

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

5. 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	8
2.2.3 User Profile 3: angehender Landwirt	9
2.3 Persona	10
2.3.1 Personae 1: Nikolas Stein	10
2.3.2 Personae 2: Emily Solay	11
2.3.3 Personae 3: Thomas Blaumann	12
2.4 Fazit Benutzermodellierung	13
3 Benutzungsmodellierung	14
3.1 Szenarien	14
3.1.1 Szenario 1: Düngungsprozess bei einem erfahrenen Landwirt	14
3.1.2 Szenario 2: Düngungsprozess bei einem Landwirt mit wenig Erfahrung	15
3.2 Hierarchical Task Analysis	16
3.3 Essential Use Cases	17
3.4 Concrete Use Cases	19
3.5 Fazit Benutzungsmodellierung	20
4 Anforderungen	21
4.1 funktionale Anforderungen	21
4.2 qualitative Anforderungen	22
5 UI Prototyp	23
6 UI-Evaluation	26
7 Systemarchitektur	29
7.1 Client	29
7.2 Server	30
7.3 Datenformat und Übertragungsprotokoll	30
7.4 Externe Schnittstelle	30
7.5 Bodensensoren	30
7.6 Firebase Cloud Messaging	31

8 Proof of Concept	31
8.1 Einbindung einer externen Schnittstelle	31
8.2 Ermittlung des Standortes durch GPS in Google Maps	32
8.3 Erstellen von Polygonen auf den Satellitenbildern	33
8.4 Kommunikation zwischen Systemkomponenten	33
8.5 Nutzung des Firebase Cloud Messaging Service	34
9 Datenstrukturen	36
9.1 Pflanzen	36
9.2 Feld	36
9.3 Bodendaten	37
9.4 Düngerempfehlung	37
9.5 Firebase Cloud Messages	38
10 Architekturmerkmale	38
10.1 REST-Spezifikation	38
10.2 Anwendungslogik	40
10.2.1 Berechnung einer Düngerempfehlung	40
10.2.2 Erstellen der Düngerkarte bzw. Erstellen von Teilflächen .	41
10.2.3 Ausbringen des Düngers basierend auf der Düngerempfehlung	41
10.2.4 Benachrichtigung, wenn gedüngt werden sollte	42
A Erstes UI-Prototyp	47
B Finales UI-Prototyp	49
C Evaluations-Protokoll	53
D Konzept	56

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. [3] 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 Düngereplanung, bei der Dünge mengenberechnung 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.“ [[4]] 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

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"> - Landwirtschaftsmaschinen (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

2.2.2 User Profile 2: landwirtschaftlicher Mitarbeiter

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"> - Landwirtschaftsmaschinen (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

2.2.3 User Profile 3: angehender Landwirt

Tabelle 4: User Profile 3: angehender Landwirt

Merkmale	Merkmalsausprägung
Stakeholder	Landwirt
Alter	18-39
Geschlecht	männlich / weiblich
Wohnort	deutschlandweit
Berufserfahrung	<ul style="list-style-type: none"> - angehender Landwirt - kaum Erfahrung mit landwirtschaftlichen Arbeitsvorgängen
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 sehr gut - tägliche private und arbeitsbedingte Nutzung
Fachwissen	<ul style="list-style-type: none"> - Wissen über einige landwirtschaftliche Arbeitsvorgänge - Wissen über Düngung
Arbeitsmittel	<ul style="list-style-type: none"> - Landwirtschaftsmaschinen (Düngemaschine, Mähdrescher, 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	<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"> - offen für neue Technologien, die ihm bei der Arbeit unterstützen - Begeisterung in Nutzen von neuen Technologien

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.



Abbildung 1: Nikolas Stein

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 Jahren 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 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.



Abbildung 2: Emily Solay

Name: Emily Solay

Alter: 35

Geschlecht: weiblich

Familienstatus: geschieden und eine Tochter

Beruf: Angestellte in einem Ackerbaubetrieb

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.



Abbildung 3: Thomas Blaumann

Name: Thomas Blaumann

Alter: 30

Geschlecht: männlich

Familienstatus: ledig

Beruf: angehender Landwirt

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 die Personengruppen sind, die direkt in der Landwirtschaft arbeiten, also landwirtschaftliche Arbeitsprozesse durchführen. Dazu zählen Landwirte, angehende Landwirte, landwirtschaftliche Mitarbeiter oder Familienmitglieder von Landwirten, die bei der Arbeit des Landwirte mithelfen könnten.

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 untersuchen 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 Dünegeplanung 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 selbstständig 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 selbstständig 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 Düngemittel 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

- ++ Düngermenge mithilfe der Bodenanalyse ermitteln
- Für die Berechnung der Düngermenge ist er an seinem Computer
- Er muss eigenständig die Düngermenge berechnen
- Er muss selber bestimmen, was und wie viel Dünger er mischen muss
- Er muss bis zu vier Wochen auf die Ergebnisse der Bodenanalyse warten

3.1.2 Szenario 2: Düngungsprozess bei einem Landwirt mit wenig Erfahrung

Das folgende Szenario beschreibt den Prozess von der Dünegeplanung bis hin zur Ausbringung des Dünger 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. Mit diesem Wissen ist es ihm möglich, eine Düngermenge für seine Felder zu bestimmen. Seine Felder hat er vorher schon genauer studiert und sie in Schläge eingeteilt. Auf den einzelnen Schlägen hat er sich überlegt, welche Pflanzenkulturen er dort anbauen möchte. Da er von Pflanzen bisher keine Erfahrung hat, muss er sich anderweitig über diese informieren. Bspw. benötigt er Angaben darüber, wie er ihnen optimale Wachstumsbedingungen bieten könnte. Durch seine Recherche kennt er nun die Nährstoff-Sollwerte der Pflanzenkulturen, die er anbauen möchte. Mithilfe dieser Werte und den Ergebnisse der Bodenanalyse lässt er sich von einem Programm, das er im Internet entdeckt hatte, die Düngemengenberechnung durchführen, weil er selber nicht weißt, wie das funktioniert. Deshalb muss er sich für die Düngung an den Ergebnissen dieses Programms richten.

Claim Analysis

- +++ Er teilt seine Felder in Schlägen ein und sich Gedanken gemacht, welche Pflanzen er anbauen möchte
- ++ Die Durchführung einer Bodenuntersuchung
- ++ Er nimmt sich Software/Programme zu Hilfe, die ihm bei der Arbeit unterstützen
- Er muss die Bodenproben für die Bodenuntersuchung selber entnehmen
- Er weiß nicht, ob das Programm die Berechnung vollständig und korrekt durchführt, weil er kaum Erfahrungen hat.
- Er muss Recherche über Pflanzen betreiben, um die Nährstoff-Sollwerte zu ermitteln.

3.2 Hierarchical Task Analysis

Im folgenden Abschnitt soll der aktuelle Vorgang der Düngung genauer untersucht und dokumentiert werden. Da der Düngungsprozess von der Vorbereitung bis hin zur Ausbringung des Düngers ein komplexer und aufwendiger Vorgang ist, soll dieser Prozess mithilfe der Hierarchical Task Analysis detaillierter analysiert werden.

Der Prozess der Düngung

0. Düngungsprozess

1. Bodenanalyse

- 1.1 Utensilien für die Bodenprobe aus dem Lager holen
 - 1.1.1 Skizzenheft holen
 - 1.1.2 Bohrstock holen
 - 1.1.3 Stift holen
 - 1.1.4 Plastikeimer holen
- 1.2 Probe entnehmen
 - 1.2.1 Feld skizzieren im Skizzenheft
 - 1.2.2 Bodenstichprobe entnehmen
 - 1.2.2.1 Den Bohrstock ca. 30cm tief in den Boden setzen
 - 1.2.2.2 Die Erde im Bohrstock in den Plastikeimer schütteln
 - 1.2.3 Die Stelle der entnommenen Probe im Skizzenheft markieren
 - 1.2.4 (Loop) Vorgang 1.2.2 + 1.2.3 fortsetzen bis eine genügende Anzahl von Bodenproben entnommen wurde
- 1.3 Bodenproben für den Versand zum Labor bereitmachen
 - 1.3.1 Mischen der entnommenen Bodenproben
 - 1.3.2 Gemischte Bodenproben in die Plastiktüte umfüllen
 - 1.3.3 Beschriftung der Plastiktüten vornehmen
 - 1.3.3.1 Schlagbezeichnung beschriften
 - 1.3.3.2 Probenummer hinzufügen
- 1.4 Probe versenden

2. Düngung

- 2.1 Düngemengenberechnung planen
 - 2.1.1 Ergebnisse der Bodenanalyse lesen
 - 2.1.1.1 Boden wurde in Nährstoffgehaltsklasse von A(niedriger Nährstoffgehalt)-E(hoher Nährstoffgehalt) eingestuft
 - 2.1.2 Ermittlung des Nährstoffentzuges bzw. des Nährstoffsbedarfes der Pflanzen
 - 2.1.2.1 Welche Pflanzen auf dem Schlag sollen gedüngt werden?
 - 2.1.2.2 Nährstoff-Sollwerte der angebauten Pflanzen betrachten
- 2.2 Berechnung der Düngemengenberechnung durchführen

3. Dünger ausbringen

- 3.1 Düngerstreuer vorbereiten
 - 3.1.1 Streutabelle ausdrucken
 - 3.1.2 Dosierschieber laut Streutabelle einstellen
 - 3.1.3 Breite der Ausstreuung (Arbeitsbreite) einstellen
 - 3.1.4 Drehzahl nach Streutabelle und Bedarf einstellen
 - 3.1.5 Düngermischung in den Behälter einschütten
- 3.2 Dünger auf Feld ausbringen
 - 3.2.1 Durch Änderung der Fahrgeschwindigkeit das Ausbringen der Düngermenge regulieren

Die Hierarchical Task Analysis hat viele Erkenntnisse gebracht. Von der Vorbereitung der Düngung bis hin zum Ausbringen des Düngers gibt es viele Aufgaben, die vom Landwirten erledigt werden müssen. Viele dieser Aufgaben könnten durch ein System zusammengefasst und vereinfacht werden. Für die zu entwickelnde Anwendung werden bspw. Bodensensoren eingeplant, die die Bodenanalyse des Landwirten vereinfachen und beschleunigen könnte. Außerdem könnte die Düngemengenberechnung vom System durchgeführt werden, so dass die Landwirte diese nicht selber ausführen müssen.

3.3 Essential Use Cases

Um die Aufgaben der Benutzer weiter zu beschreiben, sollen mithilfe von Use Cases die Aufgaben weiter untersucht werden. Mithilfe von Essential Use Cases ist es möglich, Aufgaben abstrahiert zu beschreiben, ohne Bezug zu Technologien zu nehmen.

Tabelle 5: Essential Use Case 1: Anlegen von Schlag-Daten

Intention des Benutzers	Systemverantwortlichkeit
Benutzer möchte seine Schlag-Daten festhalten.	Zeigt die Funktion zum Anlegen von Schlag-Daten an.
Benutzer spezifiziert die Schlag-Daten	Nimmt die Schlag-Daten des Benutzers entgegen.
Benutzer spezifiziert graphisch seinen Schlag.	Nimmt graphische Spezifizierung des Schlags entgegen.
Benutzer bestätigt die Eingaben und Spezifizierung der Daten.	Speichert die Schlag-Daten

Tabelle 6: Essential Use Case 2: Bearbeiten der Schlag-Daten

Intention des Benutzers	Systemverantwortlichkeit
Benutzer möchte seine Schlag-Daten bearbeiten.	Zeigt die geforderten Daten an.
Benutzer spezifiziert die neuen Schlag-Daten	Nimmt die Schlag-Daten des Benutzers entgegen.
Benutzer bestätigt die neue Eingabe der Daten.	Überschreibt die alten und speichert die neuen Schlag-Daten

Tabelle 7: Essential Use Case 3: Zuordnen von Pflanzen zu Schlägen

Intention des Benutzers	Systemverantwortlichkeit
Benutzer möchte zu seinem Schlag, Pflanzen zuordnen.	Zeigt ihm sein Schlag und eine Liste von Pflanzen an.
Benutzer wählt Pflanze aus, die er zum Schlag zuordnen will.	Ordnet Pflanze zum Schlag zu.
Benutzer bestätigt die Auswahl.	Speichert die ausgewählte Pflanze zum Schlag.

Tabelle 8: Essential Use Case 4: Standort des Benutzers erfassen

Intention des Benutzers	Systemverantwortlichkeit
Benutzer möchte seinen Standort auf dem Schlag erfassen.	Ermittelt den Standort des Benutzers.
Benutzer will seinen Standort auf der Anwendung sehen.	Präsentiert den Standort des Benutzers auf dem Schlag.

Tabelle 9: Essential Use Case 5: Berechnung der Düngemenge

Intention des Benutzers	Systemverantwortlichkeit
Benutzer möchte eine Düngemenge für seinen Schlag ermitteln.	Empfängt Bodendaten. Greift auf Pflanzen-Daten zu. Führt Berechnung für Düngemenge durch.

3.4 Concrete Use Cases

Mit den Concrete Use Cases sollen die Aufgaben der Benutzer weiter untersucht werden. Hierbei werden die Aufgaben mit Bezug konkreter Technologien beschrieben.

Tabelle 10: Concrete Use Case 1: Anlegen der Schlag-Daten

Aktion des Benutzers	System-Erwiderung
Benutzer möchte seine Schlag-Daten im System anlegen und tätigt den Hinzufügen-Button.	Das System präsentiert ihm die Funktion zum Anlegen von Schlag-Daten.
Benutzer trägt seine Schlag-Daten (Name, Größe, Umriss,...) im Formular ein.	Das System nimmt die eingetragenen Daten im Formular an und nimmt die Eingabe entgegen.
Benutzer bestätigt seine Eingabe durch das Betätigen des Speichern-Button.	Das System speichert die Schlag-Daten in der Datenbank ab.

Tabelle 11: Concrete Use Case 2: Bearbeiten der Schlag-Daten

Aktion des Benutzers	System-Erwiderung
Benutzer wählt einen eingetragenen Schlag aus und betätigt den Bearbeiten-Button.	Das System präsentiert ihm die eingetragenen Daten und ein Formular zum Ändern der Daten.
Benutzer bearbeitet die gewünschten Einträge.	Das System nimmt die eingetragenen Daten im Formular an und nimmt die Eingabe entgegen.
Benutzer bestätigt seine Eingabe durch das Betätigen des Speichern-Button.	Das System speichert die neuen Schlag-Daten in der Datenbank ab.

Tabelle 12: Concrete Use Case 3: Zuordnen von Pflanzen zu Schlägen

Aktion des Benutzers	System-Erwiderung
Benutzer wählt einen Schlag aus.	Das System präsentiert ihm die Daten des angeforderten Schlags.
Benutzer möchte Pflanze zum Schlag zuordnen.	Das System stellt ihm die Funktion zur Zuordnung der Pflanze zur Verfügung.
Benutzer wählt Pflanze aus, die zum Schlag zugeordnet werden soll und bestätigt Auswahl.	Das System nimmt Auswahl entgegen und speichert die Pflanze zu den zugeordneten Schlag-Daten in der Datenbank.

Tabelle 13: Concrete Use Case 4: Standort des Benutzers erfassen

Aktion des Benutzers	System-Erwiderung
Benutzer möchte seinen Standort auf dem Schlag erfassen.	Das System präsentiert ihm die Satellitenkarte.
Benutzer wählt den Schlag aus, auf dem er sich befindet.	Das System präsentiert ihm den angeforderten Schlag auf der Satellitenkarte.
Benutzer schaltet sein GPS vom Smartphone ein.	Mithilfe des GPS kann das System den Standort des Benutzers bestimmen und auf der Satellitenkarte anzeigen.

Tabelle 14: Concrete Use Case 5: Berechnung der Düngerempfehlung

Aktion des Benutzers	System-Erwiderung
Benutzer möchte eine Düngermenge für seinen Schlag berechnen.	Basierend auf den Bodendaten der Bodensensoren und den gespeicherten Pflanzenwerten berechnet das System eine Düngerempfehlung. Die Düngerempfehlung enthält Nährstoffwerte, die der Benutzer dem Boden hinzufügen soll.

3.5 Fazit Benutzungsmodellierung

Durch die Untersuchung und Analyse der Aufgaben der Landwirte wurden Erkenntnisse zu den Arbeitsprozessen und Zielen der Landwirte gewonnen. Im Bereich der Landwirtschaft mit Fokus auf den Ackerbau sind viele Aufgaben vorhanden, die von den Landwirten erledigen werden müssen. Viele dieser Aufgaben lassen sich vereinfachen und bei der Erledigung durch Systeme beschleunigen. Im nächsten Schritt wird es sein, mithilfe der Benutzermodellierung und der Benutzungsmodellierung Anforderungen an das System abzuleiten und zu formulieren.

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. Die Formulierung der Anforderungen wurde mithilfe der Literatur von Chris Rupp und Rainer Joppich [5] gestaltet.

4.1 funktionale Anforderungen

- F10: Das System muss dem Landwirten die Möglichkeit bieten, im System mit seinen Feldern zu arbeiten.
 - F11: Das System muss dem Landwirten die Möglichkeit bieten, seine Felder anzulegen.
 - F12: Falls ein Feld des Landwirten im System bereits vorhanden ist, muss das System dem Landwirten die Möglichkeit bieten, seine Feld-Daten zu bearbeiten.
 - F13: Falls ein Feld des Landwirten im System bereits vorhanden ist, muss das System dem Landwirten die Möglichkeit bieten, seine Feld-Daten zu löschen.
 - F14: Das System muss dem Landwirten die Möglichkeit bieten, Pflanzenkulturen zu seinen Feldern zuzuordnen.
 - F15: Das System muss fähig sein, die Daten der Felder (Umriss, Größe, Bezeichnung...) im System speichern.
 - F16: Das System muss fähig sein, dem Landwirten eine Liste der eingetragenen Schläge anzuzeigen.
- F20: Das System muss fähig sein, Bodendaten von Bodensensoren zu empfangen.
- F30: Das System muss fähig sein, die empfangenen Bodendaten automatisch zu verarbeiten.
- F40: Das System muss dem Landwirten, eine Düngerempfehlung basierend auf den Bodendaten und den angebauten Pflanzenkulturen automatisch berechnen.
- F50: Das System soll dem Landwirten, die Düngerempfehlung in graphischer und numerischer Darstellung präsentieren.
- F60: Das System muss fähig sein, den Landwirten zu benachrichtigen, falls eine neue Düngung benötigt wird.

- F61: Das System soll fähig sein, Berechnungen durchzuführen und aufgrund dieser erkennen zu können, ob an einem bestimmten Standort eine neue Düngung erforderlich wäre.
- F70: Das System soll fähig sein, basierend auf den berechneten Düngeempfehlungen auf einem Schlag Teilflächen zu bilden.
- F80: Das System soll fähig sein, den Landwirten bei der Ausbringung des Düngers zu unterstützen.
 - F81: Das System muss dem Landwirten, die Möglichkeit bieten, seinen Standort zu erfassen.
 - F82: Das System soll fähig sein, den Landwirten die Düngeempfehlung an seinem Standort auf dem Schlag zu präsentieren.

4.2 qualitative Anforderungen

- Das System soll eine hohe Gebrauchstauglichkeit liefern.
- Das System muss ort- 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.
- Die Bodensensoren müssen funktionsfähig und vollständig korrekte Daten messen und liefern.
- Die Bodensensoren sollen im Intervall von einem Tag den Boden untersuchen.

5 UI Prototyp

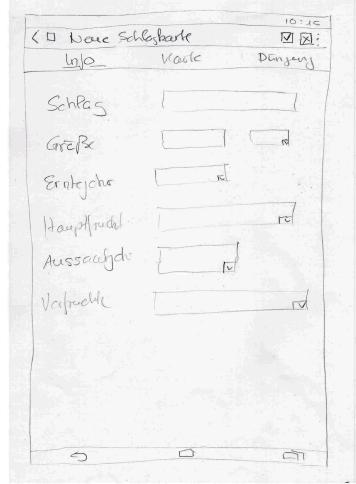
Vor der UI-Evaluation muss anhand der aufgestellten Anforderungen ein User Interface entwickelt und gestaltet werden. Im Konzept wurde bereits erläutert, dass die Erstellung des UI Prototyps in Form von paperbased Prototyp erfolgen soll. Im folgenden Abschnitt sollen wichtige Merkmale des ersten UI Prototyp erläutert und Gestaltungsentscheidungen begründet werden.

Als Erstes lässt sich sagen, dass bei der Gestaltung des User Interfaces sich hauptsächlich an den Android Gestaltungsregeln[[1]] gehalten wurde. Wichtige UI-Elemente lehnen sich daher an das Android Styleguide an.

Schlagkartei

Anhand der Nutzungsanforderungen stellt sich raus, dass der Landwirt mit seinen Schlag-Daten arbeiten muss und dass diese vom System erfasst werden müssen. Genauso ist es wichtig für ihn, den Umriss seines Schlags markieren bzw. definieren zu können.

In der Abbildung 4a erkennt man den Screen zum Hinzufügen eines neuen Schlags. Dort können vom Landwirten Angaben zu definierten Merkmalen des Schlags gemacht werden. Mit dem Betätigen des Haken-Buttons kann er die Eingaben speichern. Um in den nächsten Screen zum Definieren des Schlag-Umrisses (Abbildung 4b) zu gelangen, muss der Nutzer eine Wisch-Geste nach links ausführen. Dadurch ist es ihm möglich, vom Zustand "Info" in den Zustand "Karte" zu gelangen, die sich in der Navigations-Bar unterhalb der Action-Bar befinden. Dort kann er durch Touch-Gesten, Markierungen auf der Karte setzen. Die Markierung sollen letztendlich eine Fläche bilden, die den Schlag des Nutzers darstellen soll. Auf der Abbildung 5 ist die Liste der eingetragenen Schläge zu erkennen. Dort sind die Schläge alphabetisch sortiert. Durch das Drücken des Plus-Button am unteren rechten Rand würde man in den Schlag hinzufügen-Screen gelangen.



(a) Screen: Schlag hinzufügen



(b) Screen: Schlagumriss definieren

Abbildung 4: Screens zur Schlagkartei

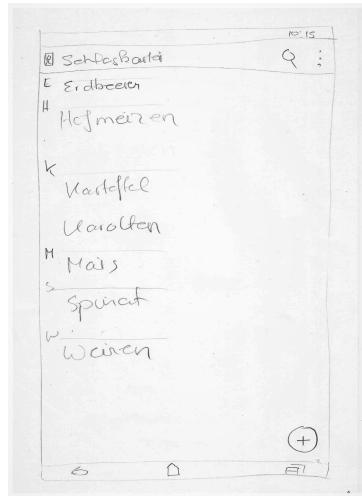


Abbildung 5: Screen: Liste der eingetragenen Schläge

Düngeempfehlung

Auf Grundlage der Anforderung F50 wurden die Screens in Abbildung 6 und 7 entwickelt und gestaltet. Das System führt die Berechnung der Düngeempfehlung durch, ohne dass der Benutzer diese Berechnung sehen kann. Deshalb müssen die Ergebnisse dieser Berechnungen, dem Benutzer präsentiert werden. Um zu den Ergebnissen der Düngereberechnung zu gelangen, muss der Benutzer nach Auswahl eines Schlags zweimal nach links wischen, um den Zustand "Düngung" in der Navigations-Bar zu erreichen. Dort wird die Fläche des zuvor definierten Schlags dargestellt sowie eine farbliche Einteilung der Fläche in Teilflächen basierend auf den Düngereberechnungen.

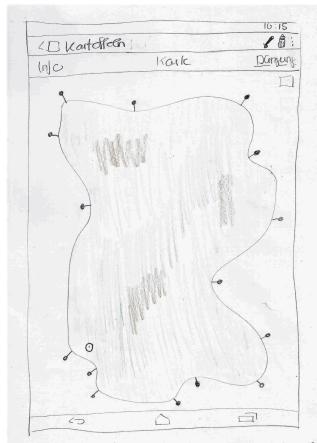


Abbildung 6: Screen: Düngeempfehlung in graphischer Darstellung

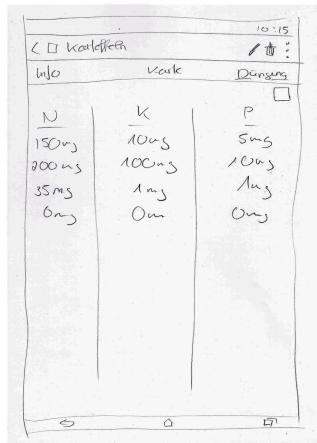


Abbildung 7: Screen: Düngeempfehlung in numerischer Darstellung

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.[[6]] 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

Durch die heuristische Evaluation mit einem Evaluator, der über moderate MCI-Kenntnisse verfügt und bereit war eine Evaluation durchzuführen, wurden Probleme des ersten User Interfaces aufgedeckt. Die gewonnenen Erkenntnisse und die Veränderungen, die auf Grundlage der Erkenntnisse entstanden sind, sollen im Folgenden beschrieben werden.

Allgemein

Hier sollen die Erkenntnisse, die sich auf die allgemeine Gestaltung des User Interfaces bezieht, beschrieben werden. Fast in jedem Screen wurde in der Action Bar das Overflow Icon dargestellt. Jedoch hatte sie nicht für jeden Screen einen Nutzen. Der Evaluator hatte sich immer gefragt, ob nach dem Tätigen des Icons weitere Einstellungen erscheinen. Deshalb wurden bzgl. des Overflow

Icons Veränderungen vorgenommen. Der Overflow Icon wurde aus den Screens entfernt, in denen er überflüssig war.

Schlagkartei

Das User Interface bzgl. der Schlagkartei wurde aufgrund der Evaluation überarbeitet. In dem alten Schlag hinzufügen-Screen waren die Eingabefelder, mit dem der Nutzer interagieren konnte, zu klein gestaltet und es wurden Merkmale für den Schlag definiert, die für die Aufgabenerledigung jedoch nicht von Nutzen sind. In Bezug auf die Heuristik "aesthetic and minimalist design" wurden die überflüssigen Merkmale gestrichen. Außerdem wurden die Eingabefelder überarbeitet, so dass sie deutlich als Eingabefelder gesehen.

Des Weiteren wurde in den Schlag hinzufügen-Screens angemerkt, dass der Haken-Button, der Abbrechen-Button und das Overflow Icon direkt nebeneinander liegen und dadurch für Verwirrung sorgen. Für die neue Version befindet sich auf der rechten Seite der Action Bar nur noch ein Haken zum Speichern der Daten. Das Abbrechen soll durch den Pfeil-Zurück Button auf der linken Seite der Action Bar erfolgen.

Außerdem bestand ein Problem bei der Definition des Schlag-Umrisses. Es wurde die Frage gestellt, wie man auf der Karte sein Feld finden sollte. Auf der UI war es lediglich nur möglich, durch Zoomen der Karte auf ein Ziel zu kommen. Jedoch wurde erkannt, dass dies für die Benutzer zu umständlich wäre, weshalb für die neue UI

Düngeempfehlung

Die Screens der Düngeempfehlung war für den Evaluator sehr verwirrend, da er kein Domänenexperte ist. Die Einteilung der Fläche in kleinere Teilflächen und die Darstellung des GPS-Standortes auf der Karte wurde erkannt. Jedoch war der Button zum Wechseln in die numerische Ansicht gar nicht erkennbar, so dass der Button gar nicht aufgefallen war. Es wurde eine Veränderung vorgenommen, in der beide Ansichten in einem Screen vereint werden, damit der Nutzer nicht ständig zwischen beiden Darstellungen wechselt.

7 Systemarchitektur

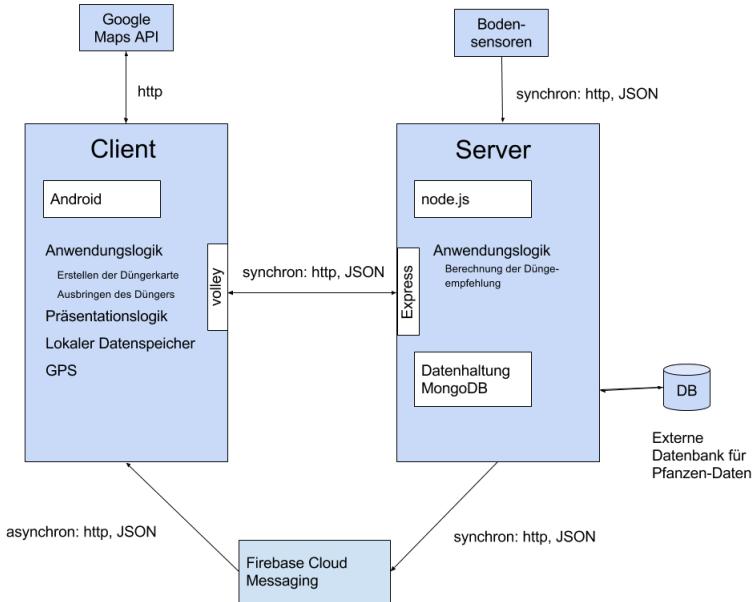


Abbildung 8: neues Architekturdiagramm

Die Abbildung 8 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.

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 8 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

Im Vergleich zur Architektur, die im Konzept erstellt wurde, wurde die neue Systemarchitektur durch den Firebase Cloud Messaging[2] Service ergänzt. Der Service sorgt für die Realisierung der asynchrone 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 uns 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 waren 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 getätigigt 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 Düngempfehlung 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 Umrissse 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_id: 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      field_id: number,
3      nutrient: {
4          nitrogen: number,
5          phosphorus: number,
6          potassium: number
7      },
8      location: {
9          latitude: number,
10         longitude: number
11     }
12 }
```

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      field_id: number,
3      nutrient: {
4          nitrogen: number,
5          phosphorus: number,
6          potassium: number
7      },
8      location: {
9          latitude: number,
10         longitude: number
11     }
12 }
```

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 Düngerempfehlung 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 Düngerempfehlung 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.

Tabelle 15: REST-Spezifikation der verwendeten Ressourcen

Ressource	Methode	Semantik	Content-Type (Request)	Content-Type (Response)
/field	GET	Anzeige aller eingetragenen Felder/Schläge	-	application/json
/field?search=query	GET	Anzeige aller eingetragenen Felder/Schläge, die zum Suchwort query gefunden wurden	-	application/json
/field	POST	Hinzufügen eines neuen Feldes/Schlags	application/json	plain
/field/:id	GET	Ausgabe eines bestimmten Feldes/Schlags anhand der Feld-ID	-	application/json
/field/:id	PUT	Aktualisierung eines bestimmten Feldes/Schlags anhand der Feld-ID	application/json	plain
/field/:id	DELETE	Löschen eines bestimmten Feldes/Schlags anhand der Feld-ID	-	plain
/plant	GET	Anzeige aller gespeicherten Pflanzen	-	application/json
/plant/:id	GET	Ausgabe einer bestimmten Pflanze anhand der Pflanzen-ID	-	application/json
/fertilizer/:fid	GET	Holen aller Düngerempfehlungen auf einem Feld/Schlag anhand der Feld-ID	-	application/json
/fertilizer/:fid/:id	GET	Holen einer bestimmten Düngerempfehlung auf einem Feld/Schlag anhand der Feld-ID und der ID der Düngerempfehlung	-	application/json

10.2 Anwendungslogik

10.2.1 Berechnung einer Düngerempfehlung

Für die Berechnung der Düngerempfehlung 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 Düngerempfehlung 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, basierend auf der Eingabe des Landwirten welche Pflanzenkultur er für die Berechnung der Düngerempfehlung nehmen muss.
4. Der Server berechnet nun basierend auf den Nährstoff-Sollwerten der zu düngenden Pflanzenkultur und auf den ermittelten Bodenwerten eine Düngerempfehlung für einen bestimmten Standort auf dem Schlag.
5. Diese Berechnung erfolgt nun für alle Standorte auf dem Schlag, die von den Bodensensoren gemessen wurden sind.

```
1   FUER alle empfangenen Bodendaten eines Schlags
2       erstelle neues Objekt fuer Duengeempfehlung
3       weise dem neuen Objekt die field_id und location zu
4       WENN plant.nitrogen - boden.nitrogen > 0
5           DANN Objekt.nitrogen = plant.nitrogen - boden.
6               nitrogen
7           SONST Objekt.nitrogen = 0
8       WENN plant.phosphorus - boden.phosphorus > 0
9           DANN Objekt.phosphorus = plant.phosphorus -
10              boden.phosphorus
11           SONST Objekt.phosphorus = 0
12       WENN plant.potassium - boden.potassium > 0
13           DANN Objekt.potassium = plant.potassium - boden.
14               potassium
15           SONST Objekt.potassium = 0
16
17   ENDE
```

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 und der berechneten Werte in der Düngeempfehlung Teilflächen zu erstellen. Diese Teilflächen sollen die unterschiedlichen Düngeempfehlungen auf einem Schlag darstellen.

Bemerkung: Aufgrund mangelnder Zeit konnte diese Anwendungslogik innerhalb des zweiten Meilensteins noch nicht komplett modelliert werden. Sie soll in der Implementierungsphase fertiggestellt werden. Der Ansatz könnte sein, durch eine Interpolation der berechneten Düngeempfehlungen auf einem Schlag eine Düngekarte zu erstellen und basierend auf den Berechnungen Teilflächen zu bilden.

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. Mithilfe des GPS vom mobilen Endgerät ist es möglich, den Standort des Landwirten während der Ausbringung des Düngers zu erfassen. Dadurch kann der Landwirt, sich auf der Düngekarte verfolgen und in Echtzeit schauen, wo er wie viel Dünger ausbringen muss. Betritt er bspw. eine neue Teilfläche auf dem Schlag, in der eine andere Düngeempfehlung vorherrscht, soll es ihm die neue Empfehlung anzeigen.

```
1      WENN GPS innerhalb der Teilflaeche
2          DANN praesentiere die Duengeempfehlung dieser
3              Teilflaeche
4      WENN GPS von dieser Teilflaeche in eine andere
5          Teilflaeche wechselt
6          DANN greife auf die Duengeempfehlung der neuen
7              Teilflaeche zu und praesentiere sie
```

10.2.4 Benachrichtigung, wenn gedüngt werden sollte

Durch die Nährstoffaufnahme der Pflanzen sinkt mit der Zeit wieder der Nährstoffgehalt im Boden. Die Bodensensoren messen den Nährstoffgehalt im Intervall von einem Tag. Befindet sich die Differenz des Nährstoff-Sollwerts und des Nährstoffgehalts im Boden über einen bestimmten Schwellwert, soll vom Server an den Client eine Benachrichtigung gesendet werden, die besagt, dass neu gedüngt werden sollte und wie viel an Nährstoffen gedüngt werden sollte.

```
1     Schwellwert = 15
2     FUER alle empfangenen Bodendaten eines Landwirten
3         WENN plant.nitrogen - boden.nitrogen > Schwellwert
4             UND plant.phosphorus - boden.phosphorus >
5                 Schwellwert UND plant.potassium - boden.potassium
6                     > Schwellwert
7             DANN sende Nachricht mit Feldname, Standort und
8                 Duengeempfehlung
9
10    ENDE
```

Abbildungsverzeichnis

1	Nikolas Stein	10
2	Emily Solay	11
3	Thomas Blaumann	12
4	Screens zur Schlagkartei	24
5	Screen: Liste der eingetragenen Schläge	24
6	Screen: Düngempfehlung in graphischer Darstellung	25
7	Screen: Düngempfehlung in numerischer Darstellung	25
8	neues Architekturdiagramm	29
9	Screen: Darstellung der Schlag-Informationen	47
10	Screen: Suche nach einem bestimmten Schlag	47
11	Screen: Bearbeiten eines Schlags	48
12	Screen: Darstellung der definierten Fläche des Schlags	48
13	Screen: Bearbeiten der definierten Fläche des Schlags	49
14	neuer Screen: Liste der eingetragenen Schläge	49
15	neuer Screen: Suche nach einem bestimmten Schlag	50
16	neuer Screen: Liste der eingetragenen Schläge	50
17	neuer Screen: neuen Schlag anlegen	51
18	neuer Screen: neuen Schlag-Umriss definieren	51
19	neuer Screen: Suche auf der Karte nach Feld	52
20	neuer Screen: Schlag-Daten bearbeiten	52
21	neuer Screen: Schlag-Umriss bearbeiten	53
22	neuer Screen: graphische und numerische Darstellung der Düngempfehlung	53

Tabellenverzeichnis

1	Stakeholder-Analyse	5
2	User Profile 1: erfahrener Landwirt	7
3	User Profile 2: landwirtschaftlicher Mitarbeiter	8
4	User Profile 3: angehender Landwirt	9
5	Essential Use Case 1: Anlegen von Schlag-Daten	18
6	Essential Use Case 2: Bearbeiten der Schlag-Daten	18
7	Essential Use Case 3: Zuordnen von Pflanzen zu Schlägen	18
8	Essential Use Case 4: Standort des Benutzers erfassen	18
9	Essential Use Case 5: Berechnung der Düngemenge	18
10	Concrete Use Case 1: Anlegen der Schlag-Daten	19
11	Concrete Use Case 2: Bearbeiten der Schlag-Daten	19
12	Concrete Use Case 3: Zuordnen von Pflanzen zu Schlägen	19
13	Concrete Use Case 4: Standort des Benutzers erfassen	20
14	Concrete Use Case 5: Berechnung der Düngempfehlung	20
15	REST-Spezifikation der verwendeten Ressourcen	39

Glossar

- Schlag** Der Begriff Schlag bezeichnet eine Fläche auf einem Feld, das vom Landwirten selber eingeteilt wurde. Oft wächst auf einem Schlag nur eine Pflanzenkultur.
- Ausbringung** Unter Ausbringung von Dünger ist der Vorgang gemeint, bei dem der Landwirt seine Felder düngt. Oft geschieht dies mithilfe von Maschinen.
- Nährstoff-Sollwerte** Unter Nährstoff-Sollwerte sind die Nährstoffwerte gemeint, die im Boden für die Pflanzen verfügbar sein sollte, damit sie optimal wachsen können.

Literatur

- [1] Android: User interface. <https://developer.android.com/guide/topics/ui/index.html>. abgerufen am: 04.12.2016.
- [2] Firebase cloud messaging. <https://firebase.google.com/>. abgerufen am: 03.12.2016.
- [3] Zeit: Der preis der billigen milch. <http://www.zeit.de/wirtschaft/unternehmen/2016-05/landwirtschaft-milchpreis-niedrig-wenig-kuehe-milchbauern-investition>. abgerufen am: 29.11.2016.
- [4] DIN EN ISO 9241 Teil 210 Prozess zur Gestaltung gebrauchstauglicher interaktiver Systeme, 2008.
- [5] Rainer Rupp, Chris und Joppich. *Requirements-Engineering und Management*. 6. Auflage Carl Hanser Verlag, München, 2014.
- [6] Werner Schweibenz and Frank Thissen. *Qualität im Web: Benutzerfreundliche Webseiten durch Usability-Evaluation*. Springer, Berlin, 1 edition, 2002.

ANHANG

A Erstes UI-Prototyp

Hier folgen die restlichen Screens, die für das UI Prototyp erstellt worden sind, aber in der Dokumentation nicht genannt werden:

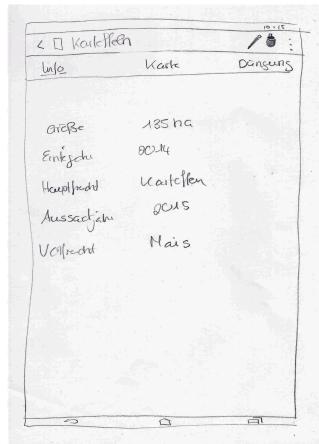


Abbildung 9: Screen: Darstellung der Schlag-Informationen



Abbildung 10: Screen: Suche nach einem bestimmten Schlag

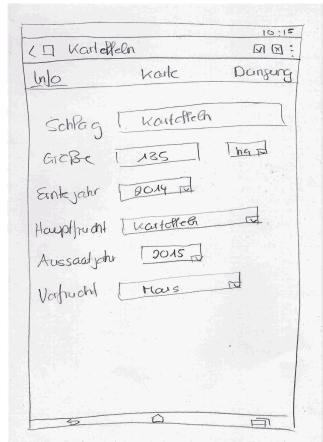


Abbildung 11: Screen: Bearbeiten eines Schlags

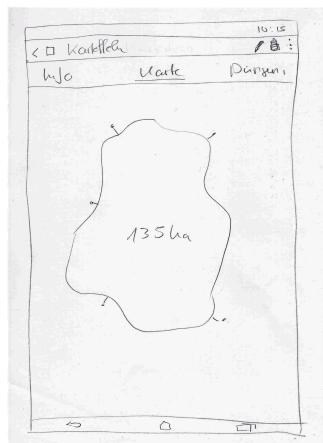


Abbildung 12: Screen: Darstellung der definierten Fläche des Schlags

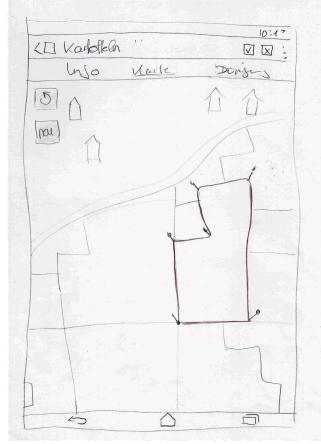


Abbildung 13: Screen: Bearbeiten der definierten Fläche des Schlags

B Finales UI-Prototyp

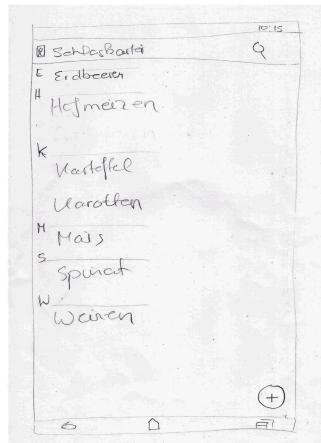


Abbildung 14: neuer Screen: Liste der eingetragenen Schläge



Abbildung 15: neuer Screen: Suche nach einem bestimmten Schlag

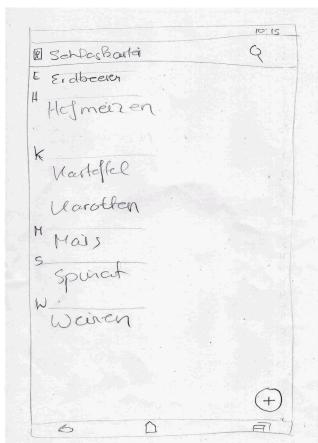


Abbildung 16: neuer Screen: Liste der eingetragenen Schläge

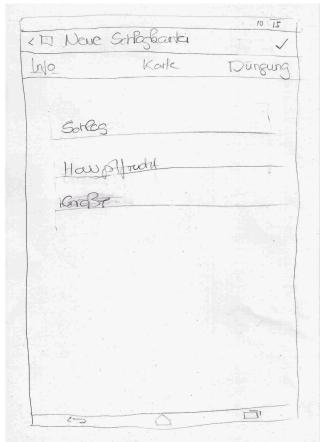


Abbildung 17: neuer Screen: neuen Schlag anlegen

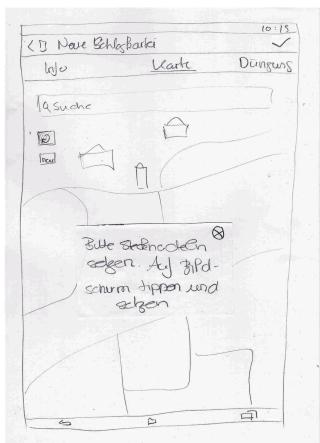


Abbildung 18: neuer Screen: neuen Schlag-Umriss definieren

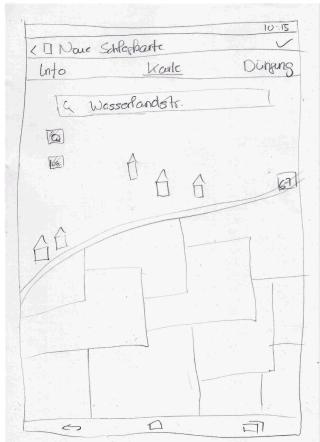


Abbildung 19: neuer Screen: Suche auf der Karte nach Feld

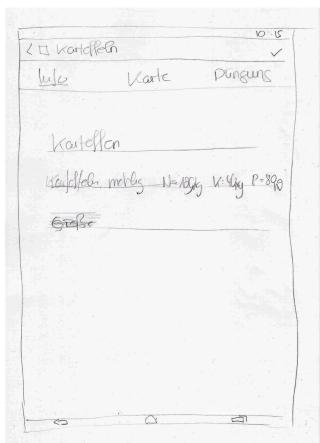


Abbildung 20: neuer Screen: Schlag-Daten bearbeiten

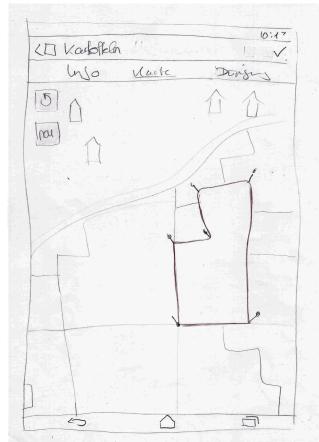


Abbildung 21: neuer Screen: Schlag-Umriss bearbeiten

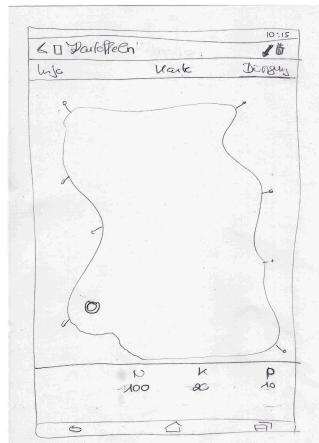


Abbildung 22: neuer Screen: graphische und numerische Darstellung der Düngerempfehlung

C Evaluations-Protokoll

Hier befindet sich das Evaluations-Protokoll, das im Rahmen der heuristischen Evaluation mit einem Evaluator entstanden ist. Das Protokoll ist eine Ansammlung, Zusammenfassung und Auswertung von Notizen, die während der Evaluation entstanden sind.

Evaluator: Vu Phi Hai Dinh Datum: 24.11.2016			
Beschreibung des Problems	Interaktionssituation/Stelle des Problems	Verletzte Heuristik	Schweregrad
Beim Markieren des Feldes auf der Karte. Der Evaluator wusste nicht, wie er nach seinem Feld suchen kann bzw. die Karte hat ihm sein Standort angezeigt. Aber er war nicht in der Nähe seines Feldes.	Er hat die Karte angeschaut und was gesucht. Er hat geschaut wo er ein tippen kann wo seine Felder sind. Er hat zwei Actions Button gesehen und versuch es anzuklicken. Nichts passiert.	Visibility of System status Flexibility and efficiency of use	4
Bei der Bearbeitung der Informationen zur der Schlagkartei. War dem Evaluator die Box fürs eintragen zu klein auf dem Smartphone.	Er versuchte auf das Wort Hauptfrucht zu klicken. Merkte das nichts passiert und hat auf das Feld daneben geklickt.	Recognition rather than recall	2
Als er sich die Graphische Darstellung angeschaut hat, war ihm die Farben nicht geläufig. Auch hat er erst später bemerkt, dass man zwischen Zahlen und Graphen wechseln kann. Ihm haben die Zahlen auch nicht viel gesagt. Ihm hat eine Agenda gefehlt. Das Wechseln zwischen zwei verschiedene Screen war ihm zur Unübersichtlich.	Das Wechseln zwischen dem Graphischen Darstellung und Zahlendarstellungen. Er hat sich die Grafik angeschaut und dann zur Zahlendarstellung. Hat versuch die Zahlen der Grafischen Darstellung zuordnen und die grafisch farblich markierte Karte zu verstehen.	Visibility of System Status Recognition rather than recall	4
Bei der Setzung der Pinnadeln, war das Symbol für den letzten Pinnadeln entfernen nicht ersichtlich. Er hat erwartet, dass es sich aktualisiert und die Flächen Größe angezeigt wird.	Der Nutzer hat seine Pinnadeln gesetzt und dann auf das Symbol mit dem Pfeil im Kreis. Er war erstaunt, dass die letzte Pinnadel entfernt würde.	Recognition rather than recall	2

Das action bar over flow Icon war in jedem Screen vorhanden, hatte aber nicht überall einen nutzen.	Der Evaluator klickt auf die drei Punkte oben rechts und erwartet das was passiert. Nichts passiert und der Evaluator fragt was das ist.	Recognition rather than recall Consistency and standards	3
---	--	---	---

Agenda

Schweregrad	Beschreibung
1	Kosmetisches Problem (nur beseitigen, wenn genügend Zeit ist)
2	Kleines Problem (geringe Priorität bei der Beseitigung)
3	Großes Problem (hohe Priorität bei der Beseitigung)
4	Usability-Katastrophe (muss unbedingt beseitigt werden)

D Konzept

Technology Arts Sciences

TH Köln

TH Köln Campus Gummersbach
Fakultät für Informatik und
Ingenieurwissenschaften

ENTWICKLUNGSPROJEKT INTERAKTIVE SYSTEME

Konzept

Thuy Trang Nguyen
Duc Giang Le

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

7. November 2016

Inhaltsverzeichnis

1 Zielhierarchie	3
2 Domänenrecherche	4
2.1 Beschreibung der Domäne	4
2.2 Identifizierung der Stakeholder	5
3 Marktrecherche	6
3.1 Systeme zur teilflächenspezifischen Bewirtschaftung	6
3.2 Dünger-Applikationen	7
4 Alleinstellungsmerkmale	7
4.1 Düngeempfehlung nach der teilflächenspezifischen Bewirtschaftung	7
4.2 Kein hoher Kostenaufwand	7
5 Methodischer Rahmen	8
5.1 User-centered Design oder Usage-centered Design	8
5.2 MCI-Vorgehensmodelle	8
5.3 Auswahl der MCI-Methoden	9
6 Kommunikationsmodell	10
6.1 deskriptives Kommunikationsmodell	10
6.2 präskriptives Kommunikationsmodell	11
7 Architekturdiagramm	12
7.1 Anwendungsart	13
7.2 Server	13
7.3 Datenformat	13
7.4 Externe Schnittstelle	14
7.5 Externe Datenquellen	14
8 Risiken	14
8.1 Datenschutz (GPS)	14
8.2 Nutzung von externen Schnittstellen	14
8.3 Ungenaue Standortbestimmung durch GPS	15
8.4 Anschaffung von Bodenanalysesensoren	15
9 Spezifikation der Proof-of-Concepts	16
9.1 Einbindung einer externen Schnittstelle	16
9.2 Ermittlung des Standortes durch GPS in Google Maps	16
9.3 Kommunikation zwischen Systemkomponenten	17
9.4 Erstellen von Polygonen auf den Satellitenbildern	17
10 Rapid Prototyping	18
11 Literaturverzeichnis	21

Abbildungsverzeichnis

1	deskriptives Kommunikationsmodell	10
2	präskriptives Kommunikationsmodell	11
3	Architekturdiagramm	12
4	Main-Activity des Android Clients	18
5	Post-Activity des Android Client	19
6	Map-Activity des Android Client	20

1 Zielhierarchie

Die Zielhierarchie dient zur Definition der strategischen (langfristigen), taktischen (mittelfristigen) und kurzfristigen (operativen) Ziele, die unser Projekt bzw. das zu entwickelnde System verfolgt. Die Zielpriorisierung erfolgt durch die verwendeten Verben (muss, soll, kann) und soll zeigen, welche Ziele welche Wichtigkeit für unser Projekt haben.

Strategische Ziele:

- Eine teilflächenspezifische Düngung muss für den Landwirt aus kleineren bis mittleren Betrieben erleichtert werden.
- Es soll für eine bedarfsgerechte Düngung gesorgt werden, so dass Ressourcen gespart werden und die Umwelt weiter geschont wird.

Taktische Ziele:

- Das System soll dem Landwirt alle für die Düngung wichtigen Informationen zusammenstellen.
- Das System muss dem Landwirt bei der teilflächenspezifischen Düngung unterstützen.
- Wichtige Informationen, wie Bodenzustand oder Nährstoffbedarf von Pflanzen, sollen erfasst werden können.

Operative Ziele:

- Der Nutzungskontext soll detailliert untersucht und analysiert werden, damit die Anforderungen der Nutzer identifiziert werden können.
- Es sollen Risiken identifiziert werden, die den Projekterfolg und das System gefährden können. Mögliche Gegenmaßnahmen sollen überlegt werden.
- Es muss ein geeignetes Vorgehensmodell der MCI abgewägt und für den weiteren Prozess ausgewählt werden.
- Es sollen technische Anforderungen in Form von einem detaillierten Architekturdiagramm festgelegt und erläutert werden.
- Es soll ein Prototyp entwickelt werden, der das zu entwickelnde System mit den Funktionalitäten repräsentieren soll.
- Es soll untersucht werden, wie Informationen wie Bodenzustand oder Nährstoffbedarf von Pflanzen vom System erfasst werden.

2 Domänenrecherche

2.1 Beschreibung der Domäne

Im Folgenden wird beschrieben, in welcher Domäne das zu entwickelnde System zum Einsatz kommen soll. Dabei sollen auf wichtige domänenspezifische Prozesse und Eigenschaften eingegangen werden.

Die Umgebung, in dem das zu entwickelnde System eingesetzt werden soll, befindet sich in der Agrarwirtschaft. Ziel der Agrarwirtschaft ist die Herstellung von pflanzlichen und tierischen Produkten. Die Beschäftigten der Agrarwirtschaft werden meist als Landwirte oder landwirtschaftliche Fachkräfte bezeichnet.

Für eine erfolgreichen Ernte müssen die Landwirte viele Faktoren beachten. Unter anderem spielen Umweltfaktoren, wie Feuchtigkeit bzw. Niederschläge, Temperatur oder Lichteinstrahlung eine wichtige Rolle. Aber auch Faktoren, wie Bodenzustand, vorhandene Bodennährstoffe oder Nährstoffbedarf von Pflanzen sind von großer Bedeutung. Wenn bspw. durch voriger Ernte nicht genügend Nährstoffe im Boden vorhanden sind und der Nährstoffbedarf der Pflanzen nicht gedeckt werden kann, müssen die Landwirte dem Boden eigenständig Nährstoffe geben. Dieser Prozess wird als Düngung bezeichnet.

Unter Dünger oder Düngemittel versteht man Stoffgemische, die in der Landwirtschaft dazu benutzt werden, die Nährstoffe im Boden zu ergänzen, so dass die angebauten Pflanzen für ihren Wachstum genügend Nährstoffe aufnehmen können. In der Agrarwirtschaft werden zwei Düngerarten als wichtig erachtet: organischer Dünger und mineralischer Dünger. Der organische Dünger besteht aus organischen Substanzen, die als Abfallstoffe in der Landwirtschaft entstehen. Tierische Abfallstoffe sind bspw. Gülle, Jauche und Mist, während Stroh oder Pflanzenrückstände zu den pflanzlichen Abfallstoffen gehören. Der mineralische Dünger hingegen beinhaltet die Nährstoffe in Salzform. Diese Salze lösen sich im Bodenwasser, so dass die Pflanze mit ihren Wurzeln die Nährstoffe aufnehmen kann. Dadurch, dass die mineralischen Dünger künstlich produziert werden, können unterschiedliche Dünger hergestellt werden, die verschiedene Konzentrationen von Nährstoffen aufweisen. Dabei lassen sie sich zwischen Einzel- und Mehrnährstoffdünger unterscheiden. Während der Einzelnährstoffdünger ausschließlich nur einen Nährstoff aufweist, können bei mineralischen Mehrnährstoffdüngern mehrere Nährstoffe, wie Stickstoff (N), Phosphor (P) oder Kalium (K) vorhanden sein.

Obwohl das Düngen einige Vorteile bietet, birgt sie auch negative Auswirkungen, die die Menschen, Tiere und Umwelt bei einer Überdüngung schädigen können. Eine Überdüngung führt zur Übersättigung des Bodens, was eine Verunreinigung des Grundwassers verursachen kann, weil die übriggebliebenen Stoffe durch Nichtaufnahme der Pflanzen ins Grundwasser geraten können. Deshalb sind Landwirte dazu verpflichtet zu kontrollieren, dass auf ihren Feldern keine Überdüngung vorliegt. Außerdem soll durch die Düngeverordnung, die am 7. Februar 1996 in Kraft getreten ist, die Anwendung von Düngemittel nach den Grundsätzen der guten fachlichen Praxis beim Düngen geschehen. Ziel der Düngeverordnung ist es durch einen schonenden Einsatz von Düngemitteln und

eine Verminderung von Nährstoffverlusten langfristig die Nährstoffeinträge in die Gewässer und andere Ökosysteme zu verringern.

Bei vielen heutigen kleineren landwirtschaftlichen Betrieben verläuft die Düngerverteilung meistens ohne genauere Betrachtung des heterogenen Boden. Die verschiedenen Bodennährstoffe, die innerhalb eines Feldes unterschiedlich hoch vorhanden sind, werden nicht exakt untersucht und in die Düngerdosierung miteinbezogen. Das hat zur Folge, dass an manchen Stellen des Feldes zu viele Nährstoffe hinzugefügt werden und an anderen Stellen wiederum weniger. Dies ist auch einer der Gründe, weshalb in der Landwirtschaft Überdüngung auftreten kann. Um diesem Problem entgegenzuwirken, wurde ein landwirtschaftliches Verfahren entwickelt, dass den heterogenen Boden genauer betrachtet.

Bei der Teilflächenbewirtschaftung (auch genannt als precision farming) handelt es sich um eine landwirtschaftliche Bearbeitungstechnik, bei der mithilfe von GPS, Karten und Ackerschlagdateien eine teilflächenspezifische Aussaat sowie bedarfsorientierte Pestizidanwendung und Düngung punktgenau ermöglicht ist. Das Ziel dieses Verfahrens ist es, die Unterschiede des Bodens und der Ertragsfähigkeit innerhalb eines Feldes zu berücksichtigen. Für das Projekt soll der Fokus jedoch nur auf die teilflächenspezifische Düngung gelegt werden. Sie ist ein wichtiger Bestandteil der Teilflächenbewirtschaftung. Es werden gezielt Informationen über den Boden und den Nährstoffbedarf von Pflanzen gesammelt, damit eine bedarfsgerechte Düngung erfolgen kann.

2.2 Identifizierung der Stakeholder

Landwirte von kleineren bis mittleren Betrieben

Landwirte sind als Menschen zu beschreiben, die in der Landwirtschaft beschäftigt sind. Sie nutzen Nutzflächen, um tierische und pflanzliche Produkte zu erzeugen, die meist als Nahrungsmittel verwendet werden. Für das Projekt haben Landwirte in kleineren bis mittleren Betrieben eine hohe Priorität, weil sie sich keine großen Investitionsmöglichkeiten wie bei Großbetrieben leisten können und sie die Haupt-Zielgruppe des Projekts sind. Sie sollen direkt mit dem System arbeiten, das ihnen bei ihren Arbeitsprozessen unterstützt.

Familie der Landwirte

Die Familien der Landwirte profitieren indirekt durch das System, denn langfristig gesehen würden durch die gezielte Düngung, Ressourcen gespart und der Ertrag gefördert werden, was eine Erhöhung des Gewinns nach sich zieht.

Lohnunternehmer

Lohnunternehmer sind Menschen, die gegen Rechnung eine gewerbliche Dienstleistung für ein anderes Unternehmen ausführen. Meistens beschränken sich ihre Dienstleistungen auf verschiedenste Arbeiten in der Landwirtschaft (zum Beispiel Ernte, Düngung und Bodenbearbeitung). Sie haben wie die Landwirte denselben Anspruch am System, da es ihnen ebenfalls bei ihren Arbeitsprozessen unterstützen soll.

Staat/Land

Der Staat bzw. das Land würden indirekt vom System profitieren, weil durch die Nutzung des Systems Dünger eingespart werden und eine Überdüngung verhindert werden soll, so dass die Umwelt geschont wird.

3 Marktstudie

Eine Marktstudie ist zwingend notwendig, um mögliche Konkurrenzprodukte zu identifizieren, deren Funktionalitäten bzw. Teifunktionalitäten Lösungsansätze zum Nutzungsproblem zur Verfügung stellen. Durch das Identifizieren von Vor- und Nachteilen der Konkurrenzprodukte und durch den Vergleich mit dem eigenen Projekt lässt sich eine Abgrenzung zur Konkurrenz durchführen.

3.1 Systeme zur teilflächenspezifischen Bewirtschaftung

Zurzeit befinden sich einige Systeme auf dem Markt, die sich auf die Bestandteile der teilflächenspezifischen Bewirtschaftung in der Landwirtschaft konzentrieren und dadurch dem Landwirten die Ausführung des precision farming ermöglicht. In unserem Projekt ist die teilflächenspezifische Düngung von großer Bedeutung, weshalb diese Systeme auf diesem Aspekt untersucht werden sollen.

Vorteile dieser Systeme sind, dass sie sehr präzise arbeiten können. Mithilfe von unterschiedlichen Sensoren auf dem Traktor können sie bspw. den Stickstoffgehalt der Pflanzen während der Fahrt auf dem Acker messen, indem sie die Lichtreflexion der Pflanzen erkennen und analysieren. Dadurch kann der Stickstoffbedarf der Pflanzen direkt ermittelt und gedüngt werden. Jedoch beschränkt sich diese Art der Düngung nur auf die N-Düngung. Außerdem benötigen die Landwirte für die Anschaffung solcher Systeme viel Kapital für derartige Investitionen, weshalb nicht jeder Landwirt solche Systeme besitzen können.

3.2 Dünger-Applikationen

Auf dem Markt befinden sich nach aktuellem Stand keine Applikationen, die dem Nutzer anhand von bestimmten Daten eine Düngeempfehlung selbstständig berechnet und vorschlägt. Es gibt Applikationen, mit dem die Nutzer die Ausbringmenge des Düngers berechnen lassen können, indem sie die benötigten Daten selber in die Anwendung eintragen. Jedoch betrachten diese Dünger-Applikationen nicht die heterogenen Bodenzustände, wodurch eine bedarfsgerechte Düngung mithilfe diesen Applikationen nicht möglich wäre.

4 Alleinstellungsmerkmale

Auf Grundlage der Marktrecherche lassen sich für das Projekt Alleinstellungsmerkmale herauskristallisieren, die im folgenden Abschnitt erläutert werden sollen.

4.1 Düngeempfehlung nach der teilflächenspezifischen Bewirtschaftung

Wie in der Marktrecherche erkannt wurde, befinden sich keine Applikationen bzw. Anwendungen, die die Berechnung der Düngeempfehlung nach dem Prinzip der teilflächenspezifischen Bewirtschaftung durchführt. Für die Berechnung der Düngeempfehlung werden zum einen der Nährstoffbedarf der angebauten Pflanzen und zum anderen die Nährstoffverteilung des Bodens vom Acker des Landwirten benötigt, damit eine bedarfsgerechte Düngung ermittelt werden kann.

4.2 Kein hoher Kostenaufwand

Durch die Untersuchung der vorhandenen Systeme für das Ausführen der teilflächenspezifischen Bewirtschaftung wurde festgestellt, dass die Anschaffung dieser Systeme aufgrund des hohen Kostenaufwands nicht für jeden Landwirt möglich ist. Aufgrund der Tatsache, dass in teure Geräte wie modernen Sensoren, High-Tech Traktoren oder Drohnen investiert werden müssen, würde sich der Einstieg für kleinere Landwirte in die teilflächenspezifischen Bewirtschaftung nicht rentieren. Deshalb wird der geringere Kostenaufwand im Vergleich zu den anderen Systemen als Alleinstellungsmerkmal angesehen.

5 Methodischer Rahmen

Im Folgenden soll eine Abwägung und Entscheidung eines Vorgehensmodell aus der Mensch-Computer-Interaktion stattfinden, die als Grundlage für den weiteren Bearbeitungsprozess im Projekt dient.

5.1 User-centered Design oder Usage-centered Design

Als Erstes ist es wichtig, zwischen den zwei Gestaltungsprinzipien “User-centered Design” (Nutzerorientierte Gestaltung) und “Usage-centered Design” (Nutzungsorientierte Gestaltung) abzuwählen. Während sich das Usage-centered Design auf die Nutzung des Systems konzentriert, werden die Aufgaben, Ziele, Bedürfnisse und Eigenschaften des Nutzers bei der Entwicklung in den Mittelpunkt gestellt, um eine hohe Gebrauchstauglichkeit zu erreichen. Aufgrund der homogenen Zielgruppe “Landwirte” erweist sich der Entwicklungsprozess mit Hilfe des User-centered Design als eine sinnvolle Herangehensweise, um den Erfordernissen und den Anforderungen der Nutzer gerecht zu werden. Dabei werden die Nutzer schon früh aktiv an der Entwicklung und Gestaltung miteingebunden.

5.2 MCI-Vorgehensmodelle

Im nächsten Schritt sollen die einzelnen Vorgehensmodelle miteinander betrachtet, projektspezifisch abgewägt und abschließend für eins entschieden werden, denn auf deren Methodik basieren die weiteren Entwicklungsprozesse im Projekt.

Das Vorgehensmodell “Scenario based usability engineering” von Rosson und Carroll stellt das Schreiben und Arbeiten mit Szenarien in den Mittelpunkt der Entwicklung. Dieses Vorgehen bringt durch die Vielzahl der zu erstellenden Szenarien einen intensiven Zeitaufwand mit sich, der im Rahmen des Projekts unser Erachtens zu viel wäre. Außerdem werden die Szenarien aus der kognitiven Sicht der Entwickler erstellt. Jedoch ist unser Wissen in Bezug auf die Domäne begrenzt, wodurch das Arbeiten mit Szenarien nicht in den Mittelpunkt der Entwicklung gestellt werden soll. Dennoch könnte die Anwendung von Szenarien als eine Methode Berücksichtigung finden.

Das Modell “Discount Usability Engineering” von Nielsen ist wegen des kostengünstigen, schnellen und einfachen Entwicklungsprozesses bekannt. Es würde im Zeitrahmen des Projekts sicherlich eine gute Anbindung an das Projekt finden. Jedoch werden die Nutzer nicht so aktiv in die Entwicklung eingebunden wie bei anderen Vorgehensmodellen. Außerdem wird die Maximierung der Gebrauchstauglichkeit durch eine unzureichende Anforderungsanalyse gefährdet, was unserer Meinung nach ein Ausschlusskriterium ist. Dieses Modell wird für das Projekt nicht in Betracht gezogen.

Das “Usability Engineering Lifecycle” von Mayhew bietet ein genaues und exaktes Vorgehen für ein User-centered Design. Vorteile dieses Modells wären zum einen die Skalierbarkeit, die durch die sogenannten “shortcuts” gewährleistet wird und zum anderen werden Artefakte angegeben, die erstellt werden müssen.

Dadurch wird ein klares Vorgehen geboten. Ein Nachteil jedoch ist, dass die tatsächlichen Nutzer erst in der Installation-Phase, also in der letzten Phase des Modells zum Einsatz kommen. Aufgrund der Tatsache, dass unser Wissen in der Domäne begrenzt ist, ist es sinnvoller so früh wie möglich mit potentiellen Nutzern zu arbeiten. Deshalb kommt dieses Modell ebenfalls nicht in Frage.

Die Wahl fällt somit auf das Rahmenwerk “Prozess zur Gestaltung gebrauchstauglicher interaktiver Systeme” aus der DIN EN ISO 9241, 210. Es gibt ein iteratives Vorgehen vor, in dem die Nutzer aktiv an der Gestaltung und Entwicklung miteinbezogen werden. Hier werden keine konkreten Vorschläge über Methoden oder Artefakte gemacht, sondern Aktivitäten, die im Entwicklungsprozess ausgeführt werden müssen. Deshalb ist es nötig, für diese Aktivitäten konkrete Methoden der Mensch-Computer-Interaktion auszuwählen. Diese sollen im Folgenden genannt und begründet werden.

5.3 Auswahl der MCI-Methoden

Für das Verstehen und Beschreiben des Nutzungskontexts müssen die Nutzer und deren Aufgaben analysiert werden. Dies soll durch die Benutzer- und Aufgabenmodellierung erfolgen. Dafür müssen in der Benutzermodellierung als Erstes die Stakeholder analysiert werden, was anschließend die Grundlage für die Erstellung der User Profiles ist. Anhand der Merkmale der User Profiles sollen Real User gefunden werden, die sich an der Entwicklung beteiligen wollen. Werden jedoch keine Real User gefunden, müssen Personas geschrieben werden, die diese Real User repräsentieren. In der Aufgabenmodellierung sollen die einzelnen Aufgaben der Landwirte untersucht und analysiert werden. Hier bietet es sich an mit Szenarien oder Use Cases, die Aufgaben zu untersuchen und auszuwerten. Um die Aufgaben der Landwirte noch feingranularer zu analysieren, lässt sich die HTA anwenden.

Auf Grundlage der Analyse des Nutzungskontexts sollen in einer Anforderungs-ermittlung funktionale, non-funktionale und organisationale Nutzungsanforderungen identifiziert und spezifiziert werden. Sie werden im Laufe des Projekts iterativ überarbeitet und weiterentwickelt.

Im nächsten Schritt müssen Gestaltungslösungen entwickelt werden, die diese Nutzungsanforderungen erfüllen. Sie dienen als Grundlage von Evaluationen mit Testpersonen. Im Rahmen des Projekts sollen paper-based prototypes ihren Einsatz finden, weil sie im Vergleich zu digitalen Mock-ups weniger zeitaufwendig sind. Dadurch kann im Falle einer Nicht-Erfüllung der Nutzungsanforderungen mithilfe von weiteren Iterationen des Prototypen zur Erfüllung der Nutzungsanforderungen beigetragen werden.

Mit der Erstellung eines Prototypen wird die Grundlage für eine Evaluation mit Testpersonen geschaffen. Hier sollte versucht werden, mit potentiellen Nutzern die Evaluation durchzuführen. Mit der Think-Aloud Methode können alle Gedankenprozesse der Nutzer, also welche Interaktionsschritte sie mit dem Prototypen ausführen würden, aufgenommen werden. Sollte es jedoch nicht möglich sein mit echten potentiellen Nutzern zu testen, müssten mit Usability-Experten mithilfe eines Cognitive Walkthroughs evaluiert werden.

6 Kommunikationsmodell

6.1 deskriptives Kommunikationsmodell

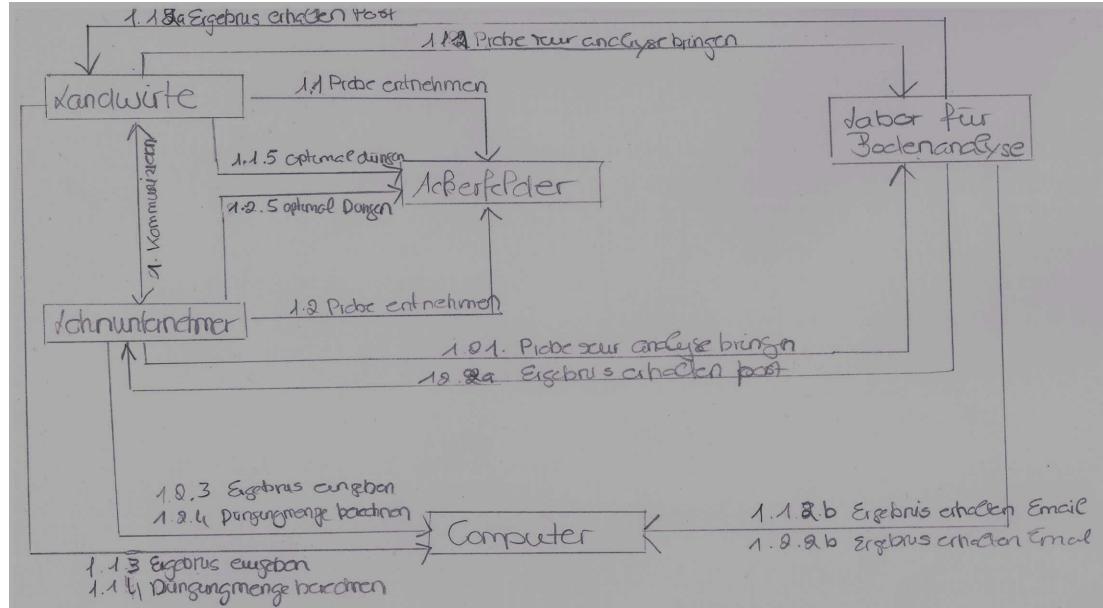


Abbildung 1: deskriptives Kommunikationsmodell

In der Abbildung 1 ist der aktuelle Ist-Zustand der Kommunikation zwischen den entsprechenden Instanzen beschrieben. Anhand der Ergebnisse der Domänenrecherche wird das Modell abgeleitet und dargestellt.

Der Landwirt muss für die Bodenanalyse extern ein Bodenanalyselabor beauftragen, damit er seine Bodenproben aus seinem Ackerfeldern analysieren kann. Neben dem Landwirten kann ein Lohnunternehmer, der auf seinen Feldern arbeitet, Bodenproben entnehmen und sie zur Analyse an das Labor schicken (1.1, 1.1.1, 1.2, 1.2.1).

Das Labor schickt nach der Analyse, dem Landwirten und seinen Mitarbeiter die Ergebnisse per Post zu (1.1.2a, 1.2.2a). Es besteht auch die Möglichkeit, dass das Labor die Ergebnisse per E-mail an den Rechner des Landwirten sendet (1.1.2b, 1.2.2b). Mit den Ergebnissen der Bodenanalyse berechnet der Bauer an seinem Rechner die Menge der Düngermenge, die für die Ackerfelder benötigt werden (1.1.3, 1.1.4, 1.2.3, 1.2.4). Anhand der berechneten Düngemenge können der Landwirt oder der Lohnunternehmer eine optimale Düngung auf den Feldern durchführen(1.1.5, 1.2.5). Der Landwirt und die Lohnunternehmern können auch jederzeit miteinander kommunizieren, um die Arbeit abzusprechen und aufzuteilen.

6.2 präskriptives Kommunikationsmodell

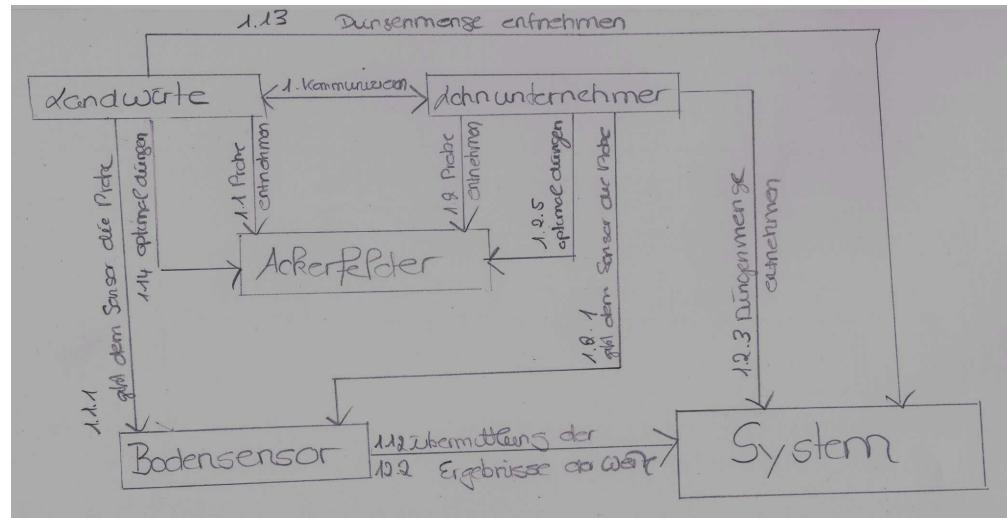


Abbildung 2: präskriptives Kommunikationsmodell

In der Abbildung 2 sieht man an dem Kommunikationsmodell deutlich wie die Kommunikation optimiert werden soll. Das interaktive System soll dem Landwirten helfen schneller und einfacher seine Ackerfelder optimal zu düngen. Die Bodenanalyse erfolgt über einen Bodensensor, der die Messwerte direkt vorort an das System übergibt. Der Landwirt muss nicht mehr auf die Ergebnisse des Labors warten. Außerdem muss er sie nicht mehr selber in das System eingeben.

Das System übernimmt die Ergebnisse des Bodensensor, verarbeitet sie und berechnet dem Landwirte die optimale Düngermenge nach dem Prinzip der teilflächenspezifischen Bewirtschaftung. So spart sich der Landwirte viel Zeit und kann jederzeit optimal düngen.

7 Architekturdiagramm

Das Architekturdiagramm soll die Verteiltheit der Anwendungslogik und die einzelnen Komponenten des System mit ihren Kommunikationsabläufen darstellen.

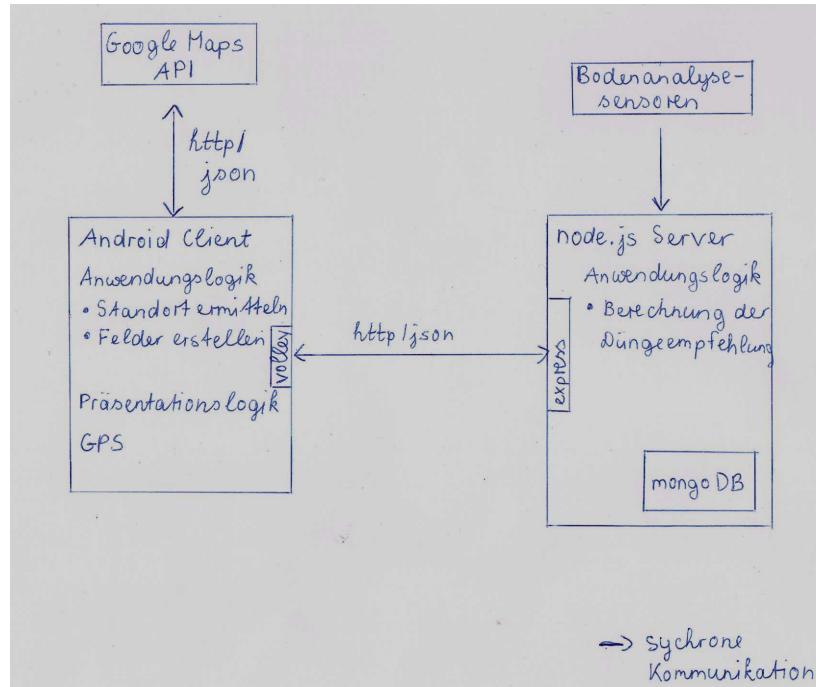


Abbildung 3: Architekturdiagramm

Die Abbildung 3 zeigt die geplante Systemarchitektur des zu entwickelnden Systems. Es handelt sich hierbei um eine Client-Server Architektur, bei der die Anwendungslogik auf beiden Komponenten verteilt läuft. Diese verteilte Anwendung soll außerdem im REST-Stil realisiert werden.

7.1 Anwendungsart

Das zu entwickelnde System soll voraussichtlich als eine mobile Anwendung realisiert werden, denn aus dem Problemraum heraus ergibt sich, dass die Landwirte viel auf ihren Felder unterwegs sind und so nicht wirklich Zeit haben, eine Desktopanwendung zu nutzen, die ihnen bei der Düngung unterstützen soll. Außerdem besitzen heutzutage viele Landwirte ein eigenes Smartphone, weshalb eine mobile Anwendung sinnvoll wäre. Auch werden durch das Smartphone wichtige Daten wie GPS-Standort gesammelt, die in der Domäne eine wichtige Rolle spielen. Für die Realisierung einer mobilen Anwendung stehen viele Möglichkeiten wie iOS, Android oder Windows Phone zur Verfügung. Jedoch wird im Rahmen des Moduls auf die Entwicklung von Android in Java beschränkt, weshalb diese Möglichkeit für die Realisierung des Clients genutzt wird.

7.2 Server

Für die Implementierung des Servers könnten hingegen verschiedene Programmiersprachen (z.B. Java, PHP oder Javascript) genutzt werden. Im Rahmen des Projekts wurde entschieden, den Server mittels Javascript mit dem Framework node.js zu realisieren, da zum einen fundierte Vorkenntnisse 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 einer Anbindung zu einer Datenbank geschehen. Zur Wahl stehen die Key-Value-Datenbank Redis und die dokumentenorientierte Datenbank MongoDB, da dort fundiertes Wissen vorliegt und die Einarbeitung in neue Datenbanken wegfallen würde. Aufgrund der Tatsache, dass Daten hauptsächlich im JSON-Format repräsentiert werden soll, ist die Wahl auf MongoDB gefallen, da JSON-Daten im Vergleich zu Redis besser gespeichert und verarbeitet werden können.

7.3 Datenformat

Da für das System eine verteilte Anwendung im REST-Stil vorgesehen ist, soll die Datenübertragung zwischen den einzelnen Komponenten über das Transportprotokoll http stattfinden. Die Daten, die als Kommunikationsmittel dienen, sollen in Form von JSON repräsentiert werden, da Daten im JSON-Format einfach zu handhaben und zu implementieren sind.

7.4 Externe Schnittstelle

Wie auf der Abbildung 3 zu erkennen ist, hat der Client eine Anbindung an eine externe Schnittstelle. Aus dem Problemraum heraus werden Geo-Daten in Form von Satellitenkarten benötigt. Diese Geo-Daten soll das System von einer externen Schnittstelle beziehen. Zur Wahl stehen die offene Schnittstelle OpenStreetMap API und die Google Maps API. Aufgrund der Tatsache, dass die OpenStreetMap API keine Möglichkeit bietet, mit Satellitenbildern zu arbeiten, wurde die Nutzung der Google Maps API entschieden.

7.5 Externe Datenquellen

Zum System sollen noch Bodenanalysesensoren gehören, die die Bodennährstoffe des Ackers bestimmen sollen. Mit diesen Geräten kann der Landwirt auf seinem Acker Bodenproben entnehmen und direkt vorort untersuchen lassen. Diese Sensoren sollen nach Analyse des Bodens die Ergebnisdaten an den Server senden. Mithilfe dieser Analysedaten kann der Server eine exakte Düngempfehlung für den Landwirten unter Berücksichtigung des heterogenen Boden berechnen und vorschlagen.

8 Risiken

In diesem Abschnitt sollen projektspezifische Risiken, die den Projekterfolg gefährden können, identifiziert werden. Zu diesen Risiken sollen anschließend mögliche Gegenmaßnahmen aufgezeigt werden, die beim Fall eines Eintritts in Einsatz kommen.

8.1 Datenschutz (GPS)

Die Anwendung muss voraussichtlich in der Lage sein, den aktuellen Standort des Nutzers mithilfe von GPS zu ermitteln. Ihr wäre es sozusagen möglich, ein detailliertes Bewegungsprofil vom Nutzer zu erstellen. Für viele Menschen ist Datenschutz ein erheblicher Grund, warum sie den Umgang mit digitalen Anwendungen meiden.

Maßnahme: Dem Nutzer muss transparent dargelegt werden, welchen Zweck und Nutzen die Daten vom Nutzer für das System haben und was mit ihnen gemacht werden.

8.2 Nutzung von externen Schnittstellen

Für die Realisierung des Systems müssen voraussichtlich auf externe Daten in Form von Geo-Daten zugegriffen werden. Diese setzt eine Einbindung von externen Schnittstellen voraus. Jedoch sollten diese externen Schnittstellen in Bezug auf unsere Anwendung kritisch betrachtet werden, denn beim Ausfall einer API würde bspw. ein Großteil der Anwendung nicht mehr funktionieren.

Maßnahme: Alle essentiellen Daten, die von der externen Schnittstelle benötigt werden, müssen in einer anderen Datenbank zwischengespeichert werden, damit im Falle eines Ausfalls der API die Anwendung weiterhin noch funktionieren kann.

8.3 Ungenaue Standortbestimmung durch GPS

Für eine exakte Arbeit in der Landwirtschaft spielt die Nutzung von GPS zur Standortbestimmung eine große Rolle. Je genauer und exakter die Standortbestimmung ist, desto exakter können wichtige Arbeitsprozesse durchgeführt werden insbesondere die bedarfsgerechte Düngung des Bodens. Jedoch bieten Smartphones nie eine exaktere GPS-Standortermittlung als GPS-spezialisierte Geräte.

Maßnahme: In einem Proof-of-Concept soll die Nutzung des GPS getestet werden. Außerdem soll auf Android-Geräten die Anwendung stets nur unter der GPS-Einstellung hohe Genauigkeit laufen.

8.4 Anschaffung von Bodenanalysesensoren

Als ein Alleinstellungsmerkmal des Projekts wurde der im Vergleich zu den anderen Systemen geringere Kostenaufwand genannt. Jedoch werden für das zu entwickelnde System Bodenanalysesensoren benötigt, die die Nährstoffvorrat im Boden messen können. Aufgrund der Tatsache, dass Landwirte von kleineren bis mittleren Betrieben zu der Haupt-Zielgruppe gehören, könnte es passieren, dass sie unser System ebenfalls nicht leisten können.

Maßnahme: Es sollen auf kostengünstige Bodenanalysesensoren zurückgegriffen werden, die der Landwirt auf seinem Feld nutzen und direkt vorort den Boden analysieren kann. Dadurch spart er sich die Kosten der Bodenanalyse durch ein Labor, wodurch die Investitionsmöglichkeit dieser Sensoren für die Landwirt ermöglicht werden soll.

9 Spezifikation der Proof-of-Concepts

Im folgenden Abschnitt soll die Spezifizierung der Proof of Concepts erfolgen. Die Risiken, die die Implementierung und die Realisierbarkeit des Projekts gefährden, werden beschrieben, spezifiziert und müssen zwingend getestet werden.

9.1 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.

9.2 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:

9.3 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.

9.4 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.

10 Rapid Prototyping

Im folgenden Abschnitt soll die Realisierung des Rapid Prototypings sowie die Prüfung der aufgestellten Proof-of-Concepts erläutert und dokumentiert werden. Für die Erstellung des Rapid Prototypings wurden die Proof-of-Concepts aus dem Abschnitt 9.1, 9.3 und 9.4 und das Alleinstellungsmerkmal aus dem Abschnitt 4.1 adressiert.

Es wurde ein node.js-Server aufgesetzt, der eine Anbindung an die Datenbank MongoDB besitzt. Außerdem verarbeitet der Server zurzeit ankommende GET- und POST-Requests. Bei einem GET-Request soll der Server, Daten von der Datenbank auslesen und als Response senden. Beim POST-Request bekommt der Server Daten, in unserem Fall statische Koordinaten für ein Polygon, die er in der Datenbank festschreiben soll. Für den Client wurde wie im Architekturdiagramm festgelegt ein Android Client realisiert. In der Abbildung 4 wird die Main-Activity dargestellt. Sie bietet lediglich die Möglichkeit auf zwei andere Activities zu wechseln.

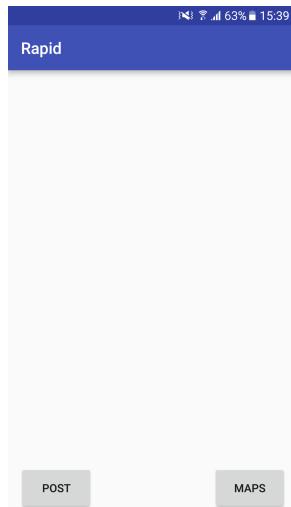


Abbildung 4: Main-Activity des Android Clients

Drückt man auf den Post-Button, wird in die Post-Activity (siehe Abbildung 5) gewechselt. Diese Activity hat im Prototyping zwei wichtige Aufgaben. Zum einen ist es möglich, durch das Betätigen des Button 'Request senden' ein POST-Request mit vorgefertigten Koordinaten an den Server zu senden. Diese Daten werden, sofern sie in der Datenbank festgeschrieben sind, in der Post-Activity durch ein GET-Request an den Server als Response dargestellt (siehe Abbildung 5(b)).

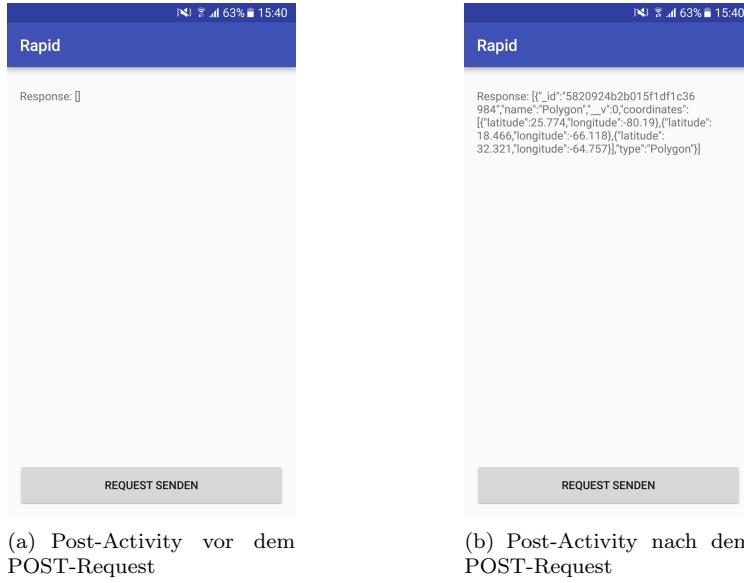


Abbildung 5: Post-Activity des Android Client

Wechselt man in der Main-Activity in die Maps-Activity durch das Betätigen des Button 'Maps', gelangt man auf eine Activity, die mithilfe der Google Maps API eine Satellitenkarte darstellen kann. Wie auf der Abbildung 6 zu sehen ist, lässt sich ein Ausschnitt der Karte mit dem jetzigen Standort erkennen. Der Zustand, der in der Abbildung 6(a) zu erkennen ist, liegt ohne Daten in der Datenbank vor. Es wurde noch keine Daten an den Server gesendet. Werden in der Post-Activity Daten durch ein POST-Request an den Server gesendet, greift die Map-Activity durch ein GET-Request auf diese Daten zu, so dass er basierend auf diesen Koordinaten das Polygon in Abbildung 6(b) zeichnen kann.



(a) Map-Activity ohne Polygon



(b) Map-Activity mit Polygon

Abbildung 6: Map-Activity des Android Client

Durch das Rapid Prototyping konnte gezeigt werden, dass die geprüften Proof-of-Concepts erfolgreich getestet wurden. Das Einbinden der Google Maps API an den Client verlief problemlos. Das Darstellen von 10 verschiedenen Orten auf der Satelliten-Karte funktioniert einwandfrei. Außerdem konnte durch das Senden von verschiedenen Polygon-Daten vom Client an den Server und das Anfordern der Daten durch den Client in Form von gezeichneten Polygonen bewiesen werden, dass die Client-Server Kommunikation zwischen den Komponenten erfolgreich läuft und das Erstellen von Polygonen in Google Maps funktioniert.

11 Literaturverzeichnis

- Dr. J. Pößneck: 'Precision Farming', unter https://www.landwirtschaft.sachsen.de/landwirtschaft/download/Precision_Farming-Endfassung-Internet-v2.pdf (abgerufen am 25.10.2016)
- Precision Farming: 'key technologies and concepts', unter <http://cema-agri.org/page/precision-farming-key-technologies-concepts> (abgerufen am 26.10.2016)
- Dr. Norbert Uppenkamp: 'Entwicklungen in der Düngetechnik', unter <https://www.landwirtschaftskammer.de/landwirtschaft/technik/aussenwirtschaft/duengetechnik.htm> (abgerufen am 20.10.2016)
- Google: 'Google Maps API Android', unter <https://developers.google.com/maps/documentation/android-api/> (abgerufen am 05.11.2016)
- Landwirtschaftskammer Nordrhein Westfalen: 'Wann und wie darf man Gülle ausbringen?' unter <https://www.landwirtschaftskammer.de/landwirtschaft/ackerbau/duengung/guelle/verordnung/guelle-ja.htm> (abgerufen am 21.10.2016)
- Bauernhof: 'Düngung', unter <http://www.bauernhof.net/glossar/duengung/> (abgerufen am 11.10.2016)
- Optifert: <http://www.optifert.eu/> (abgerufen am 04.11.2016)