

Technology Arts Sciences TH Köln

TH Köln Campus Gummersbach
Fakultät für Informatik und
Ingenieurwissenschaften

ENTWICKLUNGSPROJEKT INTERAKTIVE SYSTEME

Projektdokumentation “PreFer“

Thuy Trang Nguyen
Duc Giang Le

betreut von:
Prof. Dr. Kristian Fischer
Prof. Dr. Gerhard Hartmann
Robert Gabriel
Sheree Saßmannshausen

3. Dezember 2016

Inhaltsverzeichnis

1	Einleitung	3
1.1	Motivation	3
1.2	Herleitung des Projektnamen	3
2	Benutzermodellierung	4
2.1	Stakeholderanalyse	4
2.2	User Profiles	6
2.2.1	User Profile 1: erfahrener Landwirt	7
2.2.2	User Profile 2: landwirtschaftlicher Mitarbeiter	7
2.2.3	User Profile 3: angehender Landwirt	7
2.3	Persona	7
2.3.1	Personae 1: Nikolas Stein	7
2.3.2	Personae 2: Emily Solay	11
2.3.3	Personae 3: Thomas Blaumann	12
2.4	Fazit Benutzermodellierung	12
3	Benutzungsmodellierung	12
3.1	Szenarien	13
3.1.1	Szenario 1: Düngungsprozess bei einem erfahrenen Landwirt	13
3.1.2	Szenario 2: Düngungsprozess bei einem Landwirt mit wenig Erfahrung	13
3.2	Hierarchical Task Analysis	14
3.3	Essential Use Cases	14
3.4	Concrete Use Cases	14
3.5	Fazit Benutzungsmodellierung	14
4	Anforderungen	14
4.1	funktionale Anforderungen	14
4.2	qualitative Anforderungen	14
4.3	organisationale Anforderungen	15
5	Gestaltungslösung / UI Prototyp	15
5.1	15
6	UI-Evaluation	15
7	Systemarchitektur	16
7.1	Client	17
7.2	Server	17
7.3	Datenformat und Übertragungsprotokoll	18
7.4	Externe Schnittstelle	18
7.5	Bodensensoren	18
7.6	Firebase Cloud Messaging[1]	18

8	Proof of Concept	19
8.1	Einbindung einer externen Schnittstelle	19
8.2	Ermittlung des Standortes durch GPS in Google Maps	20
8.3	Erstellen von Polygonen auf den Satellitenbildern	20
8.4	Kommunikation zwischen Systemkomponenten	21
8.5	Nutzung des Firebase Cloud Messaging Service	22
9	Datenstrukturen	23
9.1	Pflanzen	23
9.2	Feld	24
9.3	Bodendaten	24
9.4	Düngeempfehlung	25
9.5	Firebase Cloud Messages	25
10	Architekturmerkmale	25
10.1	REST-Spezifikation	25
10.2	Anwendungslogik	26
10.2.1	Berechnung einer Düngeempfehlung	26
10.2.2	Erstellen der Düngerkarte bzw. Erstellen von Teilflächen .	26
10.2.3	Ausbringen des Düngers basierend auf der Düngeempfehlung	26

1 Einleitung

Die folgende Ausarbeitung ist eine Projektdokumentation zum Projekt “PreFer“, das im Rahmen des Moduls EIS (Entwicklungsprojekt Interaktive Systeme) entstanden ist. Sie beschreibt Prozesse, die im Projektverlauf entstanden sind, und notwendige Aspekte der Modellierung von Mensch-Computer-Interaktion und Webbasierte Anwendungen 2-Artefakten. Ziel dieser Dokumentation ist es, die ausgewählten Methoden und Modellierungen zur Gestaltung und Entwicklung des Systems festzuhalten und wichtige Abwägungen und Entscheidungen, die im Projektverlauf getroffen wurden, zu dokumentieren und zu begründen.

1.1 Motivation

Die Landwirtschaft verliert in der heutigen Zeit immer mehr an Bedeutung, obwohl sie für die Gesellschaft einen essentiellen Bereich darstellt, denn sie ist für die Versorgungssicherheit der Nahrungsmittel wie auch zunehmend auch bei Energie und Rohstoffen verantwortlich. Besonders deutlich wurde dies bei der Debatte mit den billigen Milchpreise in den Discountern. [] Durch die fallenden Milchpreise leiden viele Landwirte an wirtschaftlichen Verlusten, so dass es sogar Landwirte gibt, die existenzgefährdet sind. Gerade Landwirte in kleineren Betrieben haben es schwer, weil sie im Vergleich zu Großbetrieben die schlechteren und weniger Ressourcen zur Verfügung haben. Dadurch ist es vielen von ihnen nicht immer möglich, eine effiziente und produktive Landwirtschaft zu führen.

Die Motivation dieses Projekts besteht nun darin, diesen Kleinlandwirten dabei zu unterstützen, eine in bestimmten Bereichen möglichst effiziente und produktive Landwirtschaft führen zu können.

1.2 Herleitung des Projektnamen

Schon früh im Projektverlauf und im Konzept wurde deutlich, dass ein Ziel des Projekts ist, den Kleinlandwirten bei der Düngeplanung, bei der Düngemengenberechnung und beim Düngungsvorgang zu unterstützen. Es soll hierbei auf eine besondere Bewirtschaftungstechnik zurückgegriffen werden, die sich als Precision Farming bezeichnet. Aus der Kombination mit dem Begriff der Precision und dem englischen Begriff von Dünger (Fertilizer) wurde abschließend der Projektname “PreFer“ abgeleitet.

2 Benutzermodellierung

Der Nutzungskontext umfasst unter anderem die Benutzer eines interaktiven Systems. Diese sind ein wesentlicher Aspekt zur Identifizierung des Nutzungskontextes. Für die Benutzermodellierung soll als Erstes eine Stakeholder-Analyse durchgeführt werden, die Grundlage zur Erstellung von User Profiles dienen. Aus diesen User Profiles soll im Anschluss Persona abgeleitet werden, die die potentiellen Benutzer repräsentieren.

2.1 Stakeholderanalyse

Bereits in der Domänenrecherche im Konzept wurden einige wichtige Stakeholder identifiziert. Nach der ISO-9241, Teil 210 sind Stakeholder eine “Einzelperson oder Organisation, die ein Anrecht, einen Anteil, einen Anspruch oder ein Interesse auf ein bzw. an einem System oder an dessen Merkmalen hat, die ihren Erfordernissen und Erwartungen entsprechen.“ [[2]] Diese Eigenschaften und Merkmale der Stakeholder soll im Folgenden analysiert werden. Basierend auf der Stakeholder-Analyse können wichtige Stakeholder identifiziert werden, die Grundlage zur weiteren Modellierung der User Profiles und Persona dienen.

Tabelle 1: Stakeholder-Analyse

Bezeichnung	Beziehung zum System	Beschreibung	Erwartungen & Erfordernisse	Priorität im Projekt
Landwirte	Anspruch	Durch die Verwendung des Systems können sie die Düngerplanung und Düngungsvorgang nach dem Prinzip der Teilflächenbewirtschaftung gestalten.	Vereinfachung der Arbeitsprozesse beim Düngen, Düngung nach der Teilflächenbewirtschaftung, Schonen von Ressourcen	hoch
Familie von Landwirten	Interesse	Sie interessieren sich für den wirtschaftlichen Gewinn des Landwirten. Sie können aber auch als Familie im Betrieb mithelfen.	Schonen von Düngerressourcen, Unterstützung bei der Düngung	mittel
Landwirtschaftliche Mitarbeiter	Anspruch	Sie sind beim Landwirten angestellt und erledigen landwirtschaftliche Arbeiten auf seinem Betrieb.	Vereinfachung der Arbeitsprozesse beim Düngen	hoch
Lohnunternehmer	Interesse, Anspruch	Sie erledigen in Rechnung gestellte landwirtschaftliche Arbeiten. Teilflächenspezifische Düngung ist durch das System möglich.	Vereinfachung der Arbeitsprozesse beim Düngen	mittel
Staat / Land / Landwirtschaftskammer	Interesse	Sie sind daran interessiert, dass die Landwirte ihren Dünger bedarfsgerecht ausbringen, damit keine Überdüngung entsteht. Eine Überdüngung schädigt die Umwelt.	Unterstützung der Landwirte bei der bedarfsgerechten Düngung, Landwirte sollen sich an Düngeverordnung halten.	gering
Abnehmer der Ernte	Interesse	Sie arbeiten zusammen mit den Landwirten, indem sie die Produkte der Landwirte verkaufen. Sie profitieren durch die Verbesserung der Qualität ihrer gekauften Produkte.	Verbesserung der Qualität der Ernte der Landwirte durch Nutzung des Systems	gering
Konsumenten	Interesse	Sie konsumieren die aus der Landwirtschaft erzeugten Produkte der Landwirte.	Verbesserung der Qualität der landwirtschaftlichen Produkte durch Nutzung des Systems, Konsumenten erwarten stets eine gute Qualität ihrer gekauften Produkte.	gering

2.2 User Profiles

Nach Abschluss der Stakeholder-Analyse können für die wichtigsten Stakeholder, die direkt mit dem System interagieren, User Profiles erstellt werden. In diesem Fall sind Landwirte und die landwirtschaftlichen Mitarbeiter primäre Benutzer. Diese sollen Im Folgenden detaillierter charakterisiert und deren Eigenschaften und Merkmale definiert werden.

2.2.1 User Profile 1: erfahrener Landwirt

2.2.2 User Profile 2: landwirtschaftlicher Mitarbeiter

2.2.3 User Profile 3: angehender Landwirt

2.3 Persona

In diesem Abschnitt werden auf Basis der zuvor definierten User Profiles einige Persona abgeleitet und erstellt. Sie dienen zur Repräsentation von potentiellen Benutzern und erweitern den Grad der Detailliertheit der Benutzermodellierung.

2.3.1 Personae 1: Nikolas Stein

Diese Personae ist aus dem User Profile 1 abgeleitet und entstanden und ist an keinem Real User angelehnt.

Name: Nikolas Stein

Alter: 54

Geschlecht: männlich

Familienstatus: verheiratet und 5 Kinder

Beruf: Landwirt

Einkommen:

Ziele: Aufrechterhaltung des Familienbetriebs

Nikolas Stein ist 54 Jahre alt, wohnt in der Nähe von Vechta auf seinem landwirtschaftlichen Familienbetrieb und ist mit seiner Frau Nici Stein seit 30 Jahren verheiratet, mit der er gemeinsam 5 Kinder hat. Der Familienbetrieb ist seit zwei Generationen im Besitz der Familie Stein. In seinen jungen Jahren hat er schon angefangen, seinen Eltern auf dem Hof zu helfen, wodurch er seine Begeisterung zur Landwirtschaft entdeckt hatte.

Nach seinem Realabschluss hat er eine duale Ausbildung zum Landwirten begonnen, die er nach drei Jahre erfolgreich abgeschlossen hat. Nach der Ausbildung übernahm er den Betrieb seiner Eltern. Mit der Zeit wurden die Aufgaben und Arbeiten im Betrieb Routine für ihn. Er mochte es, seine Tiere zu füttern, sich um seine Pflanzen zu kümmern, sogar die Reparatur seiner Maschinen hatte ihm Spaß gemacht.

Mittlerweile besitzt er Ackerfelder mit einer gesamten Größe von ca. 60 ha mit verschiedenen angebauten Pflanzen. Da er sich alleine nicht um alles kümmern kann, müssen seine Kinder auf dem Hof mithelfen. Ein Teil seiner Ernte verkauft er selbständig auf seinem Hof und einen anderen Teil gehen an Lebensmittelhersteller, die schon seit Jahren Kunden von ihm sind.

Für seine Arbeit auf dem Hof werden viele wichtige Informationen auf seinem Hofcomputer gespeichert. Außerdem hat er sich vor 3 Jahren eine Software

Tabelle 2: User Profile 1: erfahrener Landwirt

Merkmale	Merkmalsausprägung
Stakeholder	Landwirt
Alter	40-70
Geschlecht	männlich / weiblich
Wohnort	deutschlandweit in ländlichen Gegenden
Berufserfahrung	<ul style="list-style-type: none"> - Landwirt - mindestens 15 Jahre Berufserfahrung - fortgeschrittene Erfahrung in landwirtschaftlichen Arbeitsvorgängen und -prozessen
Informationen zum Unternehmen	<ul style="list-style-type: none"> - arbeitet auf einem landwirtschaftlichen Betrieb - Betrieb mit bis zu 100 ha landwirtschaftlich genutzter Fläche
Computerkenntnisse und -erfahrung	<ul style="list-style-type: none"> - durchschnittlich bis gut - tägliche Nutzung für seine landwirtschaftliche Arbeit
Fachwissen	<ul style="list-style-type: none"> - Wissen über alle seine Felder und angebauten Pflanzen - Wissen über Düngemittel und Düngungsvorgang - Wissen über landwirtschaftliche Arbeitsvorgänge (Ernte, Düngung, Bodenbearbeitung, Tierfütterung, ...)
Arbeitsmittel	<ul style="list-style-type: none"> - Landwirtschmaschinen (Düngemaschine, Mähmaschine, Traktoren, ...) - Computer - Smartphone - Werkzeug (Hammer, Scharuber, ...) - Boden / Erde
verfügbare Technologie	<ul style="list-style-type: none"> - Computer/Desktop-PC - Notebook - Smartphone - Tablet - Landwirtschaft-Software - Landwirtschaftsmaschinen
Spezielle Produkterfahrung	<ul style="list-style-type: none"> - Erfahrung in Nutzen von Landwirtschaft-Software
Motivation	<ul style="list-style-type: none"> - Verbesserung und Vereinfachung des Düngerprozesses - Bedarfsgerechte Düngung, Schonen von Düngerressourcen
Einstellungen und Werte	<ul style="list-style-type: none"> - nutzen gerne leicht bedienbare und verständliche Software - Angst neue Technologien zu probieren: lieber alt und funktionsfähig als neu und schwer bedienbar

Tabelle 3: User Profile 2: landwirtschaftlicher Mitarbeiter

Merkmale	Merkmalsausprägung
Stakeholder	Landwirtschaftliche Mitarbeiter
Alter	18-50
Geschlecht	männlich / weiblich
Wohnort	deutschlandweit
Berufserfahrung	<ul style="list-style-type: none"> - landwirtschaftlicher Mitarbeiter - mindestens 3 Jahre Berufserfahrung - Erfahrung in landwirtschaftlichen Arbeitsprozessen
Informationen zum Unternehmen	<ul style="list-style-type: none"> - sind in landwirtschaftlichen Betrieben eingestellt - Landwirte sind Arbeitgeber - werden in landwirtschaftlichen Betrieben mit bis zu 100 ha landwirtschaftlicher genutzter Fläche eingesetzt
Computerkenntnisse und -erfahrung	<ul style="list-style-type: none"> - durchschnittlich bis sehr gut - tägliche Nutzung für landwirtschaftliche Arbeit - tägliche private Nutzung - Dokumentation seiner Arbeit/Arbeitsfortschritte
Fachwissen	<ul style="list-style-type: none"> - Wissen über einige landwirtschaftliche Arbeitsvorgänge - Wissen über bestimmte zugeordnete Ackerfelder
Arbeitsmittel	<ul style="list-style-type: none"> - Landwirtsmaschinen (Düngemaschine, Mähmaschine, Traktoren, ...) - Computer - Smartphone - Werkzeug (Hammer, Scharuber, ...) - Boden / Erde
verfügbare Technologie	<p>Ein Landwirt kann folgende Technologien besitzen:</p> <ul style="list-style-type: none"> - Computer/Desktop-PC - Notebook - Smartphone - Tablet - Landwirtschaft-Software - Landwirtschaftsmaschinen
Motivation	- Verbesserung und Vereinfachung des Düngerprozesses
Einstellungen und Werte	- offen für neue Technologien, die ihm bei der Arbeit helfen können

Tabelle 4: User Profile 3: angehender Landwirt

Merkmale	Merkmalsausprägung
Stakeholder	Landwirt
Alter	18-39
Geschlecht	männlich / weiblich
Wohnort	deutschlandweit
Berufserfahrung	- angehender Landwirt - kaum Erfahrung mit landwirtschaftlichen Arbeitsvorgängen
Informationen zum Unternehmen	- arbeitet auf einem landwirtschaftlichen Betrieb - Betrieb mit bis zu 100 ha landwirtschaftlich genutzter Fläche
Computerkenntnisse und -erfahrung	- durchschnittlich bis sehr gut - tägliche private und arbeitsbedingte Nutzung
Fachwissen	- Wissen über einige landwirtschaftliche Arbeitsvorgänge - Wissen über Düngung
Arbeitsmittel	- Landwirtsmaschinen (Düngemaschine, Mähmaschine, Traktoren, ...) - Computer - Smartphone - Werkzeug (Hammer, Scharuber, ...) - Boden / Erde
verfügbare Technologie	Ein Landwirt kann folgende Technologien besitzen: - Computer/Desktop-PC - Notebook - Smartphone - Tablet - Landwirtschaft-Software - Landwirtschaftsmaschinen
Motivation	- Verbesserung und Vereinfachung des Düngerprozesses - Bedarfsgerechte Düngung, Schonen von Düngerressourcen
Einstellungen und Werte	- offen für neue Technologien, die ihm bei der Arbeit unterstützen - Begeisterung in Nutzen von neuen Technologien

angeschafft, die ihm Informationen über Ressourcenverbrauch, Ernte usw. informiert und dokumentiert. Durch die Software konnten einige Arbeitsprozesse vereinfacht werden und Zeit gespart werden.

2.3.2 Personae 2: Emily Solay

Diese Personae ist aus dem User Profile 2 abgeleitet und ist an keinem Real User angelehnt.

Name: Emily Solay

Alter: 35

Geschlecht: weiblich

Familienstatus: geschieden und eine Tochter

Beruf: Angestellte in einem Ackerbaubetrieb

Einkommen:

Ziele: eine gute Mutter für ihre Tochter sein, ihre Interesse ausleben

Emily Solay ist 35 Jahre alt, wohnt in einer kleinen Stadt in der Nähe von München. Sie ist von ihrem Ehemann geschieden, mit dem sie gemeinsam eine Tochter hat. Bevor ihre Tochter auf die Welt kam, hat sie eine Ausbildung im Bereich der Agrarwirtschaft erfolgreich abgeschlossen und konnte bereits erste Erfahrungen sammeln.

Schon früh hat sie gemerkt, dass sie sich für die Agrarwirtschaft interessiert. Direkt nach ihrem Schulabschluss hat sie sich um eine Ausbildung bemüht. Nach ihrer Ausbildung hat sie begonnen, als Landwirtin im Ackerbau eines landwirtschaftlichen Großbetriebes in der Nähe von Dresden zu arbeiten. Zu ihren Tätigkeiten gehörten unter anderem die Bodenbearbeitung, das Düngen und Gießen der angebauten Pflanzen oder das Einholen der Ernte. In diesem Betrieb wurden moderne Maschinen und Technologien für die Arbeit eingesetzt. Sie hat jedes System probiert und war begeistert, wie diese ihre Arbeit vereinfacht haben. Sie musste vieles nicht mehr selber tun, die Maschinen haben viel für sie übernommen. Durch ihre Arbeit hat sie viele Erfahrungen im Hinblick auf die Agrarwirtschaft und die modernen Technologien sammeln können.

Aufgrund ihrer Schwangerschaft musste sie ihren Job vorerst aufgeben. Sie ist dann zurück in ihre Heimat gezogen, wo ihre Tochter bekommen hatte. Nachdem ihre Tochter nun älter ist und sie nicht ständig auf sie aufpassen muss, hat sich Emily für mehrere Stellen als landwirtschaftliche Mitarbeiterin im Ackerbau beworben. Nun arbeitet sie auf einem kleinen Ackerbaubetrieb in ihrer Gegend. Dort werden jedoch keine modernen Maschinen eingesetzt, weshalb sich Emily erst neu einarbeiten muss.

2.3.3 Personae 3: Thomas Blaumann

Diese Personae ist aus dem User Profile 3 abgeleitet und ist an keinem Real User angelehnt.

Name: Thomas Blaumann

Alter: 30

Geschlecht: männlich

Familienstatus: ledig

Beruf: angehender Landwirt

Einkommen:

Ziele: Erfolg im Berufsleben, Aufbau eines Familienbetriebs

Thomas Blaumann ist 30 Jahre alt, wohnt in der Nähe von Dresden mit seinen Eltern. Er lässt sich als eine sehr ehrgeizige und motivierte Person beschreiben. Alle Aufgaben und Hindernisse, die in seinem Leben schon aufgetreten sind, hat er immer mit viel Engagement erledigt.

Er hat einen Bachelor of Science durch ein Informatik-Studium erworben und arbeitete in einer IT-Firma als Entwickler. Schon früh hatte er seine Begeisterung zu neuen Technologien erkannt, die ihm sein Leben vereinfachen. Thomas besitzt auch stets die neuesten technischen Geräte wie Smartphone, Tablet oder Computer. Jedoch hat er sich die Arbeit in der IT-Firma ganz anders vorgestellt. Er kann sich nicht vorstellen, für immer in einem Büro oder am Schreibtisch zu arbeiten.

Thomas und seine Eltern haben schon immer davon geträumt, die Großstadt Dresden zu verlassen und einen Familienbetrieb auf dem Land aufzubauen. Zusammen haben sie nun entschieden, einen kleinen Hof mit landwirtschaftlichen Flächen zu erwerben und den Familienbetrieb aufzubauen. Bevor sie diesen Traum in die Tat umsetzen, besucht Thomas Abendkurse um sich landwirtschaftliches Wissen anzueignen. Dadurch lernt er verschiedene Arbeitsvorgänge kennen, die er auf dem Ackerfeld anwenden muss, um einen erfolgreichen Familienbetrieb führen zu können. Jedoch besitzt er nur das theoretische Wissen, praktische Erfahrungen hatte er noch nie gesammelt.

2.4 Fazit Benutzermodellierung

Zusammenfassend lässt sich sagen, dass die wichtigsten Benutzer des Systems

3 Benutzungsmodellierung

In diesem Kapitel soll die Benutzung des zu entwickelnden System analysiert werden. Dabei sollen einige Methoden der Mensch-Computer-Interaktion zum Einsatz kommen, um die Aufgaben des Benutzers exakt und detailliert untersu-

chen zu können. Anhand der zuvor erstellten Persona können zuerst Szenarien geschrieben werden, die die Aufgaben in einer narrativen Form darstellen.

3.1 Szenarien

3.1.1 Szenario 1: Düngungsprozess bei einem erfahrenen Landwirt

Das folgende Szenario beschreibt den Prozess von der Düngeplanung bis hin zur Ausbringung des Düngers mit der Personae “Nikolas Stein“.

Es ist Dienstag morgen. Nikolas Arbeitstag beginnt jeden Tag um 7:00 Uhr. Heute stehen wichtige Aufgaben auf seinen Ackerfeldern an. Die Felder müssen gedüngt werden. Zusammen mit seinem Mitarbeiter kümmert er sich um die Düngung der Ackerfelder. Vor ca. 4 Wochen hat Nikolas zusammen mit seinem Mitarbeiter Bodenproben von seinen Feldern entnommen und an ein Labor zur Bodenanalyse geschickt. Gestern sind die Ergebnisse dieser Bodenanalyse angekommen und er kennt nun den Zustand und den Nährstoffgehalt der einzelnen Felder. Mithilfe dieser Informationen ermittelt er selbständig die Düngermenge, die er auf seinen Feldern verteilen möchte. Dazu geht er in sein Büro und betrachtet seine zuvor dokumentierten Informationen über seine ganzen angebauten Pflanzen und vergangenen Ernten. Mithilfe seines Fachwissen, seiner Erfahrung und der Informationen von der letzten Ernte bestimmt Nikolas selbständig für jeden einzelnen Schlag eine Düngermenge, die dort ausgebracht werden muss. Diese Ergebnisse notiert er sich für jeden Schlag. Nachdem er die Düngerbedarfsermittlung abgeschlossen hat, schaut er in seinem Düngerlager nach, wie viel Dünger und welche Düngermittel er noch zur Verfügung hat. Er merkt, dass er nicht genügend Düngemittel im Lager hat, um seine ganzen Felder ausreichend düngen zu können. Deshalb muss er zusätzlich Düngemittel dazukaufen, die er anschließend mit seinem vorhandenem Dünger mischen muss, um das optimale Nährstoffverhältnis des Düngers für die einzelnen Schläge erreichen zu können. Die Berechnung, wie er jeweils mischen muss, führt er mithilfe des Computers durch. Den gemischten Dünger füllt er und sein Mitarbeiter anschließend in einen Düngerstreuer. Mit diesem Düngerstreuer fährt er mit seinem Traktor über die Felder, um den Dünger auszubringen. Dabei kann er durch Erhöhung bzw. Reduzierung der Fahrgeschwindigkeit die Ausbringmenge selber bestimmen.

Claim Analysis

3.1.2 Szenario 2: Düngungsprozess bei einem Landwirt mit wenig Erfahrung

Das folgende Szenario beschreibt den Prozess von der Düngeplanung bis hin zur Ausbringung des Düngers mit der Personae “Thomas Blaumann“.

Thomas hat vor kurzem mit seiner Familie den Familienbetrieb erworben. Durch

seine Besuche der Abendkurse über Landwirtschaft hat er grundlegendes Wissen in Bezug auf ackerbauliche Vorgänge erworben. Weiteres Wissen hat er sich durch Selbststudium und Internetrecherchen weiterangeeignet. Dadurch war ihm bekannt, dass eine Bodenuntersuchung für seine ackerbauliche Arbeit erforderlich war, bevor er überhaupt etwas mit dem Boden macht. Die Bodenuntersuchung vom Vorbesitzer war schon 7 Jahre her, weshalb Thomas eine neue Bodenuntersuchung durchführen lassen muss. Dazu entnimmt er mehrere Bodenproben von seinen Feldern. Da für eine gute Bodenuntersuchung eine exakte und sorgfältige Bodenentnahme wichtig ist, achtet Thomas auf alle Kleinigkeiten, die er gelernt hat. Die entnommenen Bodenproben schickt er anschließend an ein Bodenanalyselabor. Zwei Wochen später liegen ihm die Ergebnisse der Bodenuntersuchung vor. Er weiß nun, wie der Nährstoffgehalt seiner Felder aussieht und wie der Zustand des Bodens ist.

Claim Analysis

3.2 Hierarchical Task Analysis

3.3 Essential Use Cases

3.4 Concrete Use Cases

3.5 Fazit Benutzungsmodellierung

4 Anforderungen

Auf Grundlage der Benutzer- und Benutzungsmodellierung werden nun Anforderungen an das System abgeleitet und spezifiziert. Diese Anforderungen sollen in funktionale, qualitative und organisationale Anforderungen unterteilt werden. Sie sind die Basis für die Erstellung des UI Prototyps und werden mit anschließender Evaluation iterativ weiterentwickelt, bis sie als zufriedenstellend erfüllt angesehen werden.

4.1 funktionale Anforderungen

4.2 qualitative Anforderungen

- Das System soll eine hohe Gebrauchstauglichkeit liefern.
- Das System muss orts- und zeitunabhängig verwendet werden können.
- Das System soll eine verlustfreie Datenübertragung zwischen Systemkomponenten gewährleisten.
- Das System muss die Genauigkeit und die fehlerfreie Übertragung der Bodendaten gewährleisten.

4.3 organisationale Anforderungen

5 Gestaltungslösung / UI Prototyp

5.1

6 UI-Evaluation

In diesem Abschnitt wird der zuvor erstellte UI Prototyp evaluiert, um zu überprüfen, inwieweit die erstellten Anforderungen zufriedenstellend erfüllt wurden. Im Konzept wurde überlegt, mit echten potentiellen Nutzer zu testen. Im Verlauf des Projekts jedoch konnte das Team keinen Kontakt zu möglichen echten Nutzern herstellen, so dass das Testen mit ihnen nicht möglich ist. Deshalb soll wie im Konzept schon erwähnt aufgrund nicht verfügbarer Ressourcen eine Evaluation mit Personen durchgeführt werden, die moderate bis fortgeschrittene MCI-Kenntnisse verfügen. Die Evaluation des Prototypen soll durch eine heuristische Evaluation erfolgen. Dafür werden zum einen die Heuristiken benötigt, auf deren Grundlage getestet wird, und zum anderen die Aufgabenbeschreibungen, die als wesentlicher Leitfaden für die Durchführung der Evaluation dienen. Im Folgenden sollen diese aufgelistet und beschrieben werden:

- **visibility of system status**

Der Benutzer soll das System untersuchen und zu jeder Zeit in der Lage sein, zu erkennen, was gerade im System geschieht. Er sollte deshalb auch in der Lage sein, zu jeder Zeit wiedergeben zu können, wo er sich im System bzw. in welchem Systemstatus er sich befindet.

- **aesthetic and minimalist design**

Das Design des Systems sollte für die Wahrnehmbarkeit des Benutzers auf die nötigsten Informationen beschränkt werden. Das System soll keine irrelevanten Informationen enthalten. Diese Heuristik lässt sich beim Anlegen eines neuen Schlags gut testen.

- **user control and freedom**

Dem Benutzer wird ein Systemstatus vorgelegt, in dem er sich nicht befinden soll. Er muss diesen Fehler registrieren und durch bestimmte Interaktionen in den richtigen Systemstatus gelangen.

- **consistency and standards**

Der Benutzer soll durch die Untersuchung des Systems auf Begriffe, Symbole und Aktionen achten und wiedergeben können, ob Konsistenz gewahrt wurde und Konventionen eingehalten wurden sind. Er sollte sich nicht fragen, ob verschiedene Begriffe oder Symbole dasselbe bedeuten.

- **error prevention**

Hier wird verlangt, dass durch ein sorgfältiges Design, Fehler “verhütet” werden und gar nicht auftreten sollten. Das Verhalten des Benutzers sollte beim Untersuchen des Systems beobachtet werden. Wo treten bei ihm oft

Fehler auf? Welche Fehler lassen sich durch Überarbeitung des Designs vermeiden?

- **recognition rather than recall**

Der Benutzer sollte zwischen verschiedenen Screens keine Informationen merken müssen. Da in unserem System beim Anlegen und Bearbeiten der Schlag-Daten mit den meisten Informationen gearbeitet wird, sollte hier getestet werden, inwieweit das Design vom Benutzer verlangt, sich Informationen merken zu müssen.

- **flexibility and efficiency of use**

Das System sollte für Anfänger, aber auch für Experten gestaltet werden. Daher ist es wichtig, mögliche Interaktionsschritte bspw. durch Shortcuts zu beschleunigen. Da das System als eine mobile Anwendung realisiert werden soll, können auf mobilen Endgeräte keine Shortcuts genutzt werden. Während der Evaluation sollte beobachtet werden, wie der Benutzer sich durch das System navigiert. Dadurch könnten Erkenntnisse gewonnen werden, ob und welche Interaktionsschritte verbessert werden können.

- **recognition, diagnosis and recovery from errors**

Aufgrund der Tatsache, dass der UI-Prototyp in Form eines paperbased Prototyps erstellt wurde, ist es nicht möglich, detaillierte Fehlermeldungen zu erzeugen und zu testen, inwieweit sie dem Benutzer helfen. Deshalb kann für die Evaluation keine Aufgabe zu dieser Heuristik gestellt werden.

- **help and documentation**

Jede Information der Hilfe oder Dokumentation sollte leicht zu finden sein, auf die Aufgabe abgestimmt sein und die konkreten Schritte zur Lösung auflisten.[[3]] Im Rahmen der Evaluation ist noch keine Dokumentation bzw. Hilfestellung fertiggestellt, weshalb diese noch nicht getestet werden kann.

- **match between system and real world**

Der Benutzer soll anhand der “Schlag-Interfaces“ wiedergeben können, ob die verwendeten Begriffe seiner Sprache entsprechen. Es sollen keine systemorientierte Terminologie benutzt werden, sondern Worte, Phrasen und Konzepte, mit denen der Benutzer vertraut ist.

Evaluationsergebnisse und Erkenntnisse

7 Systemarchitektur

Die Abbildung 1 stellt die geplante Systemarchitektur der zu entwickelnden Anwendung grafisch dar. Im folgenden Abschnitt sollen alle wichtigen Merkmale der Architektur erläutert werden. Dabei werden die einzelnen Komponenten und deren Kommunikation miteinander betrachtet und die ausgewählten Technologien beschrieben und abgewägt. Außerdem sollen Veränderungen und Überarbeitungen der Architektur im Vergleich zum Konzept erläutert werden.

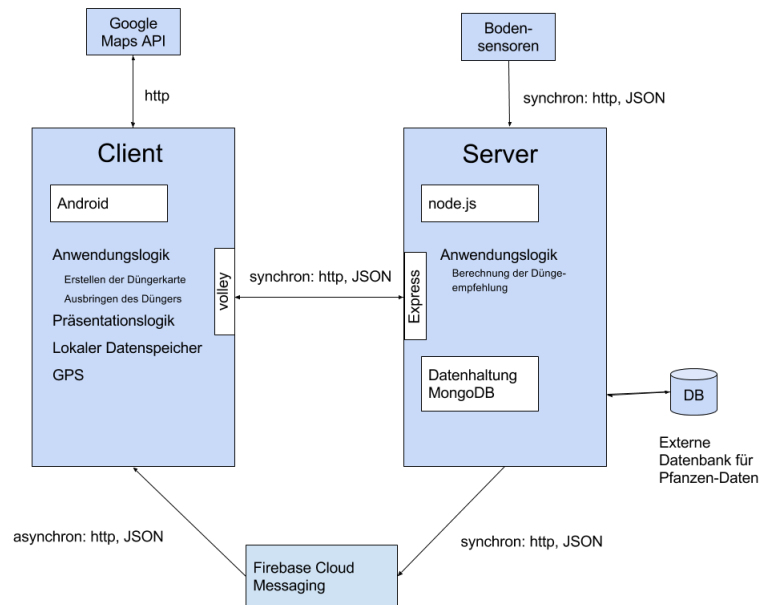


Abbildung 1: neues Architekturdiagramm

7.1 Client

Die geplante Anwendung soll als mobile Anwendung im REST-Stil realisiert, da die Mobilität für den Landwirten bei der Düngung eine große Rolle spielt. Durch das mobile Endgerät können Daten wie GPS-Standort gesammelt und verarbeitet werden. Im Rahmen des Moduls soll die Entwicklung der mobilen Anwendung in Android mit Java erfolgen. Mithilfe der volley-Bibliothek soll die Client-Server Kommunikation implementiert werden. Mit ihr ist es möglich, HTTP-Requests zu erzeugen und sie zu behandeln. Außerdem befindet sich auf dem Client ein lokaler Datenspeicher, in dem Daten für die Erstellung der Düngerkarte zwischengespeichert werden können, so dass der Client nicht ständig Anfragen an den Server senden muss.

7.2 Server

Für die Implementierung des Servers wurde für die Programmiersprache JavaScript mit dem Framework node.js entschieden, da zum einen fundierte Kenntnisse in dieser Sprache vorliegen und zum anderen node.js eine hohe Skalierbarkeit aufweist, wodurch Anpassungen einfach und ohne Probleme durchgeführt werden können. Die Datenhaltung auf dem Server soll mithilfe der dokumentenorientierte Datenbank MongoDB geschehen. Außerdem soll der Server eine

Anbindung an eine externe Datenbank für Pflanzen-Daten haben. Die Pflanzen-Daten sind Grundlage für die Berechnung der Düngeempfehlung. Aufgrund der Tatsache, dass keine geeigneten Pflanzendatenbanken gefunden worden sind, soll diese durch Testdatensätze simuliert werden.

7.3 Datenformat und Übertragungsprotokoll

Für die geplante verteilte Anwendung ist eine Realisierung im REST-Stil vorgesehen. Deshalb soll die Datenübertragung zwischen den einzelnen Systemkomponenten über das Transportprotokoll http erfolgen. Die Daten, die zwischen den Komponenten ausgetauscht werden, sollen im Datenformat JSON (JavaScript Object Notation) repräsentiert werden, da sie einfach zu handhaben sind und aufgrund der dokumentenorientierten Eigenschaft der Datenbank MongoDB dort gut gespeichert werden können.

7.4 Externe Schnittstelle

Auf der Abbildung 1 ist zu erkennen, dass die Google Maps API an den Client angebunden ist. Sie wird unter anderem für die Darstellung der Satellitenkarte und für geographische Funktionen benötigt. Die Daten, die zwischen der Schnittstelle und dem Client ausgetauscht werden, werden ebenfalls durch http übertragen.

7.5 Bodensensoren

Zum System gehören Bodensensoren, die die Nährstoffwerte im Boden messen und an den Server senden. Dadurch kann der Landwirt ohne Untersuchung im Labor, den Nährstoffgehalt im Boden selber bestimmen. Diese Bodensensoren finden eine Anbindung an den Server. Mithilfe von http sollen die Bodendaten im JSON-Format von den Bodensensoren an den Server gesendet werden. Dadurch ist es dem Landwirten möglich, eine Düngeberechnung nach dem Prinzip der teilflächenspezifischen Bewirtschaftung durchzuführen. Die Bodensensoren sollen im Intervall von einem Tag den Boden messen und die Ergebniswerte an das System senden, da die Nährstoffwerte im Boden sich nicht schnell verändern.

7.6 Firebase Cloud Messaging[1]

Im Vergleich zur Architektur, die im Konzept erstellt wurde, wurde die neue Systemarchitektur durch den Firebase Cloud Messaging Service ergänzt. Der Service sorgt für die Realisierung der asynchronen Kommunikation im System. Der Landwirt soll vom System benachrichtigt werden, wenn der Nährstoffgehalt im Boden durch die Nährstoffaufnahme der Pflanzen unter einem gewissen Schwellwert sinkt und dadurch neu gedüngt werden müsste. Diese Push-Notification soll durch die Nutzung vom FCM realisiert werden.

8 Proof of Concept

Bereits im Konzept wurden Proof of Concepts aus den Projektrisiken abgeleitet und spezifiziert. Durch die Realisierung des Rapid Prototyping wurden die Machbarkeit einiger Proof of Concepts gezeigt. Durch das Hinzufügen des Firebase Cloud Messaging Service in das System muss dafür ebenfalls ein neues Proof of Concept spezifiziert, getestet und dokumentiert werden. Im Folgenden sollen der neue Proof of Concept und die vorhandenen Proof of Concept nochmals aufgelistet und die Durchführung dieser dokumentiert werden.

8.1 Einbindung einer externen Schnittstelle

Beschreibung: Für die Erkennung und Erfassung der einzelnen Felder bzw. Schläge der Landwirte werden Karten in Form von Satellitenbildern benötigt. Die Google Maps API bietet diese Form von Daten an.

Exit-Kriterium: Die Google Maps API wird vom Android Client erfolgreich erkannt. Die Daten, die von der API gesendet werden, können in Form von Satellitenbildern auf dem Client angezeigt werden. Das Exit-Kriterium wird als erfolgreich angesehen, wenn Satellitenbilder von 10 verschiedenen Orten auf dem Client angezeigt werden können.

Fail-Kriterium: Das Fail-Kriterium tritt ein, wenn das Abrufen und Darstellen der Satellitenbilder auf dem Client nicht möglich ist.

Fallback: Es wird auf eine andere Schnittstelle zugegriffen, die Daten in Form von Satellitenkarten zur Verfügung stellt. Die Einbindung der Bing Maps API soll im Fall eines Fails erfolgen.

Durchführung: Für die Überprüfung dieses Proof of Concepts wurde im Rapid Prototyping ein Android Client angelegt, der nur aus einer **Main-Activity** und einer **Map-Activity** bestand. Damit die Google Maps API genutzt werden konnte, musste vorerst ein API-Schlüssel von Google angefordert werden. Dieser API-Schlüssel wurde anschließend zur Anwendung hinzugefügt, indem dieser in die **AndroidManifest.xml** kopiert wurde. Dadurch wurde die Grundvoraussetzung zur Nutzung der Google Maps API erfüllt.

Durch die Erstellung der **Maps-Activity** konnte eine von Google bereits erstellte Code-Vorlage genutzt werden. In der **onCreate-Funktion** wird der Anwendung gesagt, dass die Activity in dem Google Maps-Layout dargestellt werden soll. In der **onMapReady-Funktion** können Einstellungen an der Map vorgenommen werden. Für das Proof of Concept wurde der Parameter in **setMapType** von **ROADMAP** zu **SATELLITE** gewechselt, damit mit der Satellitenkarte gearbeitet werden konnte. Anschließend konnte in der Anwendung verschiedene Satellitenbilder von zehn verschiedenen Orten gezeigt werden, weshalb das Exit-Kriterium eingetreten ist.

8.2 Ermittlung des Standortes durch GPS in Google Maps

Beschreibung: Die Ermittlung des Standortes ist in der Domäne der Landwirtschaft eine essentielle Funktion, um exakt arbeiten zu können. Sie ist ebenfalls wichtig, um mit den Daten aus der Google Maps API effizient arbeiten zu können.

Exit-Kriterium: Die Ermittlung des Standortes soll mithilfe des GPS des Smartphones an 20 verschiedenen Orten getestet werden. Das Exit-Kriterium tritt ein, wenn alle 20 Orte vom Smartphone in Google Maps richtig geortet werden können.

Fail-Kriterium: Sobald eines der Orte nicht geortet werden kann, gilt dieses Proof-of-Concept als fehlgeschlagen.

Fallback: Beim Eintreten des Fail-Kriteriums muss auf die Nutzung von GPS durch das mobile Endgerät verzichtet werden, was für das Projekt gravierende Folgen haben wird, denn wie bereits beschrieben ist das Arbeiten mit dem Standort in der teilflächenspezifischen Bewirtschaftung eine Voraussetzung.

Durchführung: Für die Durchführung dieses Proof of Concept wurde das zuvor realisierte Rapid Prototyping als Testgrundlage genommen, da dort die Implementierung der Google Maps API in das System gelungen war. Die Anwendung hatte bis dahin nur die Satellitenkarte in der **Map-Activity** dargestellt. Weitere Einstellungen wurden noch nicht durchgeführt. Damit es möglich wurde, den Standort des Nutzers auf der Satellitenkarte anzeigen zu lassen, mussten zuerst Standort-Berechtigungen vom Nutzer angefordert werden. Erst durch Akzeptieren dieser Berechtigungen konnte mit der Funktion `setMyLocationEnabled(true)` der Standort des Nutzers in Google Maps ermittelt werden. Dadurch war es nun möglich, mithilfe der Anwendung an 20 verschiedenen Orten in der Nähe von Gummersbach den Standort über Google Maps ermitteln zu lassen. Alle 20 ermittelten Standorte war exakt und korrekt ermittelt, so dass dieses Proof of Concept als erfolgreich getestet gilt.

8.3 Erstellen von Polygonen auf den Satellitenbildern

Beschreibung: Für die Erstellung von Nährstoffkarten auf bestimmten Schlägen bzw. Feldern ist das Erstellen von Polygonen auf den Satellitenbildern eine Voraussetzung.

Exit-Kriterium: Funktioniert das Erstellen von 10 verschiedenen Polygonen mit verschiedenen Standorten in Google Maps einwandfrei, tritt das Exit-Kriterium ein.

Fail-Kriterium: Das Proof-of-Concept gilt als fehlgeschlagen, wenn keine bzw. keine 10 verschiedene Polygone erstellt werden können.

Fallback: Im Fall eines Fails muss auf eine andere Schnittstelle gewechselt werden, die zum einen Satellitenkarten zur Verfügung stellen und zum anderen die Möglichkeit bietet, Polygone erzeugen zu können. Hier würde wieder auf die Bing Maps API zurückgegriffen werden.

Durchführung: Die Erstellung der Polygone in Google Maps wurde in der Maps-Activity durchgeführt. Dazu wurde ein Objekt des Typen `PolygonOptions` erzeugt, das die einzelnen Einstellungen zu den Polygonen beinhaltet. In diesem Objekt wurden Koordinaten definiert, die Grundlage für die Erzeugung des Polygons dienen. Mit der Funktion `Polygon polygon = mMap.addPolygon(rectOptions)` wurden nun die zuvor definierten Einstellungen genommen und damit ein Polygon auf der Karte erzeugt. Durch die Veränderung der Koordinaten in den `rectOptions` konnten andere Polygone erstellt werden. Das Exit-Kriterium ist eingetreten, da es uns gelungen ist, zehn verschiedene Polygone auf der Google Maps Satellitenkarte zu erzeugen.

8.4 Kommunikation zwischen Systemkomponenten

Beschreibung: Eine verteilte Anwendung kann nur funktionieren, wenn die einzelnen Komponenten der Anwendung miteinander kommunizieren können. Die Kommunikation soll wie vorher entschieden nach dem Client-Server Paradigma funktionieren.

Exit-Kriterium: Das Exit-Kriterium tritt ein, wenn 5 Datensätze mithilfe der POST-Methode vom Client an den Server gesendet werden kann und diese persistent in der Datenbank gespeichert werden können. Außerdem müssen diese Datensätze im Client durch die GET-Methode geholt und dargestellt werden können.

Fail-Kriterium: Das Proof-of-Concept gilt als fehlgeschlagen, wenn die Datensätze nicht in der Datenbank festgeschrieben werden können oder diese Datensätze nicht auf dem Client dargestellt werden können.

Fallback: Sollte das Fail-Kriterium eintreten, gilt das Projekt als gescheitert, weil dieses Paradigma für die Realisierung des Systems unabdingbar ist.

Durchführung: Durch die Realisierung der Android Anwendung wurde der Client bereits implementiert. Dieser hatte jedoch noch keine Funktionen, die eine Client-Server-Kommunikation behandelt. Deshalb wurde eine neue Activity erstellt (Post-Activity). Diese beinhaltet eine TextView, die die Response des Servers anzeigen soll, und einen Button, der durch das Drücken einen POST-Request vom Android Client an den Server senden soll. Mithilfe der volley-Bibliothek ist es möglich, schnell und einfach HTTP-Requests durchzuführen. In der TextView wurde ein GET-Request realisiert, der an den Server unter der Ressource `polygon` gesendet wird und auf eine Antwort des Servers wartet. In

der Methode `postMessage` wurde ein statisches JSON-Objekt erstellt, der Koordinaten für die Erzeugung eines Polygons enthielt. Dieses Objekt wurde durch ein POST-Request an den Server gesendet, sobald der Button getätigt wurde.

Der Server wurde in JavaScript mit dem Framework `node.js` implementiert. Mit dem `node-module` “`express`” konnten Routen erstellt werden, die auf HTTP-Request auf bestimmte Ressourcen reagieren. Für die Ressource `polygon` wurden zwei Methoden implementiert:

- **GET-Request**

Der Server sucht in der MongoDB-Datenbank nach Daten, die in der Collection `polygon` gespeichert wurden, greift anschließend auf diese zu und sendet dem Client die angefragten Daten. Ist in der Datenbank nichts gespeichert, wird eine leere Response gesendet.

- **POST-Request**

Durch ein POST-Request des Clients reagiert der Server damit, diese aufzuarbeiten, auf der Konsole auszugeben und anschließend in der MongoDB-Datenbank zu speichern. Dadurch wäre es dem Client möglich, durch ein GET-Request diese zuvor gespeicherten Daten anzufragen.

Wie genau die für das Proof of Concept erstellte Anwendung funktioniert, kann im Konzept im Kapitel *Rapid Prototyping* (siehe Anhang) nachgelesen werden.

8.5 Nutzung des Firebase Cloud Messaging Service

Beschreibung: Der Landwirt soll vom System benachrichtigt werden, wenn der Nährstoffgehalt des Bodens durch die Nährstoffaufnahme der Pflanzen unter einem gewissen Schwellwert sinkt und dadurch eine neue Düngeempfehlung berechnet wird. Dies soll durch Push-Notifications mithilfe des FCM-Services realisiert werden.

Exit-Kriterium: Es sollen vom Server zehn verschiedene Nachrichten zu zehn verschiedenen Zeitpunkten gesendet werden. Wenn der Client alle zehn Nachrichten erhalten hat und diese darstellen kann, gilt das Proof of Concept als erfolgreich.

Fail-Kriterium: Es können nicht alle zehn Nachrichten auf dem Client angezeigt werden.

Fallback: Die Realisierung mit Push-Notifications wird für die geplante Anwendung gestrichen.

Durchführung: Für die Durchführung dieses Proof of Concept wurde ebenfalls das Rapid Prototyping als Grundlage zum Testen benutzt. Damit der Server dem Client über FCM eine Nachricht schicken kann, müssen auf beiden Systemkomponenten FCM eingerichtet werden. Im Folgenden sollen für beide die Einrichtung getrennt beschrieben werden:

- **Client**

Auf Android müssen zwei Java-Klassen `MyFirebaseInstanceIdService` und `MyFirebaseMessagingService` implementiert werden. In der `MyFirebaseInstanceIdService` wird zum einen der Registrierungstoken erzeugt und an den Server gesendet und dort gespeichert. In der `MyFirebaseMessagingService` wird definiert, was bei Erhaltung einer Nachricht geschehen soll. Hier soll nach Erhalt der Nachricht eine Push-Notification erzeugt und dargestellt werden.

- **Server**

Auf dem node.js Server muss zuerst das module `fcm` importiert werden. Dadurch, dass der Client dem Server den Registrierungstoken sendet, muss im Server eine Funktion implementiert werden, der diesen Token empfängt und zwischenspeichert. Außerdem muss ein Message-Objekt erzeugt werden, in dem die Nachricht mit weiteren Einstellungen enthalten ist.

Nach Einrichtung des FCM-Services auf Client und Server kann nun mittels des Server, Nachrichten an den Client gesendet werden, die dort als Push-Notifications angezeigt werden. Es wurden zehn verschiedene Nachrichten zu verschiedenen Zeitpunkten gesendet. Diese konnten als Push-Notifications auf dem mobilen Endgerät angezeigt werden. Deshalb gilt dieses Proof of Concept als erfolgreich getestet.

9 Datenstrukturen

Im folgenden Abschnitt werden alle modellierten Datenstrukturen des Systems abgebildet. Schon früher im Projektverlauf wurde für die Nutzung des Datenformats JavaScript Object Notation (JSON) entschieden. Deshalb werden die folgenden Datenstrukturen in Form von JSON präsentiert und beschrieben.

9.1 Pflanzen

Die Daten zu den Nährstoff-Sollwerten von den angebauten Pflanzenkulturen sind für die Düngeberechnung eine Voraussetzung. Da im Rahmen des Projekts keine externe Pflanzen-Datenbank gefunden wurde, die die Daten kostenlos zur Verfügung stellt, wurde entschieden, für den vertikalen Prototypen eine eigene Datenbank zu implementieren, die Testdatensätze zu den Pflanzenkulturen mit den notwendigen Informationen enthalten soll. Für die Düngeberechnung werden die Nährstoff-Sollwerte der Pflanzen unter dem Key "nutrient" aus der Datenbank genutzt.

```
1 {  
2   id: number,  
3   name: string,  
4   nutrient: {
```



```

5         nitrogen: number,
6         phosphorus: number,
7         potassium: number
8     }
9 }

```

9.2 Feld

Die Datenstruktur “Feld” soll alle wichtige Informationen zu den einzelnen Feldern des Landwirten beinhalten, die für das System von Nutzen sind. Besonders die Umrisse der Felder sind Grundlage für eine standortspezifische Düngeberechnung. Sie sollen in einem Array gespeichert werden, der die einzelnen Koordinaten eines Umrisses beinhaltet.

```

1 {
2     id: number,
3     name: string,
4     size: number,
5     outline: [object] array,
6     plant: number
7 }

```

9.3 Bodendaten

Die Bodensensoren messen den Nährstoffgehalt des Bodens. Der Standort, an dem die Messung stattgefunden hat, ist für das System wichtig. Deshalb werden in der Datenstruktur der Bodendaten zum einen der Nährstoffgehalt des Bodens an der gemessenen Stelle und zum anderen den Standort der Messung in Form von Koordinaten festgehalten.

```

1 {
2     nutrient: {
3         nitrogen: number,
4         phosphorus: number,
5         potassium: number
6     },
7     location: {
8         latitude: number,
9         longitude: number
10    }
11 }

```

9.4 Düngempfehlung

Die Düngempfehlung wird aus den Werten der Bodendaten und den Nährstoff-Sollwerten der Pflanzenkulturen berechnet. Sie beinhaltet die Nährstoffwerte, die der Landwirt dem Boden hinzufügen muss, um den Pflanzen ein optimales Wachstum bieten zu können.

```
1      {
2          nutrient: {
3              nitrogen: number,
4              phosphorus: number,
5              potassium: number
6          },
7          location: {
8              latitude: number,
9              longitude: number
10         }
11     }
```

9.5 Firebase Cloud Messages

Firebase Cloud Messages werden vom System genutzt, wenn der Nährstoffgehalt des Bodens durch die Nährstoffaufnahme der Pflanzen unter einem bestimmten Schwellwert sinkt, so dass eine neue Düngempfehlung berechnet wird. Dies wird durch die Untersuchung des Bodens mit den Bodensensoren erkannt. Der Landwirt soll daraufhin vom System eine Benachrichtigung bekommen, dass an dem bestimmten Standort eine neue Düngempfehlung berechnet wurde und er dort neu düngen sollte.

```
1      {
2          to: string,
3          notification: {
4              body: string
5          }
6      }
```

10 Architekturmerkmale

10.1 REST-Spezifikation

Alle Ressourcen, die im Projekt eine Rolle spielen und modelliert werden, sollen im Folgenden bzgl. ihrer HTTP-Methoden, ihrer Semantik, ihres Content-Types bei einem Request und bei einem Response spezifiziert werden.

10.2 Anwendungslogik

10.2.1 Berechnung einer Düngeempfehlung

Für die Berechnung der Düngeempfehlung werden zum einen die Bodendaten, die durch die Analyse des Boden mit den Bodensensoren entstanden sind, und zum anderen die Nährstoff-Sollwerte der angebauten Pflanzenkulturen benötigt. Im Folgenden soll der Ablauf der Berechnung der Düngeempfehlung beschrieben werden:

1. Der Landwirt definiert seinen Schlag (Informationen und Umriss) und welche Pflanzenkultur er auf diesem Schlag anbauen will.
2. Die Bodensensoren senden dem Server, den Nährstoffgehalt des Boden an verschiedenen Standorten dieses Schlags in Form von Bodendaten.
3. Der Server greift auf die in der externen Datenbank gespeicherten Pflanzendaten zu und überprüft, welche Pflanzenkultur er für die Berechnung der Düngeempfehlung nehmen muss.
4. Der Server berechnet nun basierend auf den Nährstoff-Sollwerten der zu düngenden Pflanzenkultur und auf den ermittelten Bodenwerten eine Düngeempfehlung für einen bestimmten Standort auf dem Schlag.
5. Diese Berechnung erfolgt für alle Standorte auf dem Schlag, die von den Bodensensoren gemessen wurden sind.

10.2.2 Erstellen der Düngerkarte bzw. Erstellen von Teilflächen

Für das Düngen nach dem Prinzip der teilflächenspezifischen Bewirtschaftung müssen auf dem Schlag basierend auf den Nährstoffwerten bzw. Düngeberechnung Teilflächen gebildet werden. Durch den zuvor definierten Umriss des Schlags durch den Landwirten soll es dem System möglich sein, auf Grundlage dieses Umrisses Teilflächen zu erstellen. Diese Teilflächen entstehen

10.2.3 Ausbringen des Düngers basierend auf der Düngeempfehlung

Die vom Server berechneten Düngeempfehlungen müssen für den Landwirten optimal verwertet werden können. Deshalb ist es wichtig, dem Landwirten ebenfalls beim Ausbringen des Düngers zu unterstützen.

Abbildungsverzeichnis

1	neues Architekturdiagramm	17
---	-------------------------------------	----

Tabellenverzeichnis

1	Stakeholder-Analyse	5
2	User Profile 1: erfahrener Landwirt	8
3	User Profile 2: landwirtschaftlicher Mitarbeiter	9
4	User Profile 3: angehender Landwirt	10

Literatur

- [1] Firebase cloud messaging. <https://firebase.google.com/>. abgerufen am: 03.12.2016.
- [2] Din en iso 9241 teil 210 prozess zur gestaltung gebrauchstauglicher interaktiver systeme, 2008.
- [3] Werner Schweibenz and Frank Thissen. *Qualität im Web: Benutzerfreundliche Webseiten durch Usability-Evaluation*. Springer, Berlin, 1 edition, 2002.

Anhang