

# **Dokumentation des Projekts HarvestHand**

Studierende

Franziska Gonschor  
Sergej Atamantschuk

Betreuer

Robert Gabriel  
Prof. Dr. Gerhard Hartman  
Prof. Dr. Kristian Fischer

## Inhaltsverzeichnis

Inhaltsverzeichnis.....	II
Tabellenverzeichnis.....	IV
1. Einleitung.....	1
1.1. Nutzungsproblem.....	1
1.2. Ziel des Projekts .....	1
2. Proof of Concepts.....	2
2.1. Benutzer anlegen .....	2
2.2. Eintrag mit Ackerdaten erstellen .....	2
2.3. Collaborators dem Eintrag hinzufügen .....	2
2.4. Effiziente Datenübertragung.....	3
2.5. Auswertung der Bodendaten .....	3
2.6. Klimatische Daten der Externen Dienste .....	3
2.7. Effiziente Erstellung interaktiver Tutorials.....	4
2.8. Erstellung der Anleitungen für Mobiltelefone .....	4
2.9. Didaktische Darstellung der Informationen.....	5
3. Requirements Analysis .....	5
3.1. User Profiles .....	5
3.2. Personae.....	7
3.3. Contextual Task Analysis .....	7
3.3.1. Work Environment Analysis .....	9
3.3.2. User Task Organization Model .....	10
3.4. Systemanforderungen.....	11
3.4.1. Funktionale Anforderungen .....	11
3.4.2. Organisationale Anforderungen.....	12
3.4.3. Qualitative Anforderungen .....	12
3.5. Plattform Constraints .....	13
3.6. General Design Principles.....	14
3.7. Usability Goals.....	14
3.8. Fazit .....	17
4. Work Reengineering.....	18
5. Conceptual Model Design .....	19
5.1. Conceptual Mock – ups.....	21
5.2. Evaluation.....	24
6. Screen Design Standards.....	25
6.1. Control Standarts .....	25
6.2. Process window standarts.....	25

6.3.	Weitere Standarts .....	26
6.4.	SDS Prototyping.....	27
7.	Detailed UI Design.....	27
7.1.	Fazit .....	31
8.	Evaluation.....	31
8.1.	Ergebnisse der Evaluation .....	31
8.2.	Fazit .....	32
9.	Systemarchitektur .....	33
9.1.	Ressourcen .....	33
9.2.	Datenstrukturen.....	35
9.3.	Anwendungslogik des Systems .....	40
9.3.1.	Bodenanalyse .....	40
9.3.2.	Interaktive Tutorials .....	41
9.4.	Datenschutz.....	42
10.	Fazit .....	42
11.	Installationsdokumentation .....	44
11.1.	Systemanforderungen.....	44
11.2.	Server .....	44
11.3.	Client .....	44
12.	Anhang .....	45
12.1.	POCs (überarbeitet).....	45
12.1.1.	Benutzer anlegen .....	45
12.1.2.	Collaborators dem Eintrag hinzufügen .....	45
12.1.3.	Effiziente Datenübertragung.....	45
12.1.4.	Auswertung der Bodendaten .....	46
12.1.5.	Effiziente Erstellung interaktiver Tutorials.....	46
12.1.6.	Klimatische Daten der Externen Dienste.....	46
12.1.7.	Erstellung der Anleitungen für Mobiltelefone .....	47
12.1.8.	Didaktische Darstellung der Informationen .....	47
12.2.	Personae.....	48
12.3.	Evaluation.....	50
12.3.1.	Tasks .....	50
12.3.2.	User 1 .....	52
12.3.3.	User 2 .....	54
12.3.4.	User 3 .....	56
12.3.5.	User 4 .....	58
12.4.	Glossar.....	60

12.5.	Quellenverzeichnis .....	61
-------	--------------------------	----

## Abbildungsverzeichnis

Abbildung 1 Leistungen Weather Underground API.....	4
Abbildung 2 HTTP Request.....	4
Abbildung 3 User Task Organization Model.....	10
Abbildung 4 Reengineered Task Organization Model.....	18
Abbildung 5 Sign up und Sign in .....	21
Abbildung 6 Liste mit Entries .....	22
Abbildung 7 Neuen Eintrag erstellen .....	22
Abbildung 8 Entrie Details.....	23
Abbildung 9 Tutorial zu Einem Eintrag.....	23
Abbildung 10 Ausführliche Anbauerklärung .....	24
Abbildung 11 Sign In Fragment .....	28
Abbildung 12 Sign up Fragment .....	28
Abbildung 13 Liste mit Entries .....	29
Abbildung 14 Item - Präsentation für Analphabeten.....	29
Abbildung 15 Erstellung eines Entrys.....	29
Abbildung 16 Dialogbox .....	29
Abbildung 17 Tutorial Activity.....	30
Abbildung 18 Entry Details.....	30
Abbildung 19 Fehlerbehandlung.....	30

## Tabellenverzeichnis

Tabelle 1 Data Collection .....	7
Tabelle 2 Style Guide.....	26

## 1. Einleitung

---

### 1.1. Nutzungsproblem

In vielen Entwicklungsländern ist die Bevölkerung sehr hoch und ein Großteil davon sind Kinder. Die Menschen leben in Armut und haben meist nur ungefähr einen US-Dollar pro Tag zum Leben zur Verfügung. Ca. 20% der dort lebenden Menschen sind stark unterernährt, was sich auf die dortige Armut zurückführen lässt. Oft wird dies durch mangelnde Aufklärung z.B. im Bereich Ackerbau verursacht, da das nötige Wissen für einen ertragreichen Anbau fehlt. Die Aufklärung der Bevölkerung wird jedoch dadurch erschwert, dass in diesen Entwicklungsländern ein Großteil der Menschen Analphabeten sind. So ist eine Aufklärung in Form von schriftlichen Informationen nicht möglich. Beispiel für diese Länder sind: Burkina Faso, Peru, Togo und Nepal. Die Bildungsbarriere zwischen den Entwicklungs- und den Industrieländern stellt ein erhebliches Problem dar.

### 1.2. Ziel des Projekts

Ziel ist es ein System zu entwerfen, welches die Bevölkerung der Entwicklungsländer im Bereich Ackerbau aufklärt. Dabei soll es zum einen den Menschen direkt etwas vermitteln, beispielsweise durch Abbildungen und Erklärungen die als Ton abgespielt werden können. Zum anderen, soll es auch die Helfer vor Ort bei der Vermittlung wichtiger Strategien und Untersuchungen in diesem Bereich unterstützen. So können zum Beispiel ermittelte Werte des Bodens, gespeichert und verarbeitet werden, so dass eine Anbauempfehlung vorgeschlagen wird. Außerdem werden kurzfristige Informationen über bestimmte Ereignisse, z.B. das Wetter (Dürreperioden oder Niederschlag) an die Landwirte gesendet. Doch nicht nur der aktuelle Anbau soll gefördert werden, sondern das System soll die Landwirtschaft nachhaltig unterstützen, so dass auch langfristige Klimaveränderungen in Betracht gezogen werden. In Folge dessen, sollen die Ackerbauern darüber informiert werden, ob es beispielsweise noch Sinn macht Kakaopflanzen in den nächsten 5 Jahren anzubauen, oder ob sie auf eine andere Pflanze zurückgreifen sollen, da durch die Klimaveränderungen die Erträge einbrechen könnten.

Besonders wichtig ist es dabei, die Wissensbarriere zwischen den Industrie- und Entwicklungsländern zu schließen. Die Menschen in Afrika sollen von unserem Wissen profitieren können und daher werden die Informationen für sie gerecht aufbereitet.

## 2. Proof of Concepts

---

Die Durchführung der POCs sollte wesentliche Merkmale des Systems in früheren Phasen der Entwicklung testen und die Realisierbarkeit des Systems gewährleisten. Im Folgenden wird die Durchführung der POCs dokumentiert. Die aus dem Konzept Überarbeiteten POCs sind in dem Anhang zu finden.

### 2.1. Benutzer anlegen

Dieses POC ist relativ einfach realisierbar und aus dem Grund getestet, dass das Entwicklungsteam vorher keine Erfahrung mit der Userauthentifikation hatte. Das System erfordert wenige persönliche Datenfelder zu Registrierung. Falls der User kein Smartphone hat, wird er vom System über Telefonnummer identifiziert. Serverseitig werden zur Authentifikation Module *Passport* und *Sessions* verwendet. Clientseitig erfolgt die Dateneingabe über die Textfelder der Android App. Wichtig ist zu beachten, dass das UI Concept für die Analphabeten akzeptable Informationspräsentationen enthält. Nach der Durchführung wurde festgestellt, dass die UI Elemente entsprechende visuelle Form haben oder verbal dargestellt werden sollen. Des Weiteren sollen die Helfer den Benutzern bei der Erstellung des Profils unterstützen.

Falls der Benutzer kein Smartphone besitzt und als Konsequenz kein Userprofil hat, sollte falls vorhanden seine Telefonnummer eingetragen werden. Falls der Benutzer gar kein Gerät besitzt, ist die direkte Interaktion mit dem System nicht möglich. Daher werden alternative Authentifikationsmethoden betrachtet.

### 2.2. Eintrag mit Ackerdaten erstellen

Erstellung des Eintrags ist eine weitere relativ einfache Input Funktion. Hier werden zur Verarbeitung relevante Ackerdaten eingegeben und in DB gespeichert. Scheitern könnte das POC bei der Erhebung der spezifischen Bodendaten vor Ort. Die Recherche während der Durchführung hat ergeben, dass es passende Werkzeuge gibt, mit den man die Werte vor Ort ermitteln kann. So gibt es z. B. einen kompakten PH – Wert Messer. Die Helfer müssen in dem Fall über solche Werkzeuge verfügen. Zudem gibt es mehrere APIs, die die Daten über Bodentemperatur - und Feuchtigkeit, Luftfeuchtigkeit usw. Regional zur Verfügung stellen. Im Fallback Fall kann man auf die APIs zugreifen. Die Datenvollständigkeit kann einfach auf dem Client vorm Speichern kontrolliert werden. Daher kann das System mit großer Wahrscheinlichkeit nicht an der Unvollständigkeit der Ackerdaten scheitern. Die Angabe der Daten erfolgt in der Regel vom Helfenden, so dass die eine gewöhnliche Darstellung der UI Elementen ausreicht.

Die Daten werden vom Client im JSON – Format an den Server übertragen, dort verarbeitet und in der Datenbank gespeichert.

### 2.3. Collaborators dem Eintrag hinzufügen

Bei Speichern eines Eintrages kann der User, der den Eintrag erstellt, in der Datenbank nach anderen Usern suchen, um die dem Eintrag als sogenannte Collaborators hinzufügen. Die Collaborators bekommen somit die Zugriffsmöglichkeit auf den jeweiligen Eintrag. Der Kritische

Punkt ist hier die Suche nach einem bestimmten User. Daher wird die Suche nach Email oder Passwort durchgeführt, sodass immer nur ein bestimmter User identifiziert wird. Die Durchführung hat ergeben, dass die Datenübertragung bei der Suche nach dem User sehr schnell passiert. Doch im echten Nutzungskontext des Systems herrschen andere Bedingungen, daher werden die Suchzeiten von 5-10 Sekunden akzeptiert.

## 2.4. Effiziente Datenübertragung

Die früheren Tests ergaben die Datenmengen bei einer Übertragung von ca. 10 – 30 kbps und die Übertragung erfolgte in wenigen Millisekunden. Die Daten wurden mit Android Studio Monitoringtools gemessen. Damit war die Durchführung des POCs erfolgreich. Um dies zu erreichen wurde die Repräsentation der Daten im JSON - Format so gestaltet, dass nur primitive Datentypen verwendet wurden. Das Modell der Tutorials wurde z.B. so entworfen, dass die Abweichung der Bodenwerte nach der Bodenanalyse in einer Art Flags gespeichert waren. Auf diese Weise konnte z.B. der PH – Wert auf NORM, GREATER oder LESS gesetzt werden, wobei den Konstanten NORM, GREATER, LESS ein integer wert zugewiesen ist und dem System bekannt ist, wo es Abweichungen gibt. Dies wurde für die Testzwecke und aus Zeitgründen sehr vereinfacht dargestellt und wird in folgenden Entwicklungsschritten weiter ausgebaut.

## 2.5. Auswertung der Bodendaten

Die Bodenanalyse umfasst den Vergleich der angegebenen Daten mit den normalen Standartwerten. Bei Abweichungen werden die Gegenmaßnahmen berechnet und in Form von Tutorials in der Datenbank gespeichert. Der Client kann im Folgenden auf die Tutorials zugreifen. So wird es z.B berechnet, wie viel und wie oft man eine Pflanze gießen soll in Abhängigkeit von den für die Pflanze bestimmten Standartwerten und der Fläche des Ackers, Art des Bodens, der Wetterbedingungen usw.

Bei der Durchführung wurde festgestellt, dass es keine passende Pfalzen Datenbank existiert, die direkt über ein API im System zur Bodenanalyse verwendet wird. Somit ist eine Datenbank zu vergleich der Werte notwendig. Daher wird Erstellung einer eigene Pflanzentopologie überlegt, um die für die Bodenanalyse relevanten Daten zu standarisieren und die Vollständigkeit zu gewährleisten.

## 2.6. Klimatische Daten der Externen Dienste

HarvestHand benötigt für eine vollständige Funktionalität Daten zur lokalen Wetterlage des Benutzers. Sie nehmen immerhin Einfluss auf den Anbauvorgang. Daher muss eine Wetter API implementiert werden, welche Daten zum Wetter zur Verfügung stellt. Als erstes musste viel Zeit in die Recherche nach einer geeigneten API investiert werden. Einige sind nicht frei verfügbar, andere gewährleisten nur eine bestimmte Anzahl von Abfragen in einem bestimmten Zeitraum.

Abschließend wurde sich für die Weather Underground API entschieden. Sie bietet mehr Daten als andere APIs an, es können 10 Abfragen pro Minute und 500 pro Tag getätigt werden. Dies ist als freie Version gegenüber anderen APIs eine hohe Anzahl von Abfragen pro Tag. Außerdem war ein ausschlaggebendes Kriterium, dass wir bereits das Wetter für die nächsten 10 Tage voraussagen können.



Abbildung 1 Leistungen Weather Underground API

Um einen Key für die Nutzung zu erhalten, wurde ein Konto angelegt bei welchem Daten zum Projekt und dem Nutzen der API angegeben werden mussten. Anschließend wurde der Key direkt mitgeteilt und die API konnte implementiert werden.

API Requests funktionieren ausschließlich über HTTP. Beispielhaft wurde ein Request für die Stadt San Francisco durchgeführt.

```
var requestUrl = 'http://api.wunderground.com/api/28182cd556dbb993/conditions/q/CA/San_Francisco.json'
```

Abbildung 2 HTTP Request

Die Daten werden im JSON Format präsentiert. Man erhält zum einem ausführliche Daten zur Lokalität und zum anderen detaillierte Wetterdaten. Aus dieser großen Menge an Daten wurden die für das System relevanten Daten herausgefiltert. Der Anbieter bezieht die Daten von einigen Wetterstationen, sodass die Datenvalidität gewährleistet werden soll. Des Weiteren wurden zahlreiche weitere Wetter – APIs betrachtet, damit alternative vorliegen.

## 2.7. Effiziente Erstellung interaktiver Tutorials

Für die Ermittlung der Daten wurde ebenso Android Studio Monitoringtools verwendet. Das POC wurde durch Darstellung der Tutorials mit Bildern getestet. Die Materialien werden Dabei auf dem Lokalen Speicher des Geräts abgelegt und bei Verwendung in den RAM geladen. Daher ist besonders wichtig darauf zu achten, dass die Materialien in möglichst komprimierter Form vorliegen und nicht zu viele Ressourcen verbrauchen. Die Durchführung lieferte akzeptable werte, sodass das POC als erfolgreich gilt. Bei der RAM Nutzung wurden Werte im Bereich zwischen 5 bis 40 MB festgestellt von dem Start des Programms bis zur Darstellung der Tutorials. Die Prozessorbeltastung lag im Schnitt bei 10 Prozent.

## 2.8. Erstellung der Anleitungen für Mobiltelefone

Für die Erstellung und das Verschicken der SMS – Nachrichten wurde ein API zu einem SMS Gateway getestet. Der Inhalt der SMS – Nachrichten wird im JSON – Format mit einem Request inklusive der Zielnummer an den Dienst geschickt und kann von dort weltweit weiter



übermittelt werden. Zum Testen wurde API von websms (websms.de) verwendet. Der kritische Punkt war, dass dieses Unternehmen kostenlos nur eine 14 Tage Testversion von dem Dienst anbietet. Zur permanenten Nutzung dieser Funktion sollte ein anderer Anbieter recherchiert werden.

## 2.9. Didaktische Darstellung der Informationen

Um dieses POC zu Testen wurden Information mit passenden Bildern dargestellt. Es wurden Icons erstellt um die Information möglichst deutlich für die Analphabeten präsentieren zu können. So wurden z.B. Buttons mit den Icons versehen. Des Weiteren wurde festgestellt, dass bestimmte Elemente sich nicht in visueller Form darstellen lassen. In dem Fall wurde als Lösung eine verbale Präsentation des Elements entwickelt. Dabei kann bei Interaktion mit dem Element sein Inhalt oder Funktionalität von der App vertont werden. Im Android System eignet sich ideal hierfür die Klasse **MediaPlayer**. Die Klasse ist dem Entwicklerteam von früheren Projekten bekannt und wurde schon zur Vertonung bestimmter Elemente verwendet. Daher wird dieses Konzept auch für dieses Projekt übernommen.

Im Prozess der Durchführung des POCs wurde die **TextToSpeech** Klasse von Android entdeckt, die es ermöglicht dynamische Textinhalte auf dem Bildschirm vorzulesen. Die Implementierung der Klasse ist wesentlich einfache und die Klasse ist flexibler. Unten ist ein Beispiel vorgeführt, wie man den Inhalt eines **TextViews** beim Anklicken vorlesen lässt. Der Text kann zu dem in verschiedenen Sprachen vorgelesen werden. Da ein **onClickListener** praktisch an jedes Element angebunden werden kann, können auch Inhalte von Bildern vertont werden, um die für die Analphabeten noch deutlicher zu präsentieren.

```
textView.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String toSpeak = "Speak";
        textToSpeech.speak(toSpeak, TextToSpeech.QUEUE_FLUSH, null);
    }
});
```

## 3. Requirements Analysis

### 3.1. User Profiles

#### Landwirt

Merkmal	Ausprägung	
	Alphabet	Analphabet
Demographische Charakteristiken	Jeden Alters und Geschlechts, sind meistens Familienbetriebe, Wohnhaft in einem Entwicklungsland	Jeden Alters und Geschlechts, sind meistens Familienbetriebe, Wohnhaft in einem Entwicklungsland
Qualifikationen	Haben meistens mittlere schulischen Qualifikationen vorzuweisen	Haben meistens keine schulischen Qualifikationen vorzuweisen
Fachwissen	vorhanden, bisheriges Wissen basiert auf Tradition und erlernten Qualifikationen, ist	Kaum vorhanden, bisheriges Wissen basiert auf Tradition, ist aber oft nicht auf dem neuesten

	aber oft nicht auf dem neuesten Stand	Stand
Fähigkeiten bzw. Einschränkungen	Können Lesen und Schreiben, sind meist in der Lage körperliche Arbeit zu verrichten	Fehlende Lese- und Schreibkompetenz (Analphabetismus), sind meist in der Lage körperliche Arbeit zu verrichten
Verfügbare Technologien	Smartphones, Tablets, ältere Mobiltelefone	Smartphones, ältere Mobiltelefone
Computerkenntnisse bzw. -erfahrung	Teilweise vorhanden	Kaum vorhanden
Motivation	Ernteerträge steigern und somit ihre Lebensumstände verbessern, Qualifikationen steigern	Ernteerträge steigern und somit ihre Lebensumstände verbessern
Produkterfahrung	Keine, benötigt Einführung in das System	Keine, benötigt Einführung in das System
Aufgaben	Anbauempfehlungen und Tipps und relevante Daten zum Ackerbau erhalten	Anbauempfehlungen und Tipps zum Ackerbau erhalten
Auswirkung von Fehlern	Niedrige Ernteerträge oder ganze Ernteauffälle, sorgen für Verschlechterung der Lebensumstände, Wissenstand wird nicht verbessert	Niedrige Ernteerträge oder ganze Ernteauffälle, sorgen für Verschlechterung der Lebensumstände
Einstellungen und Werte	Technische Lösungen sind bekannt	Sind nicht den technischen Lösungen zugeneigt, die Sitten der Vorfahren haben größeren Wert

## Helfer

Merkmal	Ausprägung
Demographische Charakteristiken	Über 18 Jahre, können männlich und weiblich sein, jeder Familienstand ist möglich, Wohnhaft in einem Industrieland
Qualifikationen	Abgeschlossene Berufsausbildung oder Studium im Bereich Landwirtschaft
Fachwissen	Allgemeines Wissen über die Funktionsweise in der Domäne
Fähigkeiten bzw. Einschränkungen	Können Fachwissen vermitteln, allerdings nicht in afrikanischer Sprache oder nicht in für Analphabeten akzeptabler Form
Verfügbare Technologien	Smartphones, Tablets, Computer
Computerkenntnisse bzw. -erfahrung	Sehr gut
Motivation	Berufliche Kompetenz, den Menschen in Entwicklungsländern Wissen vermitteln und ihr Leben somit zu verbessern.
Produkterfahrung	Vorhanden, wurde zuvor genau in das System eingeführt um das Lehrmaterial anwenden zu

	können
Aufgaben	Den Helfer bei der Vermittlung von Wissen durch Visualisierungen, Animationen und Audio unterstützen.
Auswirkung von Fehlern	Die Wissensvermittlung kann nicht vollständig stattfinden.
Einstellungen und Werte	Sind den technischen Lösungen zugeneigt, bevorzugen Verwendung von IT-Systemen

### 3.2. Personae

Da die Stakeholderanalyse und Userprofiles ein sehr breites Spektrum an unterschiedlichen Benutzern geliefert haben, wurden die Personae von erstellt, um ein klareres Bild von den potenziellen Benutzern des System zu machen. Die vorherigen Analysen ergaben zwei Schlüsselakteure – Helfer der Hilfsorganisationen und Landwirte in den Entwicklungsländern. Doch es gibt verschiedenste Organisationen und verschiedene Entwicklungsländer, in denen unterschiedliche Lebensbedingungen herrschen. Um die Benutzer besser zu verstehen und eine gewisse Abgrenzung zu schaffen, eignen sich optimal die Personae. Die Personae findet man im Anhang 9.2

### 3.3. Contextual Task Analysis

Die Contextual Task Analysis soll Bedürfnisse und Wünsche der Benutzer an das System identifizieren. Dafür sollte das Umfeld der Benutzer betrachtet werden und es können Interviews mit zukünftigen Benutzern geführt werden, um Erfordernisse identifizieren zu können.

Im Bereich dieses Nutzungskontexts ist es dem Entwicklerteam nicht möglich, den Alltag der Benutzer genau zu beobachten und zu analysieren oder ein Interview mit zukünftigen Nutzern zu führen. Daher werden im Folgenden die Tasks anhand der angelegten User Profiles abgeleitet.

Aus der Stakeholderanalyse lassen sich zwei primären Stakeholdergruppen identifizieren – Helfer (von Hilfsorganisationen) und Landwirten der Entwicklungsländer. Im ersten Schritt wurde der Arbeits - und Kommunikationsumfeld dieser beiden Gruppen beobachtet und Informationen gesammelt. Da die direkte Konfrontation mit den primären Usern nicht möglich ist, müssen die Information aus den Recherchen bezogen werden. In der folgenden Tabelle werden die gesammelten Infos dargestellt.

*Tabelle 1 Data Collection*

Actor	Trigger	Use Case(Task)	Task Scenario Sequence	Errors, Problems, Comments
Landwirt	Pflanzenwachstum steigern	Landwirte müssen Wasser aus den Brunnen holen, welches sie zu	1. Bodentrockenheit visuell feststellen 2. Wasser aus dem	Bodentrockenheit kann nur an der Oberfläche festgestellt werden, die

		ihren Feldern transportieren.	Brunnen beziehen 3. Behälter zu den Feldern transportieren 4. Pflanzen gießen	benötigte Wassermenge kann nicht bestimmt werden
		Pflanzen benötigen Dünger um den niedrigen Nährstoffgehalt im Boden ausgleichen zu können.	1. geringes Pflanzenwachstum wird festgestellt 2. Düngemittel werden ggf. gekauft 3. Düngemittel werden auf den Äckern verteilt	Landwirt weiß nicht, wie und womit er düngen soll und in welcher Menge.
	Qualitativ hochwertige Pflanzen ernten	Pflanzen werden nach der Reifezeit geerntet.	1. Pflanzen werden auf Reife geprüft 2. Pflanzen werden zu Nahrung zubereitet oder auf dem Markt verkauft	Fehlendes Wissen zu Reifegraden von Pflanzen
	Fachwissen erweitern	Neue Techniken und Methoden zum Ackerbau erlernen	1. Informationen sichten 2. Informationen verinnerlichen und verstehen	Analphabeten können nur erschwert Informationen aufnehmen.
	Äcker nachhaltig und effizient bearbeiten	Erlernen neuer Methoden um Felder zu bearbeiten, da Maschinen fehlen.	1. neue Methoden kennenlernen 2. Methoden anwenden	Alle Äcker können nur per Hand bearbeitet werden, was oft sehr aufwendig und mühsam ist. Oft sind Böden steinig und halten kaum Wasser.
	Wetterunabhängiger Anbau	Nach starken Regenfällen den Böden Nährstoffe zuführen.	1. starke Regenfälle bemerken 2. Äcker düngen	Viele Böden sind unfruchtbar, da durch die starken Regenfälle die Nährstoffe ausgewaschen werden.
	Unfruchtbare Böden wieder fruchtbar machen	Bodenfruchtbarkeit wiederherstellen.	1. Boden düngen 2. Boden wässern 3. Schädlingsbekämpfung	Viele Ackerflächen werden bis auf ihre letzten Nährstoffe ausgeschöpft. Anschließend liegen diese Flächen 30 Jahre brach bevor neue Pflanzen angebaut werden können.
	Unabhängigkeit vom Staat	Genügend Pflanzen zum Ernähren der Familien und dem Verkauf auf dem Markt erzeugen.	1. Pflanzen ernten 2. Familien ernähren 3. übrige Ernte verkaufen	Die Entwicklung einer gesunden Landwirtschaft wird verhindert, in dem Subventionen ohne Ziel und zeitlichen Rahmen verteilt werden.
Helfer	Gefährdete Existenz der Landwirthaushalte	Unterstützen der Bauer bei Findung und Umsetzung einer Lösung	1. Problem identifizieren 2. Informationen zum bestimmten Problem beziehen 2. Passende Lösung entwickeln 3. Lösung umsetzen	Die Informationen zu den verschiedensten Pflanzenarten müssen gesammelt werden. Individuelle Maßnahmen müssen erarbeitet werden. Ständige

				Begleitung der Landwirte nötig
	Wissensbarriere schließen	Fachwissen vermitteln in Form von Schulungen vor Ort	1. Informationen vermitteln 2. Lehrmaterial zur Verfügung stellen	Helfer und Landwirte sprechen unterschiedliche Sprachen. Schriftliche Informationen müssen für Analphabeten in ein Format übertragen werden, welches sie verstehen.
	Schulung	Übermittlung des Fachwissens in einer Lerngruppe	1. Aufbereitung der Informationen im Vorfeld 2. Lerngruppen werden organisiert 3. Information wird in visueller und verbaler Form übermittelt	Lern Gruppen müssen organisiert werden. Die Informationen müssen in einer für Alphabeten und Analphabeten geeigneter Form dargestellt werden

### 3.3.1. Work Environment Analysis

Entwicklungsländer sind geprägt von einer hohen Bevölkerung, welche insbesondere aus Kindern besteht. Ungefähr ein US-Dollar steht jedem Menschen pro Tag zur Verfügung. Die Menschen leben oft in starker Armut und ca. 20% der Menschen sind mangelernährt. Der Grund dafür sind zu wenig finanzielle Mittel, wie auch mangelndes Fachwissen im Bereich Ackerbau um die Familien zu ernähren. Hinzu kommen erschwerte Anbaubedingungen für die Landwirte. Viele Flächen liegen brach, da man sie nicht mehr nutzen kann. Die Böden sind bereits ausgelaugt und benötigen mehrere Jahrzehnte bis sich dieser erholt hat und erneut bewirtschaftet werden kann. Daher kommt es immer wieder zu Engpässen bei den Ackerflächen und zu niedrigen Erträgen, da den Böden bereits alle Nährstoffe entzogen wurden. Nur Wenigen wird das Besuchen einer Schule ermöglicht, so dass viele Menschen nicht einmal Lesen und Schreiben in ihrem Leben gelernt haben. Durch fehlende Bildung, fehlt auch Erfahrung im Umgang mit modernen Geräten, wie Smartphones, Tablets etc. Eine vollständige Nutzung solcher Geräte ist den Menschen nicht möglich, da durch ihre fehlenden Kompetenzen im Bereich Alphabetismus, sie wichtige Informationen nicht aufnehmen und verarbeiten können. So ist es ihnen nicht, wie Menschen aus Industrieländern spontan möglich, eine Suchanfrage bei Google zu starten und sich über ein bestimmtes Thema zu informieren.

Außerdem haben die Menschen mit Konflikten zu kämpfen, welche ihnen das Leben erschweren. Es kommt immer wieder zu Konflikten unter den Landwirten, welche Fläche wem gehört. „Flächenklau“ ist keine Seltenheit, welcher die Landwirte belastet. Die Regierung subventioniert landwirtschaftliche Betriebe. Meist eher große Betriebe, welche Kleinbauern verdrängen da ihnen ihre Flächen genommen werden.

Abgesehen davon, sind die meisten Familienbetriebe so arm, dass sie sich nicht einmal benötigtest Equipment zulegen können. Alle Arbeit müssen per Hand erledigt werden, denn für große Maschinen welche die Arbeit erheblich erleichtern würden, fehlt das Geld. Bei

Hilfe die von außen angeboten wird ist es besonders wichtig die verfügbaren Ressourcen für den Ackerbau zu betrachten, denn Anschaffungen können kaum getätigt werden.

Für die Ackerbauern ist die Situation frustrierend. Sie geben Alles um ihre Familien ernähren zu können und dies klappt meist eher nur mit mäßigem Erfolg. Den Mut, nachhaltig und effizient zu produzieren haben die meisten bereits verloren.

### 3.3.2. User Task Organization Model

Anhand der gesammelten Daten wird schließlich das User Task Organization Model erstellt, um die Aufgaben der Primären Stakeholder zu veranschaulichen. Im Mittelpunkt der Aufgabenorganisation steht die Wissensvermittlung zwischen Helfern und Landwirten. Die Helfer müssen hierfür die Informationen zur Pflanzen aufbereiten und durch die Schulungen sie in geeigneter Form den Landwirten übermitteln. Zur Aufgaben der Helfer gehört ebenso die Überzeugung der Bauer in dem, dass die Bauer in der Lage sind, mit der Unterstützung der Helfer ihre Lebenslage nachhaltig zu ändern und Ernteerträge zu steigern. Das geplante System sollte als Schnittstelle zur Wissensübermittlung zwischen den Helfern und Landwirten agieren und ebenso die Landwirte unabhängiger von der externen Unterstützung machen, indem sie Informationen von dem System beziehen. In folgender Abbildung wird das User Task Organization Model dargestellt.

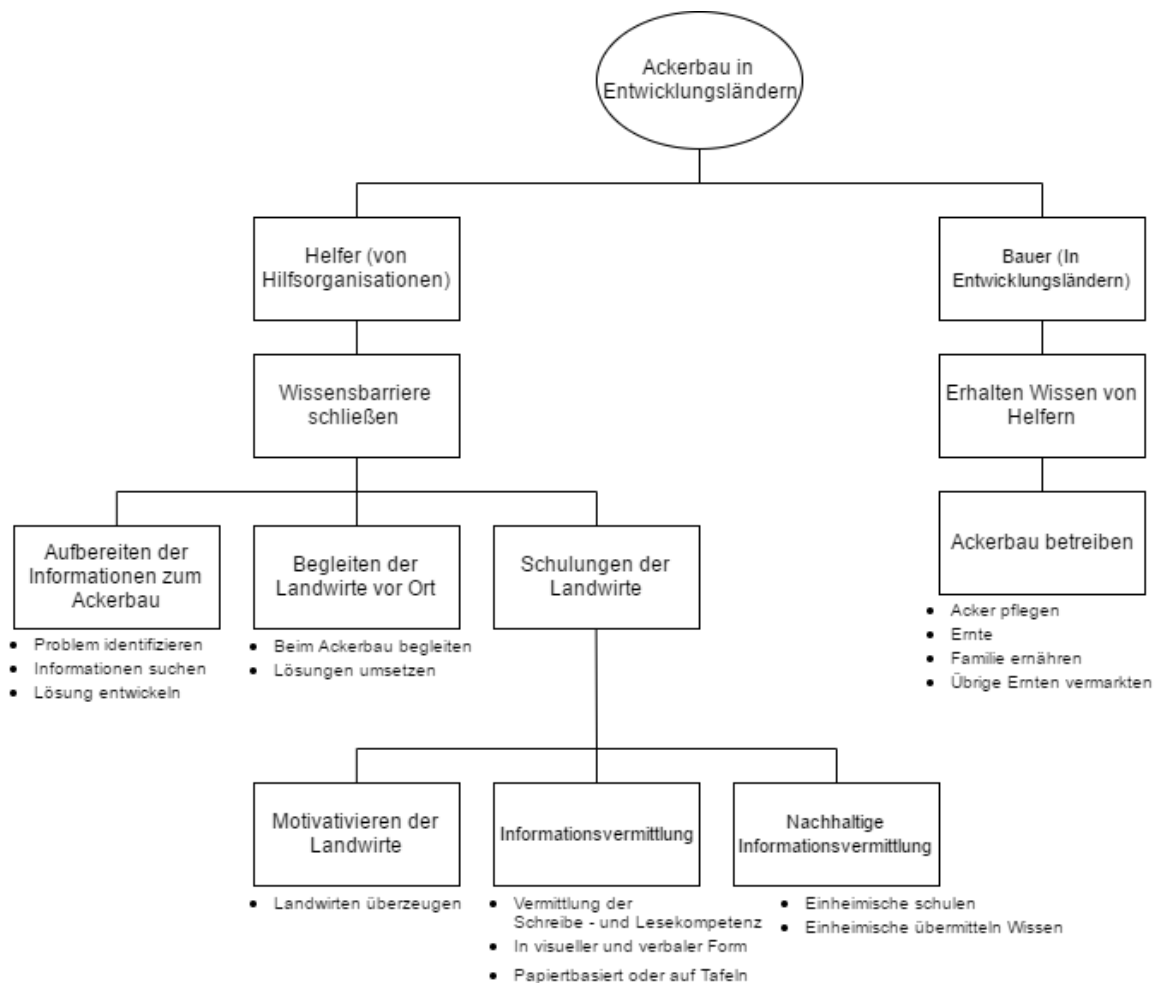


Abbildung 3 User Task Organization Model

### 3.4. Systemanforderungen

Im Folgenden werden Anforderungen anhand der in vorherigen durchgeführten Kontextanalysen an das System bzw. an das Projekt spezifiziert. Abweichend von dem Mayhew Modell werden nicht nur qualitative, sondern auch funktionale und Organisationale Anforderungen aufgelistet.

#### 3.4.1. Funktionale Anforderungen

- F10 Das System muss den Usern verschiedene Möglichkeiten zur Authentifikation bieten, damit Benutzer mit verschiedenen Kompetenzen die Zugriffsmöglichkeiten bekommen
  
- F20 Sobald der Helfer sich im System authentifiziert hat, muss das HarvestHand System ihm die Möglichkeit bieten, Daten eines Ackers einzutragen
  
- F30 Das System muss dem Helfer/Landwirt die Möglichkeit bieten jederzeit Daten von Ackerflächen zu protokollieren und zu verwalten
  
- F40 Das System muss die Möglichkeit bieten, den Landwirt oder den Helfer einem Eintrag als Collaborator zu hinzufügen, damit er die Zugriffsmöglichkeiten auf einen Eintrag bekommt
  
- F50 Das System muss Daten eines Ackers analysieren um eine passende Anbauempfehlung zu erstellen. Für die Analyse müssen relevante Bodendaten im System verfügbar sein
  
- F60 Das System muss Helfer/Landwirte ermöglichen Daten zu den Äckern jederzeit zu aktualisieren, um passendes Wissen vermitteln zu können. Diese Funktionalität muss 24 Stunden am Tag zur Verfügung stehen und fehlerfrei ablaufen
  
- F70 Das System muss Landwirten als Folge einer Bodenanalyse Anbauempfehlungen präsentieren. Dabei soll nach jeder Analyse mindestens eine Empfehlung präsentiert werden
  
- F80 Die Helfer müssen Möglichkeit haben, auf eine Anbauempfehlung zuzugreifen
  
- F90 Das System soll Landwirten Benachrichtigungen über vorhersehbare Wetterereignisse zur Verfügung stellen. Dabei sollen Warnungen vor extremen Wetterereignissen mindestens 24 Stunden vor dem Eintreffen herausgegeben werden, um Vorsichtsmaßnahmen ergreifen zu können
  
- F100 Das System muss den Helfern/Landwirten ermöglichen die Wissens- bzw. Sprachbarriere zu überwinden. Dabei soll auf textuelle Informationen möglichst verzichtet werden und Informationen über Visualisierungen, Animationen und Audio vermittelt werden
  
- F110 Das System muss den Helfern ausführliche Informationen zum Ackerbau in verschiedenen Medienformaten zur Verfügung stellen. Dabei sollte zu jedem Thema mindestens eine Visualisierung oder eine Audiodatei zur Verfügung stehen, um damit das Wissen an die Landwirte weiterzugeben

F120 Das System muss den Benutzern aktuelle und valide Daten zum Ackerbau zur Verfügung stellen. Daher sollte in regelmäßigen Abständen von ca. 1 Monat die Daten auf Aktualität überprüft werden

F130 Das System muss Helfern und Landwirten eine Pflanzendatenbank mit relevanten Informationen zur Verfügung stellen. Dabei müssen zu jeder Pflanze mindestens Angaben zur Bewässerung, Bodenbeschaffenheit, Temperatur, Lichteinstrahlung, Wachstumszeit, Erntezeit und Nährstoffbedingungen zur Verfügung stehen

F140 Das System soll fähig sein, den Landwirten ohne Smartphones die Informationen zum Ackerbau zu Verfügung zu stellen.

### 3.4.2. Organisationale Anforderungen

- Im Vorfeld müssen PoC's definiert werden, um die Realisierbarkeit des Systems zu überprüfen.
- Die Entwicklung des Systems soll nach dem Usability Engineering Lifecycle von Deborah Meyhew erfolgen. Dabei sollte jeder Teilschritt dieses Vorgehensmodells durchlaufen und dokumentiert werden.
- Das System soll während der Entwicklung evaluiert werden (s. Punkt 2).
- Ein Projektplan muss für die Umsetzung des Systems erstellt werden. Dabei sollen Aufgaben und Deadlines gesetzt werden.

### 3.4.3. Qualitative Anforderungen

- Alle funktionalen Anforderungen sollten am Ende des Projekts erfüllt sein, damit die Funktionalität des Systems nicht eingeschränkt wird und zu 100% gewährleistet werden kann.
- Das System sollte fehlerfrei und zuverlässig laufen. Bei 20 Testläufen sollte maximal einmal ein Problem auftauchen.
- Zur Erstellung eines Userprofils sollen nur minimal erforderliche Daten eines Users vom System erhoben werden. Maximal 5 Datenfelder sollen dem System übermittelt werden
- Das System soll 24/7 verfügbar sein, damit sich zu jeder Tageszeit zum Thema informiert werden kann.
- Das Nutzungsproblem muss mit Hilfe des Systems gelöst werden. Um dies überprüfen zu können, sollen User in Systemevaluation involviert werden. Dabei sollen mindestens 90% der Probanden den Inhalt des Systems erfassen können.
- Das System muss Erfordernisse/Erwartungen/Interesse der Stakeholder erfüllen. Dabei müssen vor allem alle Erfordernisse/Erwartungen/Interessen der Hauptakteure Helfer und Landwirte berücksichtigt werden. Alle Punkte dieser beiden Akteure müssen im System umgesetzt werden.
- Die Kommunikation zwischen den Komponenten muss effizient realisiert werden, die Datenübertragungszeiten sollen nicht 10 Sekunden überschreiten um eine akzeptable Reaktionszeit zu gewährleisten.
- Es müssen nur geringe Datenmengen verarbeitet werden, um eine effizienten Datenaustausch gewährleisten zu können. Nicht mehr als 100 KB pro Anfrage/Antwort sollten übertragen werden.



- Das System muss effizient die Ressourcen des Endgeräts verwenden um die Leistungsfähigkeit des Geräts nicht zu beeinträchtigen. Es soll nicht mehr als 50-100 Mb RAM verbrauchen und die Prozessorbelastung sollte 10% nicht überschreiten.
- Das System soll skalierbar sein, so dass weitere Erweiterungen effizient durchgeführt werden können.
- Das UI muss einen hohen Grad der Usability aufweisen. Diese muss durch die Evaluation des Systems gewährleistet werden. Dabei müssen mindestens 90% der User das System nachvollziehen und Inhalte verstehen können.
- Das System soll die in den Entwicklungsländern meistgesprochenen Sprachen unterstützen. Minimum Englisch, Französisch und Spanisch sollen verfügbar sein

### 3.5. Plattform Constraints

Als Client agieren im System Android Smartphones und mobile Telefone, was sehr unterschiedliche Plattformeinschränkungen ergibt. Android - Plattform und ihre Beschränkungen ist dem Entwicklerteam bereits bekannt. Im Folgenden werden wesentliche Einschränkungen der Zielplattformen aufgelistet.

#### **Android Client**

- Herstellerspezifische Ansätze. Unter Android laufen unterschiedlichsten Geräten mit verschiedenen Spezifikationen und physischen Daten, so dass die Evaluation des Systems auf allen möglichen Geräten nicht realisierbar ist.
- Unterschiedliche Android Versionen. Je älter die Versionen, desto mehr Einschränkungen gibt es
- Die Größen des Bildschirms variieren sehr stark. Aus Nutzungskontextanalyse geht hervor, dass die Benutzer des Systems zum größten Teil billigere Smartphones mit kleineren Displays besitzen. Die Displaygröße von 3,7-4 Zoll wird als Standardgröße betrachtet
- Begrenzte Akkulaufzeit
- Begrenzte Leistungsfähigkeit des Endgeräts
- In bestimmten Fällen un stabile Datenverbindung oder gar keine Verbindung
- Begrenztes Datenvolumen
- Es muss in Java programmiert werden

#### **Mobile Telefone**

Mobile Telefone sind in ihrer Funktionalität stark eingeschränkt und werden. Diese Plattform wird als Schnittstelle des Systems für die User benutzt, die kein Smartphone besitzen. Über Mobiltelefone ist die interaktive Kommunikation des Benutzers mit dem System nicht möglich. Im Folgenden werden wesentliche Einschränkungen genannt.

- Installation der App nicht möglich
- Kommunikation zwischen Server und Client nur über SMS möglich
- Die Datenübertragungsmenge der SMS – Nachrichten begrenzt
- Inhalt der SMS – Nachrichten meistens auf Text begrenzt
- Verschicken der SMS ist kostenpflichtig
- Gewisse Modelle können Farben nicht unterstützen

Zusammenfassend sollte bei der Entwicklung des Android Clients im Wesentlichen darauf geachtet werden, dass rechenaufwendige Algorithmen auf den Server überlagert werden sollen und der Datenaustausch mit dem Server effizient gestaltet werden soll. Die Mobiltelefone fordern eine geeignete Präsentation der Information, die via SMS übertragen werden kann.

### 3.6. General Design Principles

Bei der Entwicklung der General Design Principles empfiehlt Mayhew für das Produkt relevante allgemeine Literatur und Guidelines des Usability Engineerings. Da es eine Android App entwickelt werden soll, werden die Design Prinzipien von der Android Developer Webseite verwendet. Android setzt auf Material Design und beschreibt sehr detailliert die Prinzipien der Gestaltung der UI – Komponenten wie Navigation, Icons, Textes, Fonts, Animationen usw. Android Design Principles sollen helfen gebrauchstaugliche Produkte zu entwickeln und werden im Projekt eingehalten.

### 3.7. Usability Goals

Schließlich werden die Usability Goals als Bewertungskriterium spezifiziert. Die Usability Goals werden aus der Task Analysis, den Userprofiles und weiteren Erkenntnissen der Recherchen abgeleitet.

#### Qualitative Usability Goals

- Das System muss Lernprozesse des Ackerbaues fördern
- Die Landwirten sollen zum selbständigen Ackerbau motiviert werden
- Das System muss die täglichen Task der User unterstützen, ohne dass ihr Wissen überfordert wird
- Das UI muss nur für die User – Task relevante Information beinhalten

#### Quantitative Usability Goals

Alle Zeitangaben (außer Login/Registration) gelten für den Fall, dass das System schon gestartet ist und der Benutzer bereits eingeloggt ist.

#### Goal #: 1

**Task:** Registrieren im System

#### Operational Definitions

**Expert:** 3. Versuch

**Novice:** erste 3 Versuche

**Learn:** Performance ohne Fehlern

**Satisfaction:** 1-unzufrieden, 4-neutral  
7-sehr zufrieden

#### Priority Definitions:

**1:** Notwendig

**2:** Wichtig

**3:** Wünschenswert

#### Ease of Learning Goals

Priority	Measure	Goal
	Novice Time	
2	Novice Trials	3
	Novice Errors	

#### Ease of Use Goals

Priority	Measure	Goal
2	Expert Times	≤ 60 sec.
2	Expert Errors	2

#### Satisfaction Goals

Priority	Measure	Goal
2	Expert	6
2	Novice	6

**Goal #: 2****Task:** Login im System**Operational Definitions****Expert:** 3. Versuch**Novice:** erste 3 Versuche**Learn:** Performance ohne Fehlern**Satisfaction:** 1-unzufrieden, 4-neutral  
7-sehr zufrieden**Priority Definitions:****1:** Notwendig**2:** Wichtig**3:** Wünschenswert**Ease of Learning Goals**

Priority	Measure	Goal
	Novice Time	
1	Novice Trials	3
1	Novice Errors	2

**Ease of Use Goals**

Priority	Measure	Goal
1	Expert Times	≤10 sec.
1	Expert Errors	1

**Satisfaction Goals**

Priority	Measure	Goal
2	Expert	7
2	Novice	7

**Goal #: 3****Task:** Eintrag Erstellen**Operational Definitions****Expert:** 3. Versuch**Novice:** erste 3 Versuche**Learn:** Performance ohne Fehlern**Satisfaction:** 1-unzufrieden, 4-neutral  
7-sehr zufrieden**Priority Definitions:****1:** Notwendig**2:** Wichtig**3:** Wünschenswert**Ease of Learning Goals**

Priority	Measure	Goal
2	Novice Time	≤ 3 min.
	Novice Trials	
2	Novice Errors	5

**Ease of Use Goals**

Priority	Measure	Goal
2	Expert Times	≤ 2 min.
2	Expert Errors	2

**Satisfaction Goals**

Priority	Measure	Goal
2	Expert	5
2	Novice	5

**Goal #: 5****Task:** Tutorial anschauen(Helfer)**Operational Definitions****Expert:** 3. Versuch**Novice:** erste 3 Versuche**Learn:** Performance ohne Fehlern**Satisfaction:** 1-unzufrieden, 4-neutral  
7-sehr zufrieden**Priority Definitions:****1:** Notwendig**2:** Wichtig**3:** Wünschenswert**Ease of Learning Goals**

Priority	Measure	Goal
1	Novice Time	$\leq 7$ sec
	Novice Trials	
	Novice Errors	

**Ease of Use Goals**

Priority	Measure	Goal
1	Expert Times	$\leq 4$ sec.
1	Expert Errors	0

**Satisfaction Goals**

Priority	Measure	Goal
1	Expert	7
1	Novice	7

**Goal #: 6****Task:** Tutorial anschauen(Landwirt)**Operational Definitions****Expert:** 3. Versuch**Novice:** erste 3 Versuche**Learn:** Performance ohne Fehlern**Satisfaction:** 1-unzufrieden, 4-neutral  
7-sehr zufrieden**Priority Definitions:****1:** Notwendig**2:** Wichtig**3:** Wünschenswert**Ease of Learning Goals**

Priority	Measure	Goal
1	Novice Time	$\leq 5$ sec
	Novice Trials	
	Novice Errors	

**Ease of Use Goals**

Priority	Measure	Goal
1	Expert Times	$\leq 2$ sec.
1	Expert Errors	0

**Satisfaction Goals**

Priority	Measure	Goal
1	Expert	7
1	Novice	7

**Goal #: 7****Task:** Eintrag überarbeiten**Operational Definitions****Expert:** 3. Versuch**Novice:** erste 3 Versuche**Learn:** Performance ohne Fehlern**Satisfaction:** 1-unzufrieden, 4-neutral  
7-sehr zufrieden**Priority Definitions:****1:** Notwendig**2:** Wichtig**3:** Wünschenswert**Ease of Learning Goals**

Priority	Measure	Goal
2	Novice Time	≤ 3 min
	Novice Trials	
2	Novice Errors	5

**Ease of Use Goals**

Priority	Measure	Goal
2	Expert Times	≤ 1 min
2	Expert Errors	1

**Satisfaction Goals**

Priority	Measure	Goal
1	Expert	5
1	Novice	6

**3.8. Fazit**

Die Requirements Analysis ergab wichtige Erkenntnisse für das System. Die wichtigsten Stakeholder des Systems sind Helfer und Landwirte. Dennoch besonders die sehr vielfältigen Merkmale der Landwirte machen die Arbeitsreorganisation sehr kompliziert. Hier müssen variierende Schreibe – und Lesekompetenzen der Landwirte beachtet werden. Entsprechend den Kompetenzen der Stakeholder muss auch die Präsentation der Systeminformationen angepasst werden. Die Analphabeten Landwirte haben andere Anforderungen und brauchen dementsprechend andere Präsentation als die Alphabeten Landwirte und die Helfer. Des Weiteren variieren die Kenntnisse im Bereich der IT – Systeme von „Vorhanden“ bis „gar keine“. Als Konsequenz können gegeben falls die sich in der IT – Welt schon längst etablierten Interaktionsparadigmen für Landwirte nicht verständlich sein. Es ist also notwendig, für die gleichen Informationen mindestens zwei Präsentationen zu erstellen, damit die verschiedenen Stakeholder das System entsprechend eigenen Erwartungen benutzen können. Diese Erkenntnisse sind ausschlaggebend für die weitere Modellierung des Systems.

## 4. Work Reengineering

Als erster Schritt vor der eigentlichen Entwicklung und Evaluation wird die Reorganisation der deskriptiven Aufgabenmodellierung durchgeführt. In diesem Schritt wird das Reengineered Task Organisation Model erstellt.

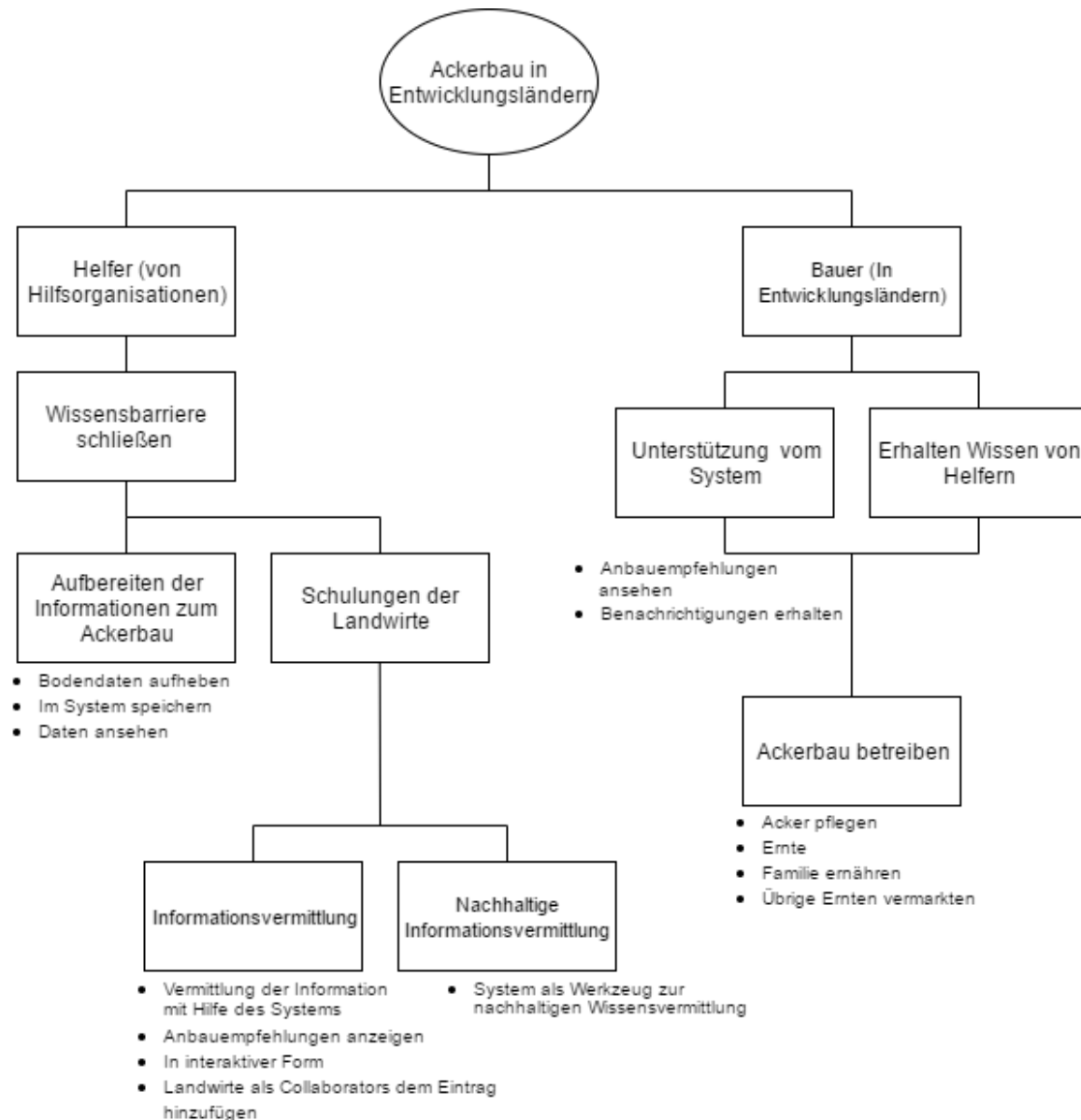


Abbildung 4 Reengineered Task Organization Model

Reorganisation der Useraufgaben sollte in der ersten Linie höherer Performance und Automatisierung bestimmter Arbeitsprozesse führen. In diesem Sinne wird das System die Informationssammlung und die Informationsvermittlung Prozesse vereinfachen und automatisieren. Zum einen soll das System die Helfer bei der Identifikation des Problems und einer Lösungsentwicklung durch die automatische Bodenanalyse unterstützen und als Medium zur Wissensübermittlung dienen. Zum anderen bekommen die Landwirte die Möglichkeit, die Information direkt vom System zu beziehen und werden dadurch autonomer.

## 5. Conceptual Model Design

Conceptual Model Designs ist erster Schritt der Design und Testing Prozesse. Hier werden die grundlegenden Regeln für high– Level UI Entwickelt und später als Grundlage für weitere Designentscheidungen verwendet. Conceptual Model Design muss in vorherigen Kapiteln durchgeführte Analysen referenzieren.

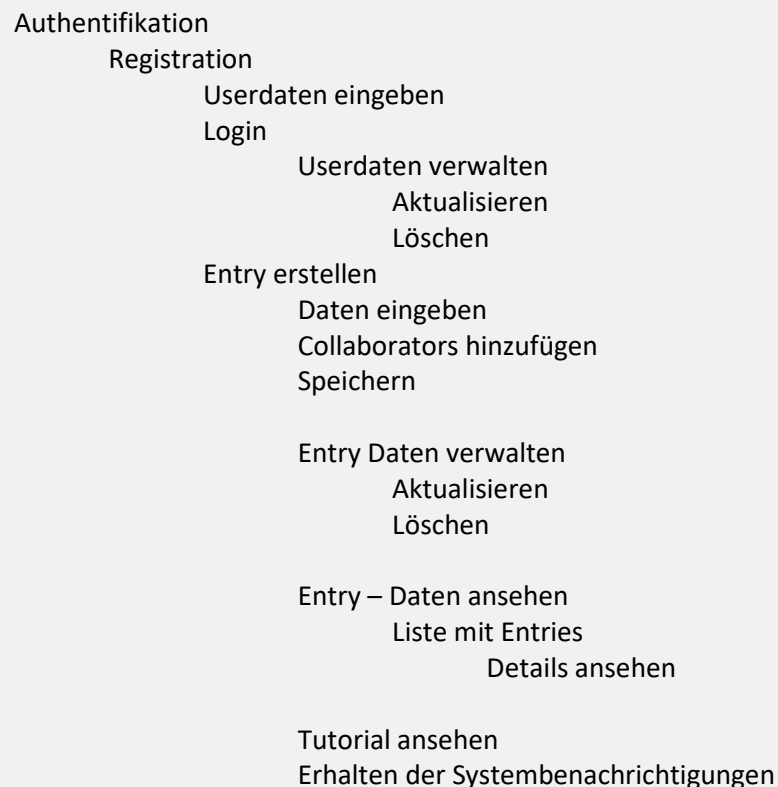
### Processes – oriented

In dem System werden von den Usern keine Produkte erstellt oder verwaltet, sondern bestimmte Prozesse zur Erledigung der Aufgaben durchgeführt. Die Prozesse sind von dem Task Organization Model abgeleitet

### Prozesse

Zu grundlegenden Prozessen im System gehören die Authentifikation der User, Erstellen der Entries, Darstellung der Entry – Daten und Darstellung der Tutorials. Hier werden die Prozesse hierarchisch Dargestellt.

System zur Unterstützung der Ackerbauer in Entwicklungsländern



Aus dem oberen hierarchischen Modell wird ersichtlich, dass die User nach der Authentifikation im System ein Eintrag mit den Ackerdaten erstellen können und danach zu dem die Infos und Tutorials ansehen können. Des Weiteren ist die Möglichkeit gegeben, die Userdaten und Entrydaten zu verwalten

## Design presentation rules for processes

Da es im Projekt Material Design Prinzipien von Android eingehalten werden, wird zu Gestaltung der einzelnen Prozesse Android Styleguide verwendet. Die High – Level und Sub – level Prozesse werden in einzelnen Windows (Activity) dargestellt, Navigation zwischen denen durch Buttons und Pfeilen erfolgen soll. Nach der erfolgreichen Registration und dem Login wird Activity mit der Liste von Entries angezeigt, auf die der bestimmte Benutzer Zugriff hat. Falls der User im System eingeloggt ist, wird bei Start des Systems direkt die Liste mit allen Entries angezeigt, von wo man zu den anderen Activities navigieren kann.

## Design rules for windows

Android Material Design Dokumentation umfasst sehr detaillierte Spezifikationen und best practices der UI Gestaltung, welche auch eingehalten werden. Die Dokumentation beschreibt vor allem die Gestaltungsprinzipien der Activities und vielen weiteren UI – Elementen wie Buttons, Dialogfenstern, Inputfeldern usw. Auf formale Auflistung wird hier verzichtet, Details können in der Android [Dokumentation](#) angesehen werden.

## Navigation

Wie schon erwähnt, erfolgt die Navigation durch das System mit Hilfe von Android Buttons. Nach der Registration/Login gelangt man in die Root – Activity. Von dem Root – Activity, wo die Liste von Entries angezeigt wird, navigiert man durch den Click auf einen Listeneintrag zur der Detail – View eines Entry. Da werden ausführliche Infos zu dem Entry präsentiert. Von dort navigiert der Button **tutorial** zu der Activity, die ein interaktives Tutorial präsentiert.

Root – Activity verfügt über ein Floating Action Button zum Öffnen eines Formulars, wo die Daten eines Entry eingetragen und gespeichert werden können. Nach dem Speichern navigiert das System automatisch in die Root – Activity. Zum Bearbeiten des Userprofiles klick man auf den Imagebutton in der Actionbar der Root – Activity. Die Zurück – Navigation erfolgt mit Hilfe von Zurück – Pfeil in der Actionbar, der immer zu einer für jede Activity definierten Parent – Activity führt.

Im Folgenden Kapitel werden die Conceptual Model Mock – ups erstellt, um die oben beschriebenen Designkonzepte zu verdeutlichen.



### 5.1. Conceptual Mock – ups

Conceptual Mock – ups stellen wichtige Prozesse des System grafisch dar. In diesem Projekt wird auf grafische Tools zur Erstellung der Mockups aus zeitlichen Gründen verzichtet. Stattdessen werden die Mockups auf einem Stück Papier gezeichnet.

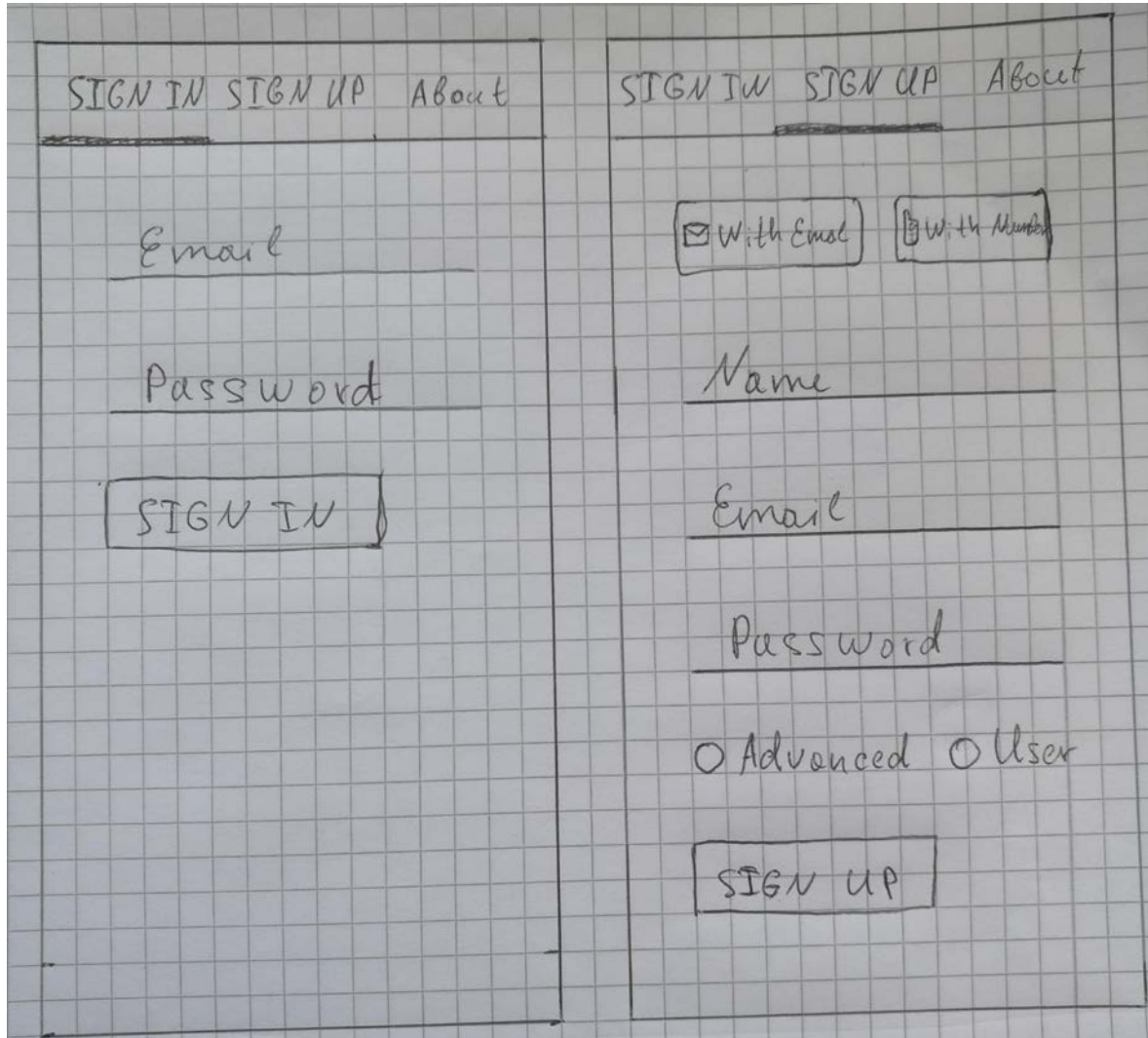


Abbildung 5 Sign up und Sign in

Registration/Login werden mit **Tabs** präsentiert, was die Navigation viel effizienter macht. **Sign Up** Tab ermöglicht verschiedene Registrationsmethoden – die klassische mit Email für erfahrene User und die Registration nur mit Mobiltelefonnummer für diejenigen, die kein Smartphone besitzen oder keine Emailadresse haben. Die Formulare können mit jeweiligen Buttons umgeschaltet werden.

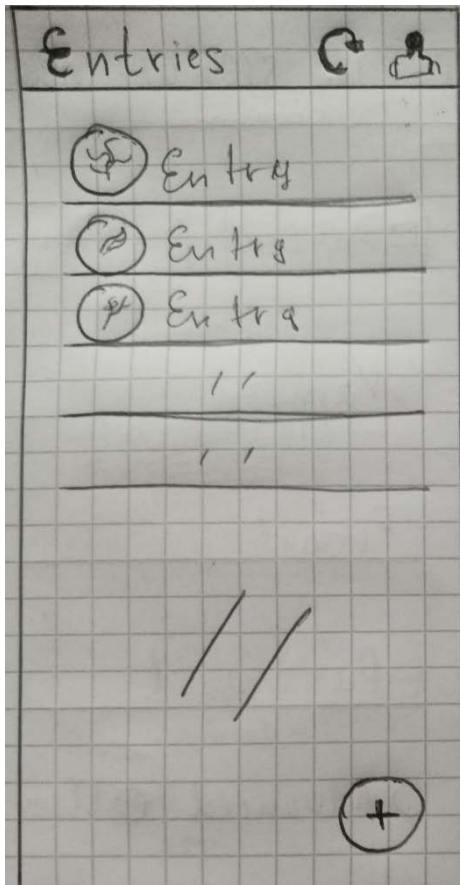


Abbildung 6 Liste mit Entries

Alle Entries, auf die der bestimmte Benutzer Zugriff hat, werden in einer Liste dargestellt. Mit dem Click auf Floating Action Buttons öffnet sich andere Activity mit dem Formular (Siehe Abbildung 6) zum Erstellen eines neuen Eintrags. Dieser Button ist nur im Modus für erfahrene Benutzer sichtbar. Die Listen Einträge sollen nur kurze Kerninformationen eines Eintrags präsentieren.

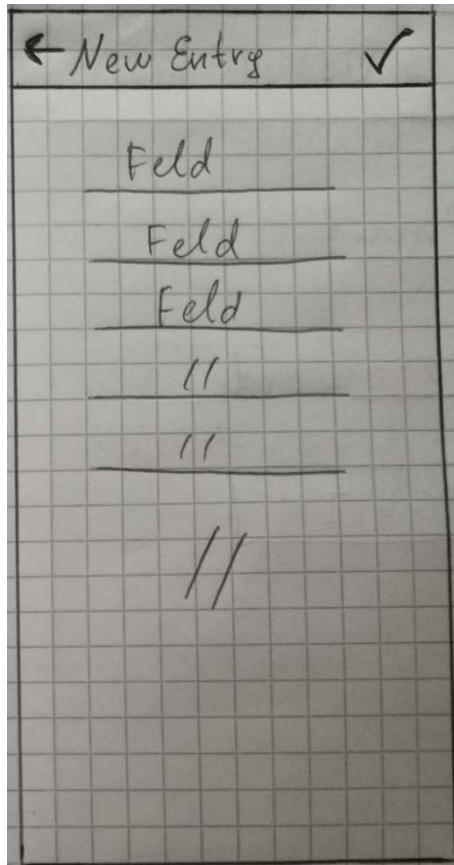


Abbildung 7 Neuen Eintrag erstellen

In dieser Activity werden Daten eines Ackers Eingetragen und im System gespeichert. Es werden mehrere Textfelder und Dropdown – Menus im Fenster zur Eingabe benutzt. Prozess der Dateneingaben kann nur von den Alphabeten erfolgen. Dieses Fenster ist daher nur für die Alphabeten User erreichbar.

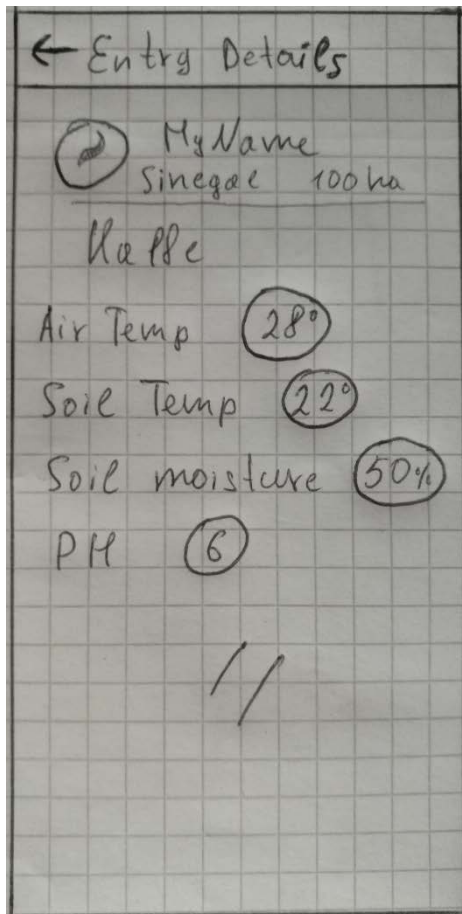


Abbildung 8 Entry Details

In diesem Fenster werden Details zu einem Entry dargestellt. Hier sollen alle einzelnen Eigenschaften eines Eintrags präsentiert werden. Die passende Visualisierung ist an dieser Stelle besonders wichtig. Die Details müssen sowohl für Alphabeten als auch für Analphabeten verfügbar sein. Das heißt, es müssen zwei verschiedenen Präsentationen der Information erstellt werden. Abbildung 7 ist die Präsentation für Alphabeten und wird ebenfalls visuelles Feedback zu dem Zustand des Ackers enthalten, abhängig von den ermittelten Daten.

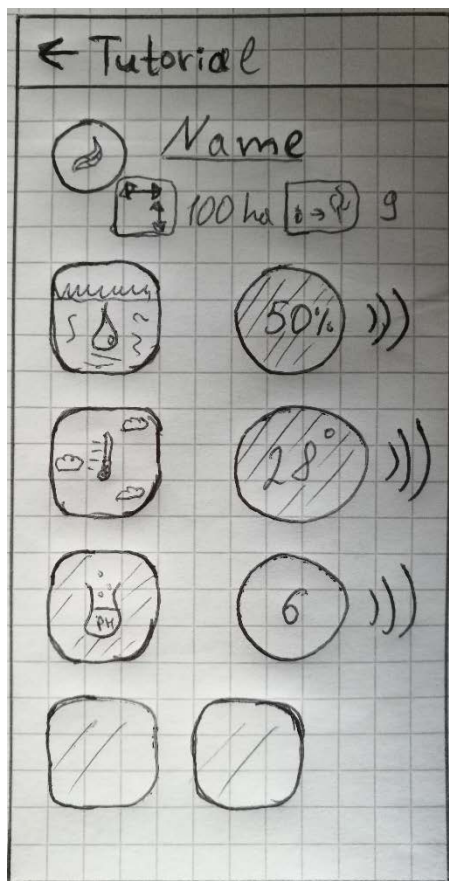


Abbildung 9 Tutorial zu Einem Eintrag

Die Präsentation der Ackerinformationen für die Analphabeten wird direkt mit den Anbauempfehlungen kombiniert und in einem Fenster dargestellt, damit der Benutzer weniger Zwischenschritte machen muss, um die Tutorials anschauen zu können. Hier ist besonders darauf zu achten, dass wo möglich die Information nicht in textueller Form, sondern in visueller oder Verbaler Form dargestellt wird. Um ein Objekt der realen Welt zu präsentieren, müssen möglichst genau beschreibende Abbildungen gewählt werden. Da wo es auf textuelle Informationen nicht verzichtet werden kann, müssen die Texte vertont werden. Die Mischung aus verbalen und visuellen Informationspräsentationen soll ein für Analphabeten gebrauchstaugliches UI schaffen. Diese Ansicht ist für Beide Profiltypen verfügbar. Auf diese Weise können sowohl Helfer auf die Tutorials zugreifen um die Schulungen durchzuführen, als auch die Landwirte selbst.

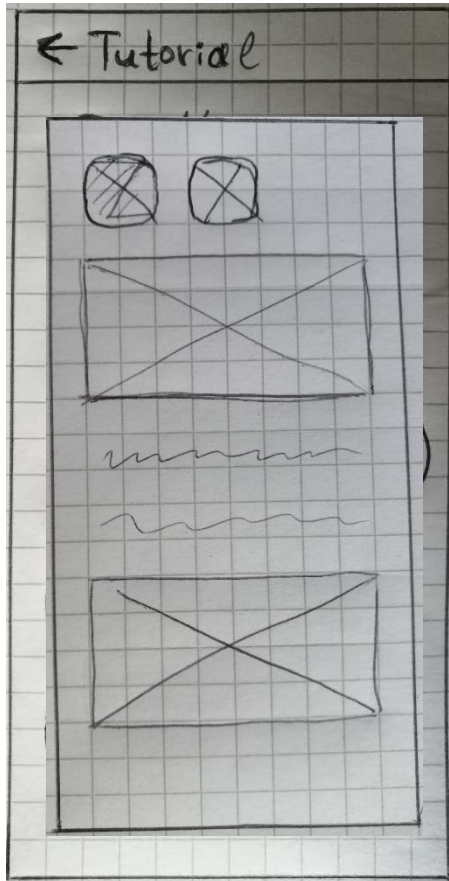



Abbildung 10 Ausführliche Anbauerklärung

Um die ausführlichen Informationen zum Ackerbau anzuzeigen öffnet sich bei Klick auf ein bestimmtes Icon über dem Tutorial – Activity ein neues Fenster. Z. B bei Klick auf Icon  werden ausführliche Anbauempfehlungen zu Verbesserung der Bodenfeuchtigkeit in weiteren Bildern, Animationen, und vertonten Texten dargestellt.

## 5.2. Evaluation

Für kleinere Projekte empfiehlt Mayhew nur die abschließende Evaluation des Projekts nach dem Level 3. Für dieses Projekt wird ebenfalls auf die Iterative Evaluation in jedem Level verzichtet. In der ersten Linie wird dies aus zeitlichen Gründen gemacht. Des Weiteren sind für die Evaluation detaillierte Bilder und Animationen notwendig, um die Gebrauchstauglichkeit des Systems besonders für Analphabeten bewerten zu können. Diese liegen im aktuellen Stadium des Projekts nicht vor. Im Rahmen des Projekts wird nur eine abschließende Evaluation nach dem Level 3 durchgeführt.

## 6. Screen Design Standards

---

In diesem Kapitel werden die Screen Design Standards festgelegt, um ein konsistentes UI zu schaffen und ease of learning und ease of use zu unterstützen. Zu Erstellung der Standards werden als Grundlage die Style Guides von Android verwendet.

### 6.1. Control Standarts

Die Steuerungselemente werden in der folgenden Tabelle aufgeführt. Diese Elemente ermöglichen die Kontrolle über alle Interaktionsprozesse im System.

Aktion	Control
Navigation	Buttons, Tabs, anklickbare Listenelemente, Image – Buttons
Eins auswählen	Checkbox
Eingabe	Textfeld, Dropdown-Menu(spinner)
Liste	Dropdown-Menu

### 6.2. Process window standarts

Die Einzelnen Activitys, in denen Prozesse ausgeführt werden, sollen ebenso einheitlich gestalten werden um die Konsistenz des UI gewährleisten zu können. Dazu schlägt Mayhew die Process window standarts festzulegen, nach denen die einzelnen Fenster designt werden sollen. Dazu werden ebenso in Android etablierten Prinzipien verwendet. Im Folgenden werden wesentliche Punkte genannt.

#### Actionbar

- Die Actionbar sollte in jedem Prozess – Fenster implementiert werden und dem User darüber Auskunft geben, wo er sich grade befindet
- Die Actionbar soll Navigations – Icons haben oder Buttons, die die Funktionalität eines Fensters erweitern
- In jedem Fenster außer dem Root – Fenster soll ein Back – Pfeil enthalten sein, um die Navigation zum Parent – Fenster zu ermöglichen

#### Activity

- Activities (Fenster) außer dem Login – Activity müssen Light – Gray – Background haben
- Activities mit vielen Inputelementen oder Informationen müssen scrollbar sein
- Aktionen wie Save/Delete müssen in der Actionbar mit entsprechenden Icons dargestellt werden
- Activities, die Usereingaben fordern, müssen benötigte Inputfields, Checkboxes und Dropdownmenüs haben
- Listenelemente sollen in Card – Form präsentiert werden, Jedes Element enthält ein Bild, Namen, Location und Fläche
- Details – Activity sollen alle verfügbare Daten Visualisieren
- Tutorials – Activity werden überwiegend mit Interaktiven Bildern und Audio dargestellt

- Activities, die in bestimmten Fällen keinen Inhalt haben, sollen den Empty – State deutlich visualisieren.
- Möglichst alle UI – Elemente der Screens sollen anklickbar sein, derer Inhalt soll vertont werden.

### 6.3. Weitere Standarts

#### Dialog box standards

Android Dialogs werden verwenden, um bestimmte Aktion des Users zu bestätigen. Z. B. Delete/Update/Discard Aktionen erfordern eine bestätigen der User, um zufällige Ausführung dieser Aktionen zu vermeiden.

#### Message box standards

Android Toast – und Snackbarmessages geben dem User zusätzliche Informationen oder informieren ihn über Background Prozesse.

#### Input device interaction standards

Das System wird mit Touch – Gesten und falls vorhanden den Hardware – Buttons gesteuert.

#### Feedback Standards

Die Ladeprozesse des Systems werden mit Android Progress – Indikatoren visualisiert. Der Status der ausgeführten Prozesse wird Toast – und Snackbarmessages dem User angezeigt. Die Snackbarmessages sollen zudem den Status farblich präsentieren. Erfolgreich ausgeführte Prozesse solle in grüner Farbe dargestellt werden, die Fehler in roter Farbe.

Nicht ausgefüllte Inputfelder müssen rot markiert werden. Zudem sollte es auf den Fehler genau hingewiesen werden. Z. B wenn der eingegebene Passwort zu kurz, soll dies explizit in der Fehlermeldung angegeben werden. Zu Kontrolle der Inputfelder werden ebenso Android – Tools verwendet.

Weitere Usereingaben wie Checkboxes und Dropdown – Menus müssen falls nötig ebenfalls kontrolliert werden und im Fehlerfall müssen die Benutzer explizit über Toast – und Snackbarmessages informiert werden.

In der folgenden Tabelle werden wichtige Designeigenschaften des Systems aufgeführt, die eingehalten werden müssen.

*Tabelle 2 Style Guide*

Element	Eigenschaft	Anmerkung
Colors		
Primary	#212121	Farbe der ActionBar
Primary Dark	#000000	
Accent Color	#47974b	
Hint Color	#969696	Wird in InputFields benutzt
Error Color	#fd5656	Farbe des Textes der Errormessages



Successful Color	#ff3c9737	Farbe des Textes der Errormessages zur erfolgreichen Aktionen
Main Background Color	#c4c4c4	Backgroundcolor der Activities
Push Button		
Color	Accent Color	Farbe der normale Push Buttons
Border	none	
Radius	none	
Shadow	noen	
Cards		
Fill	white	
Margin	6dp	
Radius	2dp	
Elevation	2dp	
Tutorial Icons		
Color	multicolored	Farbige Bilder
Radius	4dp	
Border	2dp	

## 6.4. SDS Prototyping

Da die Screen Design Standards von Android exakt übernommen werden, wäre das Prototyping überflüssig. Die Darstellung einzelner Elemente wird von den [Android Design Guidelines](#) nicht abweichen. Dieser Schritt wird daher mit Detailed User Interface Design kombiniert. Die Style Guides werden im Rahmen des Projekts ebenso nicht extra erstellt, da das ganze Vorgehen in dieser Dokumentation beschrieben wird.

## 7. Detailed UI Design

---

In diesem Kapitel werde die einzelnen Activities und UI – Elemente dargestellt und erläutert. Vorwiegen werden bei der detaillierten Erstellung des UI Android Style Guides angehalten. Solche Eigenschaften wie Iconsgröße, Textgröße, Abstände zwischen Elementen werden exakt übernommen, das diese schon längst die best practices sind. Auch die Icons, die Routinen Aufgaben wie Löschen, Aktualisieren, Überarbeiten oder Speichern, werden von [Material Design Icons](#) verwendet. Im Grunde genommen müssen nur die Activities individuell gestaltet werden, die für Analphabeten verfügbar sind. Dies bedeutet, dass die Informationen nicht nur mit Bildern präsentiert werden müssen, sondern es spezielle Icons erstellt werden müssen, die sowohl für Analphabeten als auch für die User mit eingeschränkten Kenntnissen im Umgang mit interaktiven IT – Systemen verständlich sind. Die Material Icons sind dagegen sehr schlicht und monochrom gestaltet, so dass sie gegebenenfalls nur für erfahrene Benutzer verständlich sind. Im Folgenden werden die Lösungen präsentiert.

Abbildung 10 präsentiert das Login – Fenster. Das Fenster ist in den im oberen Kapitel definierten Farben gestaltet. Zwischen **SIGN IN** und **SIGN UP** Fenstern erfolgt mit Tabs, was die Navigation sehr effizient macht. Über den Tabs ist eine Progressbar sichtbar, die die Ladeprozesse des Systems visualisiert und signalisiert dem User, dass das System nicht hängt, sondern eine interne Funktion ausführt. Die Progressbar muss in allen Activities implementiert werden, in denen zeitintensive Funktionen ausgeführt werden. **SIGN UP** ist dem entsprechend ähnlich aufgebaut. Unterschied ist nur in dem, dass im **SIGN UP** Fragment zusätzlich noch die Checkboxes zur Eingabe benutzt werden, um den Profiltyp zu ermitteln. **SIGN UP**

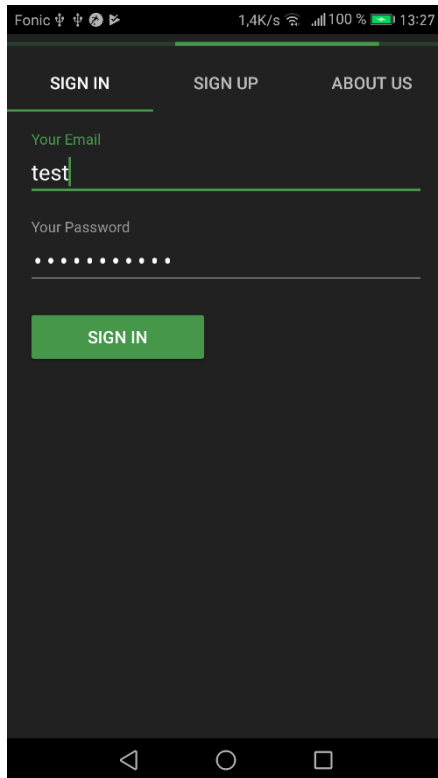


Abbildung 11 Sign In Fragment

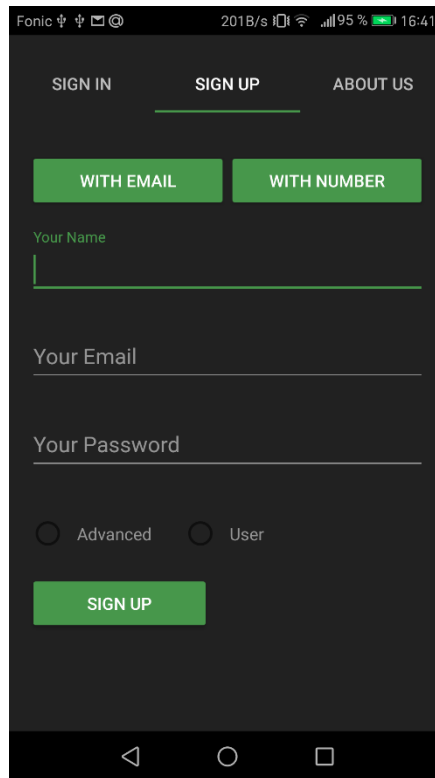


Abbildung 12 Sign up Fragment



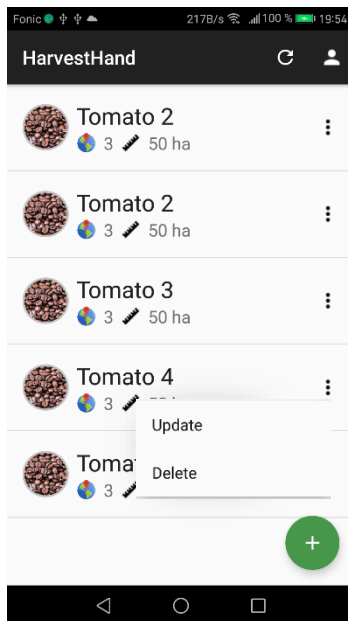


Abbildung 13 Liste mit Entries

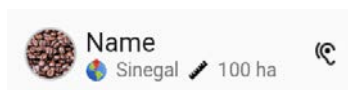


Abbildung 14 Item - Präsentation für Analphabeten

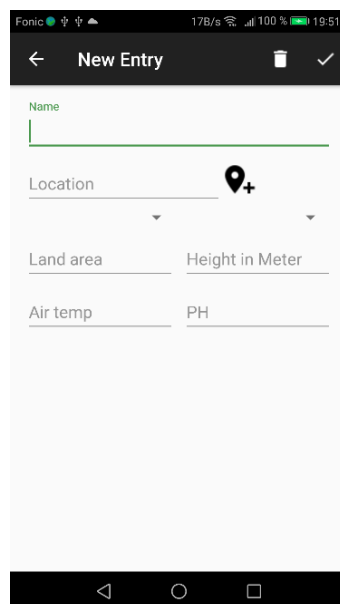


Abbildung 15 Erstellung eines Entry's



Abbildung 16 Dialogbox

Die Entry's werden in einer Liste präsentiert. Ein Listeneintrag enthält kurze Übersicht über ein Entry und ein Bild, das die Bestimmte Pflanze darstellt. Im Beispiel Links sind Kaffee – Acker mit einer entsprechenden Abbildung dargestellt. Der Plus Floating Action Button navigiert zum Formular zur Erstellung eines neuen Eintrags. Wichtig ist hier, dass die Informationen entsprechende Visualisierungen haben. Die Übersicht Informationen müssen auch für Analphabeten verständlich sein. Daher wird es auf die Material Icons verzichtet. Stattdessen müssen möglichst aussagekräftigere Abbildungen verwendet werden. Aus Zeitgründen konnten die Abbildungen nicht erstellt werden. In Abbildung 12 sind mögliche Lösungen vorgestellt. Jeder Eintrag hat ein weiteres Popup – Menü (3 vertikale Punkte rechts) mit dem man den Jeweiligen Eintrag löschen oder bearbeiten kann. Dieses Menü sowohl das Plus – Button sind nur für Alphabeten verfügbar, das es da viele textuelle Inputs benötigt werden.

Für die Analphabeten muss ein Eintrag etwas anders aussehen. Hier wird ein Icon zum Anhören des ganzen Inhaltes eines Eintrags angebracht. Da es auf die textuellen Informationen vollständig nicht verzichtet werden kann, muss diese Funktion überall da verfügbar sein, wo Textelemente vorhanden sind.

Hier ist ein Formular abgebildet, das zum Erstellen eines Neuen Eintrags verwendet wird. Diese Funktionen ist nur für User mit vorhandenen Les – und Schreibkompetenzen verfügbar. Hier werden Inputfelder mit Dropdownmenüs und Icons Kombiniert. Alle Elemente werden nach Android Style Guides gestaltet, da es hier keine besondere Präsentation nötig ist. Die ActionBar hat ein Navigationspfeil, mit dem man immer zu Parent – Activity navigieren. Des Weiteren sollen sich immer die Aktionsbuttons wie Save/Delete in diesem Fall in ActionBar befinden, damit diese Aktionen schnell erreicht werden können, ohne zu scrollen. Wenn das Fenster verlassen wird, bevor gemachte Änderungen gespeichert wurden, muss eine Dialogbox angezeigt werden, um die Aktion zu bestätigen. Löschvorgänge müssen ebenfalls bestätigt werden. Dies gilt für alle Fenster, wo ähnliche Vorgänge ausgeführt werden können.

Dialogboxen werden nach Andorid Style Guides gestaltet.

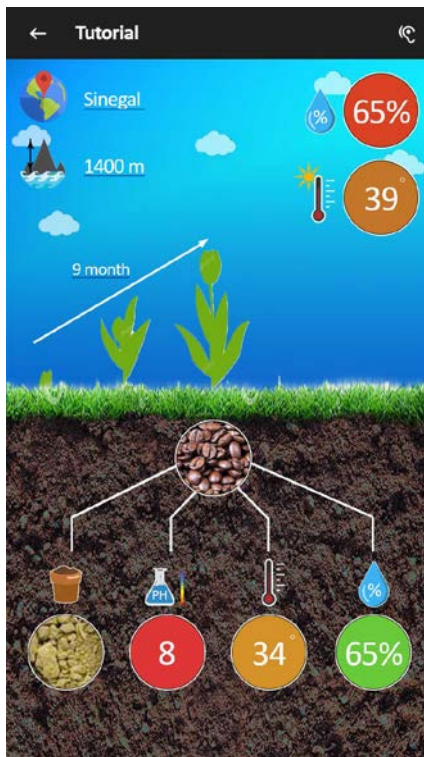


Abbildung 17 Tutorial Activity

Wie schon in vorherigen Kapitel erwähnt, können die Prinzipien von Android Material Design für die Analphabeten nicht funktionieren. Daher wurde ein eigens Konzept der Visualisierung für die Analphabeten erarbeitet. Die in Kapitel 5.1 dargestellten Tutorial Mock – ups wurden komplett überarbeitet. Bei diesem Konzept wird auf die Abbildung der realen Objekte der Welt gesetzt. Die Informationen und Objekte werden möglichst mit beschreibenden Bildern präsentiert. Die untergestrichenen Texte werden beim Anklicken vertont. Es sind in Prinzip alle Elemente interaktiv. Die einzelnen Werte wie PH – Wert oder Lufttemperatur werden in Farbigen Kreisen dargestellt. Die Hintergrundfarbe eines Kreises gibt Feedback über den Zustand bestimmter Eigenschaften. Grün ist dabei „gut“, gelb „befriedigend“ und rot „schlecht“. Beim Anklicken einer der Eigenschaft öffnet sich ein Popup – Fenster mit einer ausführlichen Erklärung und Anbauempfehlung. Dies wurde von dem Tutorial Mock – up in Kapitel 5.1 übernommen.

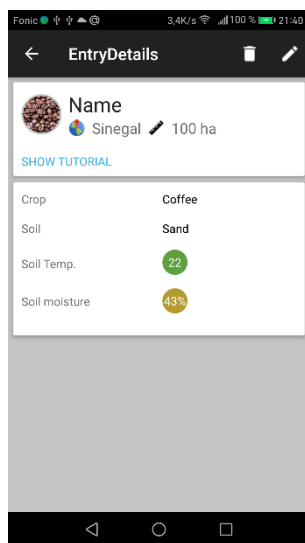


Abbildung 18 Entry Details

Die in der Abbildung 17 dargestellte Ansicht ist nur für die Alphabeten verfügbar, hier werden die Informationen klassisch dargestellt. Zu der Ansicht in der oberen Abbildung kann man aus diesem Activity ebenfalls navigieren. Dies ermöglicht z. B den Helfern den Zugriff auf die Tutorials, damit die diese bei schulengen vor Ort präsentieren können. Die Liste der Elemente in der Abbildung ist nicht vollständig. Da der Inhalt dieser Activity dynamisch ist, wurden die Elemente nur beispielhaft aufgeführt.

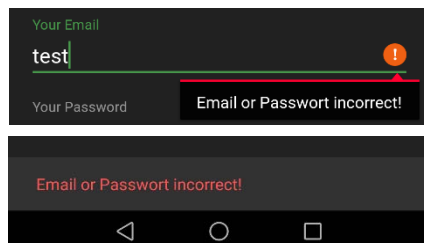


Abbildung 19 Fehlerbehandlung

In der Abbildung 17 ist die Fehlerbehandlung dargestellt. Hierfür werden Android Standardtools verwendet

## 7.1. Fazit

Die größte Herausforderung bei der detaillierten Konzipierung des UIs stellt die Anpassung für die Analphabeten. Hier muss der Kompromiss zwischen der textuellen, visuellen und verbalen Präsentationen gefunden werden. Zudem müssen sehr viele Abbildungen und Icons erstellt werden, um die Information angemessen darzustellen.

Um das UI für Alphabeten und Analphabeten zu optimieren, werden schließlich zwei Modi der Darstellung implementiert, die sich grundsätzlich in dem unterscheiden, dass der Analphabeten – Modus keinen Zugriff auf die Verwaltung der Entries hat. Delete/Update/Edit/Add Aktionen sind in dem Modus nicht verfügbar. Des Weiteren fehlt in dem Modus die klassische Darstellung der Entry – Details Informationen. Stattdessen gelangt man direkt zu der Interaktiven Ansicht aus Abbildung 16. Der Alphabeten – Modus hat Zugriffsmöglichkeiten sowohl auf alle Verwaltungsfunktionen, als auch auf die Interaktive Ansicht.

## 8. Evaluation

---

Im Folgenden wird die Durchführung der Evaluation des Detailed User Interfaces Design dokumentiert. Die Evaluation wird nach Mayhew erfolgen. Im Rahmen der Evaluation sollen sowohl ease – of – learning, als auch ease – of – use Aspekte des Systems getestet werden.

Für die Helfer ist die Effizienz der Arbeiterledigung im Fokus. Da die Interaktionsparadigmen schon bekannt sind, ist für die Helfer ease – of – use Aspekt wichtiger. Für die Analphabeten und die User mit eingeschränkten Kenntnissen im Umgang mit interaktiven Systemen steht ease – of – learning im Fokus, da diese Stakeholdergruppe sich erstmals mit Interaktionen des Systems vertraut machen muss. Des Weiteren muss Inhalt der Abbildungen verinnerlicht werden. Aus diesem Grund werden beide Aspekte getestet.

Da es für das Entwicklung Team nicht möglich ist, die direkten Stakeholder des Systems in die Evaluation zu involvieren, werden Personen für die Tests rekrutiert, die die ähnlichen Eigenschaften aus den definierten Userprofiles haben. Die Evaluation erfolgt in der gewöhnlichen Umgebung der Benutzer, in diesem Fall in einem Wohnzimmer.

### 8.1. Ergebnisse der Evaluation

Zu der quantitativen Einschätzung der Durchgeführten Tests wurden die in Kapitel 3.7 definierten quantitativen Usability Goals verwendet. Die Zeiten, die die Bestimmten User für jeweilige Tasks gebraucht haben, wurden aufgezeichnet und dokumentiert. Näher dazu im Anhang.

Grundsätzlich lieferte die Evaluation akzeptable Ergebnisse. Die Zahlen in den Usability Goals konnten in den Tests erreicht werden. Dafür war das UI so organisiert, dass die Kernfunktionalitäten des Systems in wenigen Schritten erreichbar sind.

Auch die Visualisierungen des Systems wurden von den involvierten Benutzern im Schnitt gut erkannt. Bei Thinking-Aloud-Test konnten die User die meisten Icons in Tutorial – Activity problemlos verstehen und erklären.

Das aller erste Problem wurde bei der Registration im System ermittelt. Hier wurde die Tab Navigation nicht direkt erkannt, die User versuchten sich direkt im **SIGN IN** Fenster einzuloggen und bekamen dementsprechend eine Fehlermeldung „*Incorrect Email or Password*“, was sie irritierte. Einer der Gründe dafür war, dass die Tab Überschriften sich stark ähneln und die Unterschiede nicht wahrgenommen werden. Als Lösung hierfür wird **SIGN IN** in **LOG IN** umbenannt. Des Weiteren soll im Login – Fenster genaue darauf hingewiesen werden, dass der Registrationsformular sich in anderem Fenster befindet.

Für die Analphabeten ist die Registration im System ohne weiteren Hilfe klar nicht möglich. Um aber diesen Vorgang für die Analphabeten zu optimieren, fehlen die Visuellen Informationen zu den Aktionen wie z.B Icon in den Eingabefeldern, die etwas darüber sagen würden, welche Eingabe benötigt werden. Es wird ebenfalls überlegt, die Authentifikation nur mit Hilfe der Telefonnummer durchzuführen und somit diesen Vorgang noch effizienter zu gestalten.

Einige Icons sollen überarbeitet werden, da die nicht eindeutig interpretiert werden konnten. So ein Problem wurde bei dem **Bodentemperatur – Icon** und **Fläche – Icon** festgestellt.

Weiterer kritischer Punkt war die Sprach – Funktion der App. Es wurde nicht erkannt, dass die Texte beim Anklicken vorgelesen werden könnten. Dies muss dringen korrigiert werden. Auch das **Ear – Icon** war nicht ganz eindeutig und konnte nur von den erfahrenen Benutzern als Vorlesefunktion erkannt werden.

Die schlechtesten werte lieferte erwartet die Evaluationsiteration mit dem User 4. Dies war aber ein Extremfall, da die Person weder Englischkenntnisse, noch Kenntnisse im Umgang mit Computern und mobilen Geräten hatte. Dieser Test war aber sehr wichtig, denn hier konnte überprüft werden, ob die Darstellung der Anbauempfehlungen für Analphabeten und auch Benutzer mit geringen Kenntnissen im Bereich der Computertechnik nachvollziehbar ist.

## 8.2. Fazit

Die Evaluation lieferte sehr wichtige Erkenntnisse für den Designprozess. Es wurden viele Probleme erkannt, von denen einige für das Entwicklerteam erstmals überhaupt nicht vorstellbar waren. Auch positive Aspekte wurden ermittelt und dokumentiert. Wie erwartet haben die Informationsvisualisierungen und Sprach – Funktionen die meisten kritischen Punkte aufgewiesen. Einige User haben die gleichen Tests doppelt durchgeführt, sodass bei der zweiten Iteration die Elemente wesentlich schneller wiedererkannt wurden. Das ist eine positive Erkenntnis für den ease – of – learning Aspekt. Auch die Zeiten für die Navigation zum bestimmten Fenster verbesserten sich. Zur Ausführlichen Info siehe Anhang 12.3. In den folgenden Entwicklungsschritten ist es Aufgabe des Teams, die festgestellten mangeln zu verbessern und umzusetzen.

## 9. Systemarchitektur

Im Folgenden wird die Systemarchitektur genauer erläutert. Das im Konzeptplan vorgestellte Modell wurde überarbeitet um die Unterschiede zwischen der möglichen Clients deutlicher zu machen.

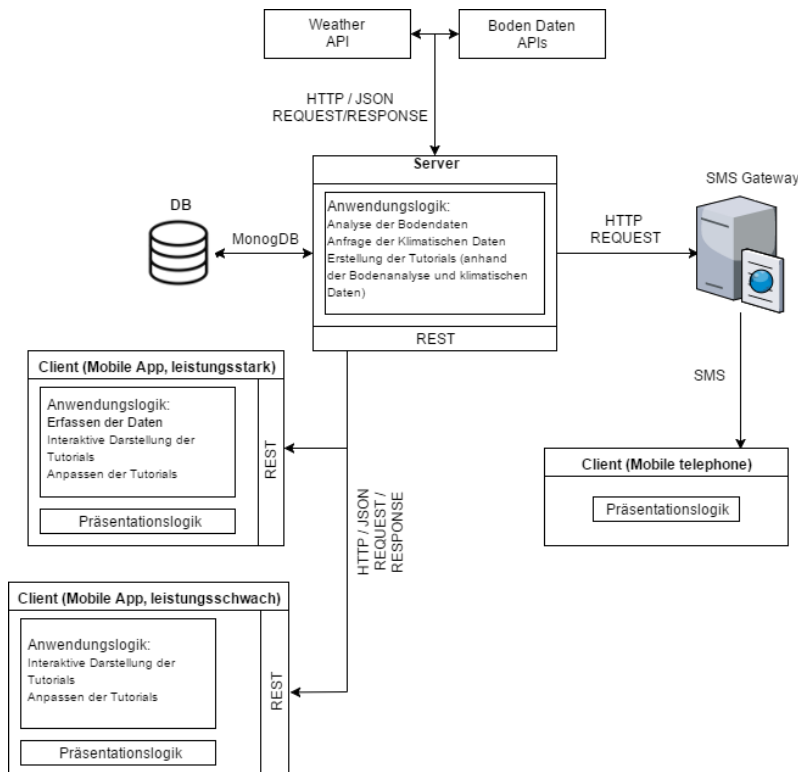


Abb. 1 Systemarchitekturmodell (Überarbeitet)

Bei der Entwicklung des Systems muss besonders beachtet werden, dass die Benutzer die Smartphones mit sehr unterschiedlichen Leistungsfähigkeiten besitzen können. Die Helfer benutzen eher moderne und leistungsstärkere Smartphones oder Tablettts. Die Bauer dagegen haben schwächere und kleinere Geräte mit älteren Android Versionen. Diese Erkenntnis aus dem Nutzungskontext ist besonders wichtig und die Verteilung des Systems muss entsprechend realisiert werden.

### 9.1. Ressourcen

Hier wird API zum Zugriff auf Systemressourcen definiert. Im Folgenden Kapitel werden die Ressourcen genauer beschrieben.

URI	Verb	Semantik	Contenttype(req)	Contenttype(res)	Statuscodes
/signup	POST	Registration der Benutzer	JSON	JSON	200 – Registration erfolgreich 500 – interner DB Fehler 409 – der User mit der eingegebenen Email existiert bereits
/signin	POST	Login des Benutzers	JSON	JSON	200 – Registration erfolgreich

					500 – interner DB Fehler
					401 – Login Daten falsch eingegeben
/users/:id	GET	Bestimmte Userdaten werden abgerufen	Parameter	JSON	200 – Daten werden abgerufen
					204 – Kein User mit der id gefunden
					500 – interner DB Fehler
/users/:id	PUT	Bestimmte Userdaten aktualisieren	Parameter, JSON	JSON	200 – Daten aktualisiert
					204 – Kein User mit der id gefunden
					500 – interner DB Fehler
/users/:id /entries	GET	Vom User Erstellte Entries werden abgerufen	Queryparameter	JSON	200 – Daten abgerufen
					204 – Kein Entry gefunden
					500 – interner Fehler
/entries	POST	Entry wird erstellt	JSON	JSON	200 – Entry erfolgreich erstellt
					500 – interner Fehler
/entries/:id	PUT	Entry wird aktualisiert	Parameter, JSON	JSON	200 – Entry aktualisiert
					204 – Kein Entry gefunden
					500 – interner Fehler
/entries/:id	GET	Daten eines Entries werden abgerufen	Parameter	JSON	200 – Daten abgerufen
					204 – Kein Entry gefunden
					500 – interner Fehler
/entries/:id	DELETE	Entry wird gelöscht	Parameter	JSON	200 – Entry gelöscht
					204 – Kein Entry gefunden
					500 – interner Fehler
/entries/:id/tutorial/:id	GET	Tutorial zu einem Entry wird abgerufen	Parameter	JSON	200 – Tutorial abgerufen
					204 – Kein Tutorial gefunden
					500 – interner Fehler
/norms	POST	Eine Norm wird erstellt	JSON	JSON	200 – Norm erstellt
					500 – interner Fehler
/norms/:id	PUT	Eine Norm aktualisieren	JSON	JSON	200 – Norm aktualisiert
					204 – Keine Norm gefunden
					500 – interner Fehler
/norms/:id	DELETE	Bestimmte Norm löschen	Parameter	JSON	200 – Norm gelöscht

					204 – Keine Norm gefunden
					500 – interner Fehler
/norms	GET	Norms nach Queryparametern abrufen	Queryparametern	JSON	200 – Daten abgerufen
					204 – Keine Norms gefunden
					500 – interner Fehler

## 9.2. Datenstrukturen

Alle Entitäten des Systems sowie Daten der externen Dienste werden im JSON – Format repräsentiert. Zu den internen Daten des Systems gehören Benutzerdaten (user), Daten eines Ackers(entry) und die dynamisch erstellten Anbauempfehlungen (tutorial). Des Weiteren werden standardisierte Daten zur den verschiedenen Pflanzenarten (norm) in der Datenbank des Systems gespeichert. Zur externen Daten gehören die Wetterdaten und bestimmte Bodendaten.

### Repräsentation eines Users

Zur Informationen eines User gehören nur für die Authentifikation benötigte Angaben.

`user_type` wird verwendet um die Präsentation der Informationen an Analphabeten anzupassen. `TYPE_USER` identifiziert einen Analphabeten und Präsentation in der App wird entsprechend.

```
var userSchema = mongoose.Schema({
  name: {type: String, required: true},
  email: {type: String, unique: true},
  pass: {type: String},
  user_type: {type: Number, required: true} /*TYPE_PROFI = 1, TYPE_USER = 0*/
  phone_number:{type: Number, unique: true }
});
```

## Repräsentation eines Entry

Entry – Struktur wird zur Speicherung bestimmter Ackerdaten verwendet.

```
var entrySchema = mongoose.Schema({
  /*Ein beschreibender Name*/
  entry_name: {type: String, required: true},
  /*Art der Pflanze, durch id identifiziert*/
  art_id: {type: Number, required: true},
  /*Verknüpfter Tutorial*/
  tutorial_id: String,
  /*ID des Erstellers*/
  owner_id: {type: String, required: true},
  /*Colaborators, die den Zugriff auf den Eintrag haben*/
  collaborators:[String],

  /*
   * Properties
   */

  location: String,
  /*Fläche in km²*/
  area: Number,
  /*Art des Bodens*/
  soil: Number,
  air_temp: Number,
  soil_temp: Number,
  soil_moisture: Number,
  ph_value: Number,
  /*Anbauhöhe über dem Meerspiegel*/
  height_meter: Number
});
```



## Repräsentation einer Norm

```
var normsSchema = mongoose.Schema({
  crop_id: Number,
  name: [String],
  family: String,
  air_temp: {
    max: Number,
    min: Number
  },
  /*Wasserbedarf 1/m^2/Woche*/
  water_requirements: Number,

  fertilizer: {
    name: String,
    id: Number
  },
  soil: {
    name: String,
    id: Number
  },
  soil_temp: {
    max: Number,
    min: Number
  },
  soil_moisture: {
    max: Number,
    min: Number
  },
  ph_value: Number,
  height_meter: {
    max: Number,
    min: Number
  },
  mature_after_month: Number
  /*.
  */
});
```

Eine Norm speichert standardwerte einer bestimmten Pflanze und wird zur Bodenanalyse verwendet. Und die Abweichungen festzustellen werden die Daten eines existierenden Eintrags mit den Standardwerten abgeglichen. Norm wird von Entwicklungsteam mit Hilfe von Recherchen erstellt, da es kein passendes API mit Pflanzendaten gefunden war. Diese Entität beinhaltet für den Anbau einer Pflanze relevanten Daten und kann im Prozess der Entwicklung verändert werden.

## Repräsentation eines Tutorials

```
var tutorialSchema = mongoose.Schema({
  air_temp: {
    status: Number,
    deviation: Number,
    norm: Number
  },
  air_moisture: {
    status: Number,
    deviation: Number,
    norm: Number
  },
  /*Wasserbedarf 1/m^2/Woche*/
  soil_moisture: {
    status: Number,
    deviation: Number,
    water_requirements: Number,
    norm: Number
  },
  soil: {
    status: Number,
    norm: Number
  },
  soil_temp: {
    status: Number,
    deviation: Number,
    norm: Number
  },
  ph_value: {
    status: Number,
    deviation: Number,
    norm: Number
  },
  height_meter: {
    status: Number,
    deviation: Number,
    norm: Number
  },
  mature_after_month: Number
  /*
   *
  */
});
```

Ein Tutorial speichert Daten einer Bodenanalyse und wird mit dem jeweiligen Entry verknüpft. Tutorial speichert aktuellen Status(Wert ist größer, niedriger oder im akzeptierbaren bereich), Abweichungen von Normen in Prozent und den gewünschten Wert jeder Eigenschaft des Entry. Der Client verwendet Tutorial, um die interaktive Präsentation zu erstellen.

## Externe Wetterdaten

Als Wetterdienst wird Weather Underground verwendet (wunderground.com). Unten ist eine beispielhafte JSON – Response. Die benötigte Daten wie Temperatur, Höhe müssen aus dem Response herausgefiltert werden. Die Wetterdaten werden vom Server nach Location abgefragt und falls nötig gefiltert an den Client gepusht. Der Client sollte nur dann benachrichtigt werden, wenn ein Wetterereignis den Pflanzenanbau beeinflussen kann. Dadurch werden die Anbauempfehlungen flexibler gestaltet und können an die aktuellen Wetterbedingungen angepasst werden.

```
{...
"local_tz_short": "PDT",
"local_tz_long": "America/Los_Angeles",
"local_tz_offset": "-0700",
"weather": "Partly Cloudy",
"temperature_string": "66.3 F (19.1 C)",
"temp_f": 66.3,
"temp_c": 19.1,
"relative_humidity": "65%",
"wind_string": "From the NNW at 22.0 MPH Gusting to 28.0 MPH",
"wind_dir": "NNW",
"wind_degrees": 346,
"wind_mph": 22.0,
...}
```

## SMS Getaway

Zum Verschicken von SMS wird ein POST – Request an websms REST API geschickt und von dort an die Telefonnummer. Dies ist eine optionale Funktion des Systems und wird nur dann implementierte, wenn Ressourcen dazu vorhanden sein werden. Besonders wichtig ist hierbei, dass die Nachrichten in einer binären Form verschickt werden müssen, damit die Mobiltelefone sie verarbeiten können. Für die Anbauempfehlungen bedeutet es, dass sie möglichst kompakt und nur in Form von Texten und einfachen gegebenenfalls schwarzweißen Bildern gestaltet werden müssen. Die SMS sind zudem weniger interaktiv und sind für das Projekt nicht relevant. Weiter wird Request Body Beispiel vorgestellt (Quelle: <https://websms.de/entwickler/apis/rest-sms-api>). Details können in der API – Doku nachgelesen werden.

```
{
"userDataHeaderPresent" : false,
"messageContent" : [ "...", ... ],
"test" : false,
"recipientAddressList" : [ ..., ... ],
"senderAddress" : "...",
"senderAddressType" : "national",
"sendAsFlashSms" : false,
"notificationCallbackUrl" : "...",
"clientMessageId" : "...",
"priority" : ...,
}
```

## Externe Bodendaten

Zur Abfrage der Bodenfeuchtigkeit und Bodentemperatur in bestimmter Location können APIs von ClearAg (<https://www.clearag.com/>) und Department of Agriculture and Food ([link zur seite](#)) verwendet werden. ClearAg liefert ausführliche Daten zur Bodenfeuchtigkeit.

Department of Agriculture and Food stellt Daten zur Bodentemperatur zu Verfügung. Diese Information können vom System abgefragt werden, falls die Daten von dem Benutzer vor Ort nicht erhoben werden können.

### 9.3. Anwendungslogik des Systems

Die Kernfunktionalität des Systems besteht aus Bodenanalyse auf dem Server und der interaktiven Präsentation der Anbauempfehlungen auf dem Client. Diese Funktionalitäten werden im Folgenden grob skizziert.

#### 9.3.1. Bodenanalyse

Die Bodenanalyse wird durchgeführt, um die Abweichungen der Eigenschaften einer Pflanze von Standardwerten festzustellen und somit die Anbauempfehlungen individueller erstellen zu können. Ergebnisse der Analyse werden in Tutorial – Entität gespeichert.

```
var NORM = 0, //flag, wenn daten von der Norm nicht abweichen
    LESS = 1, //flag, wenn daten kleiner als die Norm
    GREATER = 2, //flag, wenn daten größer als die Norm
status = 0;

function analyseData(entry) {
    /*Hol Norm aus der Datenbank zur jeweiligen Pflanze*/
    var externalRequest = http.request(){
        var norm = data;
    }

    request("URL zum Hollen der Niederschlagsdaten für eine Woche"){
        var wochenNiederschlag = response;
    }

    /*Füre anylyse für alle Eigenschaften*/
    forEach(entry.eigenschaft){
        /*Vergleiche Entry.daten mit Norm.daten*/
        if(entry.soil_moisture <= norm.soil_moisture.min){
            /*Berechne abweichung in %*/
            var deviation = entry.soil_moisture ist Prozent von norm.soil_moisture.min;
            /*Berechne benötigte Wassermenge für eine Woche*/
            var water_requirements = norm.water_requirements * entry.area - wochenNiederschlag;
            status = LESS;
        }else if (entry.soil_moisture >= norm.soil_moisture.max){
            /*Berechne abweichung in %*/
            var deviation = norm.soil_moisture.max ist Prozent von entry.soil_moisture;
            status = GREATER;
        }else{
            /*Berechne abweichung in %*/
            var deviation = entry.soil_moisture ist Prozent von norm.soil_moisture.min;
            status = NORM;
        }
    }
    /*
    .
    .
    */
}
```

```

if(entry.air_temp <= norm.air_temp.min){
    /*Berechne abweichung in %*/
    var deviation = entry.air_temp ist Prozent von norm.air_temp.min;
    status = LESS;
}else if (entry.soil_moisture >= norm.air_temp.max) {
    /*Berechne abweichung in %*/
    var deviation = norm.air_temp.max ist Prozent von entry.air_temp;
    status = GREATER;
}
else{
    /*Berechne abweichung in %*/
    var deviation = entry.air_temp ist Prozent von air_temp.min;
    status = NORM;
}
/*
 *
 */
}

function saveTutorialInDb(alldata);
}

```

### 9.3.2. Interaktive Tutorials

Der Client präsentiert die vom Server erstellte Tutorials und allgemein die Entry – Daten. Anhand der Tutorial – Daten wird visuelles Feedback zur dem Zustand der Pflanze gegeben. Abhängig von dem Status der jeweiligen Eigenschaft wird eine Anbauempfehlung für Analphabeten erstellt. Hierfür werden passende Bilder aus dem Gerätspeicher geholt und zusammengestellt, um eine passende Empfehlung zur Verbesserung bestimmter Eigenschaft wie z.B PH –Wert oder Bodenfeuchtigkeit zu erstellen. Des Weiteren ist die verbale Präsentation der Bildschirm Inhalte die Aufgabe des Clients.

```

public void requestTutorialFromServer(){
    /*Setze Backgroundfarbe für alle Icons*/
    for(eigenschaft in tutorial){
        switch(deviation){
            case eigenschaft.deviation <= 10%:
                icon.setImageResource(Grün);
                break;
            case eigenschaft.deviation <= 20%:
                icon.setImageResource(Gelb);
                break;
            case eigenschaft.deviation <= 30%:
                icon.setImageResource(Orange);
                break;
            case eigenschaft.deviation > 30%:
                icon.setImageResource(Rot);
                break;
        }
    }
}

```

```

if(eigenschaft.status = NORM){
    /*Zeige eine standardisierte Anbauempfehlung zur dieser Eigenschaft*/
    bilder = readFiles(bilder.norm);
    /*Erstelle empfehlung aus Bildern, Animationen etc.*/
    empfehlung.setImageResource(bilder);
} else if (eigenschaft.status = LESS)
    /*Zeige eine Anbauempfehlung zur verbesserung der Eigenschaft*/
    bilder = readFiles(bilder.less);
    /*Erstelle empfehlung aus Bildern, Animationen etc.*/
    empfehlung.setImageResource(bilder);
} else (eigenschaft.status = GRAETER){
    /*Zeige eine Anbauempfehlung zur verbesserung der Eigenschaft*/
    bilder = readFiles(bilder.greater);
}

```

## 9.4. Datenschutz

Das System verarbeitet nur wenige private Daten der Benutzer wie Email und Telefonnummer. Diese können in der Datenbank verschlüsselt gespeichert werden. Weiterer Austausch der privaten Daten findet im System nicht statt, so dass keine besonderen Maßnahmen der Datensicherheit nötig sind.

## 10. Fazit

---

Insgesamt mussten für das identifizierte Nutzungsproblem im ersten Schritt sehr viel an Informationen gesammelt werden. Außerdem war es nicht einfach, sich in die Situation der in Entwicklungsländern lebenden Menschen hinein zu versetzen bzw. Informationen über ihre Arbeitsabläufe zu recherchieren. Dort wurden viele wichtige Punkte durch unseren Betreuer angestoßen, welche dann weiter ausgearbeitet wurden.

Nach vielen Recherchen konnten die Ziele des Systems eindeutig identifiziert werden. Ein Problem stellte oft die Messbarkeit der Ziele dar. Nach einigen Tipps, konnten die Ziele eindeutiger beschrieben werden und erhielten messbare Werte.

Im nächsten Schritt legten wir Proof of Concepts an. Diese sollen helfen die Realisierbarkeit des Systems zu überprüfen. Nachdem wir diese genau spezifiziert haben, wurden sie auch nach und nach implementiert. Die wichtigsten Schritte, wurden auch in der Dokumentation berücksichtigt.

Anschließend erstellten wir nach der Stakeholderanalyse, Userprofiles. In dem Fall für Landwirte und Helfer. Dabei wurden Daten nach bestimmten Kriterien erhoben. Daraufhin wurden Personae im Bezug auf die identifizierten Userprofiles erstellt. Sie stellen mögliche zukünftige Nutzer des Systems dar. Nachdem die Nutzer des Systems genauer unter die Lupe genommen wurden, wurde die Contextual Task Analysis durchgeführt. Dabei wurden Bedürfnisse und Wünsche der Benutzer an das System ermittelt. Dabei war es dem Team allerdings nicht möglich, direkt in den Alltag der Benutzer einzusteigen. Daher basiert die hier durchgeführte Contextual Task Analysis auf Recherchen. Auch das physikalisch/soziale Umfeld der Menschen wurde genau betrachtet, um diese Informationen im Entwicklungsprozess zu berücksichtigen. Mit Hilfe der gesammelten Daten wurde ein User Task Organization Model erstellt. Dieses veranschaulicht die Aufgaben der primären Stakeholder, Helfer und Landwirt.

Im weiteren Verlauf wurden Systemanforderungen festgelegt. Diese sind untergliedert in funktionale, qualitative und organisationale Anforderungen. Diese wurden so weit es geht auch mit messbaren Werten ausgestattet.

Außerdem wurden Plattformeinschränkungen festgelegt.

Bei den General Design Principles werden die Design Prinzipien von der Android Developer Webseite verwendet. Android Design Principles sollen helfen gebrauchstaugliche Produkte zu entwickeln und werden im Projekt eingehalten.

Anschließend wurden die Usability Goals als Bewertungskriterium festgelegt. Diese ergeben sich aus den vorherigen Arbeitsschritten.

Als erster Schritt vor der eigentlichen Entwicklung und Evaluation wurde die Reorganisation der deskriptiven Aufgabenmodellierung durchgeführt. In diesem Schritt wurde das Reengineered Task Organisation Model erstellt. Im nächsten Schritt wurden Mockups auf Papier gezeichnet, welche die wichtigen Prozesse des Systems grafisch darstellen.

Nun wurden die Screen Design Standards festgelegt. Zu Erstellung der Standards werden als Grundlage die Style Guides von Android verwendet. Alle vorherigen Prozesse wurden nun auch evaluiert. Im gesamten Entwicklungsprozess wurde nach dem Usability Engineering Lifecycle von Deborah J. Mayhew nachgegangen.

Es wurde sich eine Systemarchitektur überlegt. Dabei musste besonders berücksichtigt werden, dass es große Unterschiede in der Leistungsfähigkeit der Endgeräte der Benutzer geben wird. Alle Datenstrukturen wurden nochmal genauestens erläutert.

Es wäre denkbar, dass dieses System zukünftig weiter ausgearbeitet werden kann um den Menschen in Entwicklungsländern Fachwissen im Bereich Ackerbau zu vermitteln. Dafür muss jedoch eine Hilfsorganisation gefunden werden, welche diese Aktion unterstützen möchte. Das System zu erweitern wird kein Problem darstellen und ist leicht umsetzbar. Wir würden uns für die Zukunft wünschen, dass so etwas in der Art den Menschen zur Verfügung gestellt wird, um ihr Wissen zu erweitern und ihr Leben nachhaltig zu verbessern.

Insgesamt ist die Entwicklung des Systems sehr gut abgelaufen. Aufgaben waren klar verteilt und es wurde immer Zeit gefunden die Ergebnisse zu besprechen und gemeinsam auszubessern. So konnten alle Ideen diskutiert und berücksichtigt werden. Es gab einzelne Themen bei denen es zu Verständnisproblemen kam. Jedoch konnten alle Fragen mit Hilfe des Betreuers geklärt werden.

## 11. Installationsdokumentation

---

### 11.1. Systemanforderungen

Zum ausführen des Systems werden Nodejs und MongoDB unter Windows 7-10 und Internetverbindung gebraucht. Für den Client ist mindestens Android 4.0 notwendig. Server und Client müssen sich in gleichem Netz befinden. Im Folgenden werden benötigte Schritte detailliert beschrieben

### 11.2. Server

- Nodejs herunterladen und installieren
- Benötigte Packages mit dem npm – Befehl über console installieren: *npm -i*
- MongoDB herunterladen und installieren
- MongoDB zur umgebungsvariablen des Systems hinzufügen
- Datenbank mit dem Konsolenbefehl starten: *mongod -dbpath db*
- Server in der Konsole starten: *node server*

### 11.3. Client

- HarvestHand.apk ausführen und installieren
- IP Des Servers ermitteln und in der App eintragen
- App benutzen



## 12. Anhang

---

### 12.1. POCs (überarbeitet)

#### 12.1.1. Benutzer anlegen

**Beschreibung:** Helfer und Bauer müssen jeweils Profile erstellen können, um Zugriffsrechte zu erhalten. Dabei werden verschiedenen Daten der Benutzer als JSON – Objekt an Server geschickt und gespeichert. Es soll einfache Authentifikation mit Email und Passwort oder nur mit Handynummer erfolgen. Zwei Profilmodi mit verschiedenen Zugriffsrechten für Alphabeten und Analphabeten sollen verfügbar sein.

**Exit:** Jeder User kann abhängig von eigene Kompetenzen Profiltyp auswählen und sich registrieren. Nach Eingabe erforderlicher Daten wird der User in Datenbank angelegt.

**Fail:** Der Benutzer hat keine Email und keine Handynummer und kann sich somit nicht registrieren. Dem Benutzer fehlen Kenntnisse im Umgang mit interaktiven Systemen.

**Fallback:** Der Helfer soll die Bauer bei Erstellung eines Profils unterstützen. Außerdem kann beim Start der App eine Anleitung zum Erstellen von Profilen vorgeführt werden. Falls der Benutzer keine Möglichkeit zur Registration im System hat, sollen alternative Authentifikationsmethoden eingesetzt werden.

#### 12.1.2. Collaborators dem Eintrag hinzufügen

**Beschreibung:** Der User muss einem Eintrag als Collaborator hinzugefügt werden, damit er die Ackerdaten ansehen kann und dazu spezialisierte Empfehlungen erhält. Der Benutzer wird nach der Email oder Handynummer gesucht, sodass immer nur ein genauer Benutzer gefunden wird.

**Exit:** Profil des gesuchten Benutzers wird in der Datenbank gefunden und dem Eintrag hinzugefügt. Die Datenbanksuche und die Datenübertragungszeit dürfen nicht länger als 5-10 Sekunden dauern.

**Fail:** Das Benutzerprofil wird nicht in der Datenbankgefunden oder die Suche überschreitet die zugelassene Suchzeit.

**Fallback:** Das Profil des Benutzers muss neu angelegt werden

#### 12.1.3. Effiziente Datenübertragung

**Beschreibung:** Die Daten Übertragung zwischen Systemkomponenten muss effizient realisiert werden, so dass im JSON – Format nur für Verarbeitung relevante Daten übertragen werden, und keine Bilder, Audio oder Video Dateien. Die Struktur der JSON – Daten sollte möglichst einfach gehalten werden und primitive Datentypen enthalten.

**Exit:** Es wird geringer Datenvolumen zwischen Komponenten transferiert. Die Datenmengen überschreiten nicht 100 KB pro eine Übertragung, Übertragungszeiten überschreiten nicht 10 Sekunden. Die Daten zur Visualisierung der Informationen werden lokal auf dem Client – Gerät gespeichert und abgerufen.

**Fail:** Die Kommunikation zwischen Systemkomponenten verbraucht mehr als 100 KB Datenvolumen pro Anfrage und dauert länger als 10 Sekunden. Datenübertragung wird stakt beeinträchtigt oder abgebrochen.

**Fallback:** Schlechte Kommunikationsinfrastruktur in den betroffenen Ländern muss bei der Entwicklung der Systemarchitektur berücksichtigt werden. Längere Datenübertragungszeiten sollen zugelassen werden, im Fail – Fall soll die Anfrage erneut gesendet werden

#### 12.1.4. Auswertung der Bodendaten

**Beschreibung:** Die aufgehobenen Daten werden im JSON an den Server geschickt, da analysiert und darauf aufbauend werden Tutorials zum Ackerbau für Benutzer erstellt. Dazu sollen auch Informationen zur aktuellen Wetterbedingungen bezogen werden. Für den Analyse – Algorithmus müssen bestimmte Bodendaten wie Bodenfeuchtigkeit, Lufttemperatur, PH – Wert etc. zur Verfügung stehen. Die rechenaufwändigen Algorithmen werden dadurch nicht auf dem Client realisiert und beeinträchtigen nicht die Performance des Endgeräts.

**Exit:** Bei der Eingabe der Daten wird auf die Vollständigkeit zugewiesen. Die Daten sind für die Verarbeitung vollständig, werden na Server geschickt, ausgewertet und zur weiteren Verarbeitung an den Client geschickt.

**Fail:** Der Algorithmus arbeitet fehlerhaft oder die Daten zur Analyse sind nicht vollständig, so dass nicht valide Empfehlungen für Benutzer erstellt werden.

**Fallback:** Dem Benutzer wird eine standardisierte nicht individuell angepasste Anleitung angezeigt.

#### 12.1.5. Effiziente Erstellung interaktiver Tutorials

**Beschreibung:** Es Werden dynamisch personalisierte Anbau - Tutorials auf dem Client in interaktiver form dargestellt. Dazu verwendete Bild – Audio oder – Videodateien sollen lokal in möglichst komprimierter Form vorliegen, so dass die Performance und Akkuleistung des Endgeräts nicht beeinträchtigt wird.

**Exit:** Die Anleitungen werden auf Basis der vom Server Erhaltenen Daten erstellt und interaktiv dargestellt. Dabei sollte nicht mehr als 50-100 MB RAM und nicht mehr als 10% Prozessorleistung pro Session verwendet werden.

**Fail:** Der Algorithmus arbeitet fehlerhaft oder die lokalen Daten werden zu lange aus dem Speicher geladen oder können nicht gelesen werden. Oder es ist nicht ausreichend RAM verfügbar und die Applikation wird geschlossen. Dadurch wird der Client stakt belastet und die Anleitungen werden nicht angezeigt.

**Fallback:** Eine in textueller Form oder nur in Bildern gestaltete Anleitung anzeigen. Auf die interaktive Visualisierung, Videos und Animationen wird verzichtet.

#### 12.1.6. Klimatische Daten der Externen Dienste

**Beschreibung:** Zur Erstellung der genaueren Empfehlungen werden klimatische Daten von externen Diensten gebraucht. Abhängig von aktuellen Wetterbedingungen werden die Empfehlungen ständig aktualisiert. Dazu wird ein Wetter – API ausgesucht, das nach REST realisiert ist und die Daten im JSON – Format abgerufen werden können.

**Exit:** Die Kommunikation mit Externen Diensten erfolgt mittels JSON – Format. Die Wetterdaten werden erfolgreich von externen Diensten abgefragt und zur Analyse der Bodendaten und Definition der Anleitungen herangezogen.

**Fail:** Der externe Dienst ist nicht erreichbar oder die gelieferten Daten sind nicht valide. Die Anleitungen werden fehlerhaft.

**Fallback:** Es müssen alternative zuverlässige Dienste verwendet werden oder es wird auf externe Daten verzichtet. Dadurch werden die Anleitungen nicht Wetterabhängig dargestellt.

### 12.1.7. Erstellung der Anleitungen für Mobiltelefone

**Beschreibung:** Falls ein Benutzer kein Smartphone besitzt, soll er die Empfehlungen in vereinfachter Form zeitnah auf seinem Mobiltelefon per SMS erhalten. Wegen der Plattformbeschränkungen können die Informationen nur in Form von Texten oder Bildern dargestellt werden. Dazu soll ein API für SMS Gateway verwendet.

**Exit:** Aktuelle Informationen werden im JSON – Format an einen externen Dienst geschickt und von dort per SMS an die Telefonnummer des Benutzers verschickt.

**Fail:** SMS wird nicht verschickt, da das SMS Gateway nicht erreichbar ist. Oder die Funktion wird gar nicht realisiert, da das Verschicken von SMS kostenpflichtig ist.

**Fallback:** Auf die Funktion kann verzichtet werden. Die Helfer informieren die Ackerbauer vor Ort.

### 12.1.8. Didaktische Darstellung der Informationen

**Beschreibung:** Das UI des Systems und die Informationen sollen didaktisch sinnvoll Visualisiert oder in anderen Formen dargestellt werden, damit auch die Analphabeten sie nachvollziehen können. Dabei ist wichtig, dass die UI Elementen Visualisierungen oder verbale Darstellung enthalten, damit die Funktionalität auch für Analphabeten verständlich wird

**Exit:** Die Informationen und UI Elemente sind für alle Stakeholdergruppen nachvollziehbar. Die Interaktionselemente sind mit sprechenden Bildern oder Audio versehen. Die User können dem Interaktionsfluss folgen.

**Fail:** Die Form der Informationsdarstellung ist ungeeignet, so dass das System nicht für alle Stakeholdergruppen gebrauchstauglich wird. Die User können das System nicht benutzen

**Fallback:** Auf die Funktion darf nicht verzichtet werden. Es sollen intensive Recherchen im Bereich der didaktischen Informationsdarstellung und Wissensvermittlung durchgeführt werden.

## 12.2. Personae

### Landwirt Amadou



Amadou ist 30 Jahre alt, ein afrikanischer Ackerbauer und lebt in Burkina Faso. Er hat eine Frau und 3 Kinder, welche er durch die Erträge seiner Hirseernte ernähren muss. Doch oft fallen die Erträge niedrig aus. Durch Dürre vertrocknen die Pflanzen, durch hohe Niederschläge werden die Pflanzen aus dem Boden gespült. Amadou besitzt ein Smartphone, allerdings kann er es nicht nutzen wie Menschen aus einem Industrieland. Lese- und Schreiben stellt für ihn ein erhebliches Problem dar, da er keine schulische Ausbildung genießen konnte und auch seine technischen Erfahrungen eher gering sind. Er weiß, dass er etwas an seinem Anbauverfahren ändern muss, denn die Existenz seiner Familie ist bedroht.

### Landwirtin Alika



Alika ist 37 Jahre alt, eine afrikanische Bäuerin und lebt in Nigeria. Sie und ihr Mann bauen Vanille an, um ihre Familie mit 4 Kindern zu ernähren. Vor einigen Jahren konnten sie immer gute Erträge erzielen und sorglos davon leben, doch ihre Ernten fallen Jahr für Jahr schlechter aus. Dies treibt die Familie immer weiter in die Armut. Woran das liegt, können Sie sich nicht erklären. Grund dafür ist, dass ihnen Fachwissen fehlt. Sie konnten keine Schule besuchen und haben den Anbau aus Tradition weitergeführt.

Außerdem sind Analphabeten und können nicht Lesen und Schreiben. Ein Smartphone konnte sich die Familie vor einiger Zeit leisten, jedoch fehlt ihnen das Wissen es richtig zu benutzen. Alika möchte die Ursache für die niedrigen Erträge herausfinden und ist auch bereit ihre Tradition des Vanilleanbaus aufzugeben, wenn sie eine Alternative finden würde mit welcher sie den Lebensunterhalt ihrer Familie sichern kann.

#### **Landwirt Bouba**



Bouba ist ein Landwirt einer Dorfgemeinschaft in Senegal. Er ist 43, hat drei Töchter und einen behinderten Sohn. Seine Familie lebt hauptsächlich von dem Reisanbau. Die Reisanbaugebiete sind aber in dem Land recht klein und machen nur geringen Anteil an der gesamten Landwirtschaft aus. Bouba tut alles, was ihm möglich ist, um seinen Familie zu ernähren. Doch trotz aller Mühen reicht es für höchstens zwei Mahlzeiten pro Tag, die hauptsächlich aus Reis bestehen, Fleisch gibt es immer seltener. Die Familie verhungert und ihre Existenz ist unter Gefahr. Den Umzug in die Stadt kann er sich nicht leisten. Außerdem muss er sich verschulden, um Saatgut und Düngemittel kaufen zu können. Selbstverständlich hat der Bouba keine Spezielle Technik und Verzüge, die ihm die Arbeit auf dem Acker erleichtern würden. Alle seine Kenntnisse im Gebiet basieren auf den Traditionen seiner Vorfahren. Er gehört zu den 58% der Analphabeten in Senegal, so dass Erwerb neues Wissen für Ihn eine erhebliche Barriere ist. Zudem hat er kein Smartphone, sondern nur ein altes Handy. Der Bouba ist trotzdem für jede Lösung offen, die ihm dabei helfen würde, Existenz seiner Familie sichern zu können.

#### **Helfer Andreas Baumann**



Andreas Baumann ist 32 Jahre alt und betreibt einen landwirtschaftlichen Betrieb mit Ackerbau in Niedersachsen (Deutschland). Er hat sich als Aufgabe gemacht sein Wissen an Menschen in Entwicklungsländern weiterzugeben. Sein Studium in Agrarwissenschaften hat er erfolgreich abgeschlossen und übernahm den Betrieb seines Vaters. Dementsprechend ist sein Fachwissen in



diesem Bereich besonders hoch und weiß dieses zu vermitteln. Als moderner Mann besitzt er Erfahrung im Umgang mit Smartphone, Tablet und Computer. Er ist offen und bereit sich für die Aufgabe als Vermittler weiter zu bilden und freut sich auf seine Aufgabe.

#### **Helferin Katharina Müller**



Katharina Müller ist 24 Jahre alt und betreibt gemeinsam mit ihrer Familie einen Ackerbaubetrieb in Schleswig-Holstein (Deutschland). Sie hat eine Ausbildung als Landwirtin und ihre Meisterprüfung absolviert. Katharina möchte den Menschen vor Ort Tipps und Tricks im Bereich Ackerbau zeigen, um die Erträge zu steigern. Dabei möchte sie auch Ansprechpartnerin für Frauen und Kinder der Familien sein. Sie besitzt ein gutes Fachwissen und ist besonders in der Praxis erfahren. Außerdem verfügt sie über ein gutes technisches Verständnis und weiß mit einem Smartphone, Tablet und einem Computer umzugehen.

### **12.3. Evaluation**

#### **12.3.1. Tasks**

##### **Test Task 1 (Helfer)**

##### **Ackerdaten eintragen**

Erstellen eines neuen Eintrags im System, nach dem die Bodendaten vor Ort aufgehoben wurden.

Sie sind in Senegal bei dem Landwirt Bouba. Im Rahmen Ihrer Mission in diesem Land, ist Ihre Aufgabe die Landwirte vor Ort zu unterstützen und das Wissen im Bereich Ackerbau langfristig zu vermitteln. Dennoch ist oft die fehlende Schreibe – und – Lesekompetenz der Landwirte eine Barriere, so dass ein passendes Werkzeug zur Wissensübermittlung fehlt. Ihnen wird das System zur Verfügung gestellt, welches dieses Problem lösen soll. Erstellen Sie dazu einen neuen Eintrag mit Ackerdaten:

1. Registrieren Sie sich im System
2. Navigieren Sie zum Formular
3. Geben Sie die benötigten Daten Ein
4. Speichern sie die Daten
5. Fertig!

##### **Test Task 2 (Helfer)**

**Anbauempfehlung anschauen**

Nach dem Sie ein Eintrag im System erstellt haben, müssen Sie nun die Anbauempfehlung dem Landwirt anzeigen. Erstmals müssen Sie die detaillierten Ackerdaten anschauen. Schauen Sie sich die Darstellung an und Identifizieren Sie die Probleme, die das Systems ermittelt hat. Navigieren sie dann zu der Anbauempfehlung des jeweiligen Eintrags.

1. Navigieren Sie zur detaillierten Informationen
2. Identifizieren Sie die Probleme der Pflanze
3. Navigieren Sie zum Anbauempfehlungen
4. Anbauempfehlungen dem Landwirt präsentieren
5. Fertig!

**Test Task 3 (Landwirt – Analphabet)****Registrieren im System**

Der Analphabet soll sich (mit der Unterstützung des Helfers) im System registrieren. Dazu soll er zum jeweiligen Fenster navigieren und sich dann entweder mit Email oder der Telefonnummer registrieren. (Auf detaillierte Beschreibung wird hier verzichtet)

**Test Task 4 (Landwirt – Analphabet)****Anbauempfehlung anschauen**

Der Analphabet soll die Anbauempfehlungen anschauen (Auf detaillierte Beschreibung wird hier verzichtet).

### 12.3.2. User 1

Location: *Room*

Username: *Dolly*

User NO: *1*

Data Collector: *Sergej Atamantschuk*

Iterations: *2*

**1. Are you:**

- ☐ Male  
☒ Female

**2. How old are you**

- ☒ 18-25  
☐ 26-40  
☐ 41-55  
☐ Over 55

**3. Are you color blind in any way?**

- ☒ No  
☐ Yes (please describe) \_\_\_\_\_

**4. Describe your education background**

- ☐ None  
☐ Elementary school  
☐ Middle schools  
☐ High school (attended but did not graduate)  
☐ High school degree  
☐ College (no degree)  
☐ College degree  
☐ Graduate school (no degree)  
☒ Graduate school degree  
☐ Other (please explain) \_\_\_\_\_

**5. How would you describe your experience level in agriculture?**

- ☒ Novice  
☐ Advanced  
☐ Competence  
☐ Expertise

**6. Do you have any experience using Android system?**

- ☐ No  
☒ Yes  
☐ Yes, but only older versions  
☐ Yes, but only newer versions

**7. How would you describe your general level of expertise in using Android applications**

- ☐ None  
☐ Low  
☒ High  
☐ Other (please describe) \_\_\_\_\_

**8. How would you describe your general level of smartphone experience?**

- ☐ None  
☐ Low



x \_\_\_\_\_ High  
 \_\_\_\_\_ Other (please describe) \_\_\_\_\_

**9. How would you describe your english language level?**

\_\_\_\_\_ None  
 \_\_\_\_\_ Low  
 \_\_\_\_\_ middle  
 x \_\_\_\_\_ High  
 \_\_\_\_\_ Other (please describe) \_\_\_\_\_

**Data Collection Sheet – Task 1(User 1)**

**Correct steps:**

Mit dem Tab zur Registrierung navigieren. Benötigte Daten eingeben. Danach landet man in dem Rootactivity. Plus Floating Action Button anklicken und zum Formular wechseln. Daten eingeben. In Actionbar Save – Button anklicken. Done!

Correct steps	Correct ?		Time	Errors/Comments
	Y	N		
<b>Touch oder Wischen zu dem SIGN UP Tab</b> , um zum Registrationsformular zu navigieren		N	1.Iteration 2 min	Versucht sich direkt im Sign in Fenster zu registrieren, bekommt Fehlermeldung. Comment vom User: „Sign In“ in „Log In“ umbenennen, Tab Navigation ist Praktisch
Datenfelder ausfüllen	Y		2.Iteration ≤1 min	Check boxen sollen sprechenden Namen haben, die Auswahl des User-Profiles war nicht eindeutig
Auf <b>SIGN UP</b> Button klicken	Y			
Plus Floating Action Button anklicken	Y			
Datenfelder ausfüllen: Falls bestimmte Daten nicht vorliegen, auf ein jeweiliges <b>Icon</b> , klicken, um diese Daten von System automatisch ermitteln zu lassen	Y		1.Iteration ≤5.5 min  2.Iteration ≤4 min	Sollte deutlicher drauf hingewiesen werden, dass Felder wie „Location“ oder „Airtemp“ vom System automatisch ausgefüllt werden müssen.
Auf <b>Save – Icon</b> in der Action Bar klicken	Y			
Feedback Meldung erhalten	Y			Toast Message erkannt

**Data Collection Sheet – Task 2 (User 1)**

**Correct steps:**

Auf Listeneintrag klicken und zur Detailansicht wechseln. Probleme einer Pflanze anhand der Visualisierung interpretieren. Auf **SHOW TUTORIAL** klicken. Zur Anzeige der detaillierten Empfehlungen auf jeweilige Icons klicken.

Correct steps	Correct ?		Time (2.Iteration)	Errors/Comments
	Y	N	≤3 sec	
Auf einen Listeneintrag in der Liste <b>klicken</b>	Y			
Angezeigte Information interpretieren	Y			Informationen verständlich dargestellt, Die Bedeutung der Farben sollte jedoch erläutert werden
Button <b>SHOW TUTORIAL</b> anklicken	Y			Der Inhalt der meisten Icons wurde erkannt, die Icons sollten jedoch noch präzisiert werden, um die Realen Objekte der Welt eindeutiger darzustellen

### 12.3.3. User 2

Location: *Room*

Username: *Maman*

User NO: 2

Data Collector: *Sergej Atamantschuk*

Iterations: 2

#### 1. Are you:

\_\_\_\_\_ Male

☒ \_\_\_\_\_ Female

#### 2. How old are you

\_\_\_\_\_ 18-25

\_\_\_\_\_ 26-40

☒ \_\_\_\_\_ 41-55

\_\_\_\_\_ Over 55

#### 3. Are you color blind in any way?

☒ \_\_\_\_\_ No

\_\_\_\_\_ Yes (please describe) \_\_\_\_\_

#### 4. Describe your education background

\_\_\_\_\_ None

\_\_\_\_\_ Elementary school

\_\_\_\_\_ Middle schools

\_\_\_\_\_ High school (attended but did not graduate)

\_\_\_\_\_ High school degree

\_\_\_\_\_ College (no degree)

☒ \_\_\_\_\_ College degree

\_\_\_\_\_ Graduate school (no degree)

\_\_\_\_\_ Graduate school degree

\_\_\_\_\_ Other (please explain) \_\_\_\_\_

#### 5. How would you describe your experience level in agriculture?

\_\_\_\_\_ Novice

\_\_\_\_\_ Advanced

\_\_\_\_\_ Competence

☒ \_\_\_\_\_ Expertise

**6. Do you have any experience using Android system?**

- ☒ No  
☐ Yes  
☐ Yes, but only older versions  
☐ Yes, but only newer versions

**7. How would you describe your general level of expertise in using Android applications**

- ☒ None  
☐ Low  
☐ High  
☐ Other (please describe) \_\_\_\_\_

**8. How would you describe your general level of smartphone experience?**

- ☐ None  
☒ Low  
☐ High  
☐ Other (please describe) \_\_\_\_\_

**9. How would you describe your english language level?**

- ☐ None  
☒ Low  
☐ middle  
☐ High  
☐ Other (please describe) \_\_\_\_\_

**Data Collection Sheet – Task 3 (User 2)**

**Correct steps:**

Auf Tab **SIGN UP** klicken. Auf Button mit **Email** oder **Telefonnummer** registrieren klicken. Daten eingeben. Auf **SIGN UP** Button klicken.

Correct steps	Correct?		Time(2. Iteration)	Errors/Comments
	Y	N		
<b>Touch</b> oder <b>Wischen</b> zu dem <b>SIGN UP</b> Tab, um zum Registrationsformular zu navigieren		N	1.Iteration 3 min	Versucht sich direkt im <i>Sign in</i> Fenster zu registrieren, bekommt Fehlermeldung.
Button <b>Mit Email</b> oder <b>Mit Telefonnummer</b> anklicken	Y	Y	2.Iteration ≤1 min	
Jeweilige Datenfelder ausfüllen		Y		Checknoxen waren unklar

**Data Collection Sheet – Task 4 (User 2)**

**Correct steps:**

Auf Listeneintrag klicken und direkt zur interaktiven Ansicht wechseln. Zur Anzeige der detaillierten Empfehlungen auf jeweilige Icons klicken.

Correct steps	Correct?		Time(2. Iteration)	Errors/Comments
	Y	N		
Auf einen Listeneintrag in der Liste <b>klicken</b>	Y		<=2 sec	
Daten interpretieren können	Y			Der Inhalt konnte sehr gut und schnell erkannt werden. Alle Icons wurden verstanden
Falls nötig, <b>Ear-Icon</b> in der Actionbar anklicken, um die Benutzungsanleitung anzuhören	Y			Icon zum Anhören wurde erkannt
Falls nötig, auf Text klicken, um den Inhalt anzuhören		N		Der Text, der vertont werden kann, sollte deutlicher dargestellt werden
Zur ausführlichen Info auf beiliegende Icons klicken	Y			

### 12.3.4. User 3

Location: *Room*

Username: *BabyMila*

User NO: 3

Data Collector: *Sergej Atamantschuk*

Iterations: 1

#### 1. Are you:

☐ Male

☒ Female

#### 2. How old are you

☒ 18-25

☐ 26-40

☐ 41-55

☐ Over 55

#### 3. Are you color blind in any way?

☒ No

☐ Yes (please describe) \_\_\_\_\_

#### 4. Describe your education background

☐ None

☐ Elementary school

☐ Middle schools

☐ High school (attended but did not graduate)

☒ High school degree

☐ College (no degree)

☐ College degree

☐ Graduate school (no degree)

☐ Graduate school degree

☐ Other (please explain) \_\_\_\_\_

#### 5. How would you describe your experience level in agriculture?

☒ Novice

☐ Advanced

\_\_\_\_\_ Competence  
 \_\_\_\_\_ Expertise

**6. Do you have any experience using Android system?**

\_\_\_\_\_ No  
 x \_\_\_\_\_ Yes  
 \_\_\_\_\_ Yes, but only older versions  
 \_\_\_\_\_ Yes, but only newer versions

**7. How would you describe your general level of expertise in using Android applications**

\_\_\_\_\_ None  
 \_\_\_\_\_ Low  
 x \_\_\_\_\_ High  
 \_\_\_\_\_ Other (please describe) \_\_\_\_\_

**8. How would you describe your general level of smartphone experience?**

\_\_\_\_\_ None  
 \_\_\_\_\_ Low  
 x \_\_\_\_\_ High  
 \_\_\_\_\_ Other (please describe) \_\_\_\_\_

**9. How would you describe your english language level?**

\_\_\_\_\_ None  
 \_\_\_\_\_ Low  
 x \_\_\_\_\_ middle  
 \_\_\_\_\_ High  
 \_\_\_\_\_ Other (please describe) \_\_\_\_\_

**Data Collection Sheet – Task 1(User 3)**

**Correct steps:**

Mit dem Tab zur Registrierung navigieren. Benötigte Daten eingeben. Danach landet man in dem Rootactivity. Plus Floating Action Button anklicken und zum Formular wechseln. Daten eingeben. In ActionBar Save – Button anklicken. Done!

Correct steps	Correct ?		Time	Errors/Comments
	Y	N		
<b>Touch oder Wischen zu dem SIGN UP Tab</b> , um zum Registrierungsformular zu navigieren		N	≤2 min	Versucht sich direkt im Sign in Fenster zu registrieren, bekommt Fehlermeldung. Usercomment: „ <i>weil in der Aufgabenstellung „Registrieren“ steht, denkt man sich, es öffnet sich direkt ein Registrierungsformular und liest nicht, was in der Tab Überschrift steht</i> “
Datenfelder ausfüllen	Y			Check boxen sollen sprechenden Namen haben, die Auswahl des User-Profiles war nicht eindeutig
Auf <b>SIGN UP</b> Button klicken	Y			
Plus Floating Action Button anklicken	Y		≤4 min	
Datenfelder ausfüllen: Falls bestimmte Daten nicht vorliegen, auf ein jeweiliges <b>Icon</b> , klicken, um diese Daten	Y			Sollte deutlicher drauf hingewiesen werden, dass Felder wie „Location“ oder „Airtemp“ vom System automatisch ausgefüllt werden müssen.

von System automatisch ermitteln zu lassen				
Auf <b>Save – Icon</b> in der Action Bar klicken	Y			
Feedback Meldung erhalten	Y			Toast Message erkannt

### Data Collection Sheet – Task 2(user 3)

#### Correct steps:

Auf Listeneintrag klicken und zur Detailansicht wechseln. Probleme einer Pflanze anhand der Visualisierung interpretieren. Auf **SHOW TUTORIAL** klicken. Zur Anzeige der detaillierten Empfehlungen auf jeweilige Icons klicken.

Correct steps	Correct?		Time	Errors/Comments
	Y	N		
Auf einen Listeneintrag in der Liste <b>klicken</b>	Y			
Angezeigte Information interpretieren	Y		≤ 6 sec	Informationen verständlich dargestellt. Die Bedeutung der Hintergrundfarben sollte jedoch erläutert werden. Bei der negativen Hintergrundfarbe nicht klar, ob der bestimmte Wert höher oder niedriger sein soll
Button <b>SHOW TUTORIAL</b> anklicken	Y			Der Inhalt der meisten Icons wurde genau erkannt. Die Unterteilung in <b>Boden</b> und <b>Luft</b> Daten wurde sehr gut erkannt.

### 12.3.5. User 4

Location: *Room*

Username: *Baban*

User NO: *4*

Data Collector: *Sergej Atamantschuk*

Iterations: *1*

#### 1. Are you:

\_\_\_\_\_ Male

x \_\_\_\_\_ Female

#### 2. How old are you

\_\_\_\_\_ 18-25

\_\_\_\_\_ 26-40

\_\_\_\_\_ 41-55

x \_\_\_\_\_ Over 55

#### 3. Are you color blind in any way?

x \_\_\_\_\_ No

\_\_\_\_\_ Yes (please describe) \_\_\_\_\_

#### 4. Describe your education background

\_\_\_\_\_ None

\_\_\_\_\_ Elementary school

\_\_\_\_\_ Middle schools

\_\_\_\_\_ High school (attended but did not graduate)

- ☐ High school degree  
☐ College (no degree)  
☒ College degree  
☐ Graduate school (no degree)  
☐ Graduate school degree  
☐ Other (please explain) \_\_\_\_\_

**5. How would you describe your experience level in agriculture?**

- ☐ Novice  
☐ Advanced  
☒ Competence  
☐ Expertise

**6. Do you have any experience using Android system?**

- ☒ No  
☐ Yes  
☐ Yes, but only older versions  
☐ Yes, but only newer versions

**7. How would you describe your general level of expertise in using Android applications**

- ☒ None  
☐ Low  
☐ High  
☐ Other (please describe) \_\_\_\_\_

**8. How would you describe your general level of smartphone experience?**

- ☒ None  
☐ Low  
☐ High  
☐ Other (please describe) \_\_\_\_\_

**9. How would you describe your english language level?**

- ☒ None  
☐ Low  
☐ middle  
☐ High  
☐ Other (please describe) \_\_\_\_\_

**Data Collection Sheet – Task 3 (User 4)**

**Correct steps:**

Auf Tab **SIGN UP** klicken. Auf Button mit **Email** oder **Telefonnummer** registrieren klicken. Daten eingeben. Auf **SIGN UP** Button klicken.

Correct steps	Correct?		Time	Errors/Comments
	Y	N		
Touch oder Wischen zu dem SIGN UP Tab, um zum Registrationsformular zu navigieren		N	∞	Die Authentifikation im System ohne weitere Hilfe war nicht möglich!!! Keine English Kenntnisse – kann nicht lesen!

Button <b>Mit Email</b> oder <b>Mit Telefonnummer</b> anklicken		N		
Jeweilige Datenfelder ausfüllen		N		

### Data Collection Sheet – Task 4(User 4)

#### Correct steps:

Auf Listeneintrag klicken und direkt zur interaktiven Ansicht wechseln. Zur Anzeige der detaillierten Empfehlungen auf jeweilige Icons klicken.

Correct steps	Correct?		Time	Errors/Comments
	Y	N		
Auf einen Listeneintrag in der Liste <b>klicken</b>	Y		≤3sec	
Daten interpretieren können	Y			Der Inhalt konnte nur teilweise erkannt werden. <b>PH – Icon</b> wurde als Feuchtigkeit interpretiert. <b>Höhe – Icon</b> wurde gar nicht erkannt (war nicht klar was abgebildet ist). Hintergrundfarbe konnte nicht interpretiert werden. Die Unterteilung in <b>Boden</b> und <b>Luft</b> Daten wurde sehr gut erkannt. Das Hintergrundbild der Activity wurde als aktuelle Wetterlage interpretiert, was es nicht war.
Falls nötig, <b>Ear-Icon</b> in der Actionbar anklicken, um die Benutzungsanleitung anzuhören		N		Die Funktion wurde nicht erkannt
Falls nötig, auf Text <b>klicken</b> , um den Inhalt anzuhören		N		Wurde nicht erkannt. Der Text, der vertont werden kann, sollte deutlicher dargestellt werden
Zur ausführlichen Info auf beiliegende Icons <b>klicken</b>		N		Funktion wurde nicht erkannt

## 12.4. Glossar

Begriff	Erklärung
Analphabeten	Menschen mit fehlenden Lese- und Schreibfähigkeiten.
Anbauempfehlung	Eine Empfehlung des Systems an die Landwirte, wie sie bestimmte Pflanzen optimal anbauen können.
Bodenanalyse	Die Bodenanalyse umfasst den Vergleich der angegebenen Daten mit den standardisierten Normwerten.
Bodendaten	Bodendaten sind alle Daten, welche die Beschaffenheit des Bodens repräsentieren. Dies können Bodenfeuchtigkeit, pH-Wert, Bodenbeschaffenheit etc. sein.
Helfer	Menschen welche sich für Hilfsorganisationen engagieren und den Menschen in Entwicklungsländern vor Ort tatkräftig zur Seite stehen.



Landwirt	Menschen welche in Entwicklungsländern Ackerbau betreiben.
Pflanzentopologie	Eine Sammlung von Daten bzw. Informationen über Pflanzen.
Root-Activity	Start-Fenster des Systems
Tutorial	Ein Tutorial ist eine interaktive Anleitung für den Landwirt. Sie beschreibt, wie er seine Pflanzen zu pflegen und anzubauen hat.
Userdaten	Daten, die im Bezug auf den User verwendet werden.
Entry	Bezeichnet einen Eintrag, der vom Benutzer erstellt und im System gespeichert wird. Beinhaltet alle Daten zu einem bestimmten Ackerfeld
Activity	Ein Fenster im Android System

## 12.5. Quellenverzeichnis

Mayhew, Deborah: The Usability Engineering Lifecycle 1999.

<https://www.oxfam.de/unsere-arbeit/projekte/burundi-laendliche-entwicklung> (30.05.17)

[https://www.oxfam.de/sites/default/files/webfm/getting\\_it\\_right\\_from\\_the\\_start\\_final010911-gs.pdf](https://www.oxfam.de/sites/default/files/webfm/getting_it_right_from_the_start_final010911-gs.pdf)  
(30.05.17)

<https://www.oxfam.de/unsere-arbeit/projekte/mauretanien-ernaehrung-sichern-existenzgrundlage-staerken> (01.06.17)

<http://www.kgv-dauieranlage-amdammsteg.de/bewaesserung.html> (06.06.17)

<http://www.beregnungsverband.de/service-nach-mass/wasserbedarf-an-pflanzen.html> (07.06.17)

<https://material.io/guidelines/> (06.06.17)