prototyp umgesetzt werden. Es konnte nur eine anfängliche Kommunikation von Dienstgeber mit der Datenhaltung realisiert werden.

Abschließend muss man sagen, dass der Hauptfokus des Projekts auf den MCI Teil gelegt wurde. Aufgrund mangelnder Programmierkenntnisse in Java und JavaScript, sowie die Anwendung dieser Sprachen auf Basis von Android wurde der WBA2 Teil hauptsächlich nur konzeptionell bearbeitet. Mehr Bearbeitungszeit hätte vermutlich zu einem funktionierenden Prototypen führen können. Zudem war die Umsetzung der vorausschauenden Vorhersage der CO2 Werte mittels Java oder JavaScript nicht möglich, da keine aktuelle library existiert die diese Algorithmen bereitstellen.