TH Köln

Studiengang Medieninformatik

EIS Projekt

Sommersemester 2017

**Konzept des Projekts**

**Harvest Hand**

Studierende:

Franziska Gonschor

Sergej Atamantschuk

Betreuer:

Robert Gabriel

Prof. Dr. Gerhard Hartmann

Prof. Dr. Kristian Fischer

Inhaltsverzeichnis

[Inhaltsverzeichnis II](#_Toc482459806)

[Abbildungsverzeichniss IV](#_Toc482459807)

[1. Einleitung 1](#_Toc482459808)

[1.1. Nutzungsproblem 1](#_Toc482459809)

[1.2. Ziel des Projekts 1](#_Toc482459810)

[2. Domänenrecherche 1](#_Toc482459811)

[3. Marktrecherche 1](#_Toc482459812)

[3.1. Konkurrenzprodukte – icow 1](#_Toc482459813)

[3.2. Konkurrenzprodukte – e-Soko 2](#_Toc482459814)

[3.3. Konkurrenzprodukte – Irmgard 2](#_Toc482459815)

[3.4. Fazit 3](#_Toc482459816)

[4. Alleinstellungsmerkmale 3](#_Toc482459817)

[5. Stakeholderanalyse 3](#_Toc482459818)

[6. Zielhierarchie 6](#_Toc482459819)

[6.1. Operative Ziele 6](#_Toc482459820)

[6.2. Taktische Ziele 6](#_Toc482459821)

[6.3. Strategische Ziele 6](#_Toc482459822)

[7. Risiken 7](#_Toc482459823)

[8. POCs 8](#_Toc482459824)

[8.1. Benutzer anlegen 8](#_Toc482459825)

[8.2. Eintrag mit Ackerdaten erstellen 8](#_Toc482459826)

[8.3. Collaborators dem Eintrag hinzufügen 8](#_Toc482459827)

[8.4. Effiziente Datenübertragung 8](#_Toc482459828)

[8.5. Auswertung der Bodendaten 9](#_Toc482459829)

[8.6. Effiziente Erstellung interaktiver Anleitungen 9](#_Toc482459830)

[8.7. Klimatische Daten der Externen Dienste 9](#_Toc482459831)

[8.8. Erstellung der Anleitungen für Mobiltelefone 10](#_Toc482459832)

[8.9. Didaktische Darstellung der Informationen 10](#_Toc482459833)

[8.10. Fazit 10](#_Toc482459834)

[9. Methodischer Rahmen 10](#_Toc482459835)

[9.1. Auswahl des methodischen Rahmens 10](#_Toc482459836)

[9.2. Auswahl des Vorgehensmodells 11](#_Toc482459837)

[9.3. Fazit 11](#_Toc482459838)

[10. Kommunikationsmodell 13](#_Toc482459839)

[10.1. Deskriptives Modell 13](#_Toc482459840)

[10.2. Präskriptives Modell 13](#_Toc482459841)

[11. Architektur 13](#_Toc482459842)

[12. Quellenverzeichnis 14](#_Toc482459843)

Abbildungsverzeichniss

[Abb. 1 Architekturmodell 4](#_Toc482110715)

# Einleitung

## 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.

## 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.

# Domänenrecherche

# Marktrecherche

## Konkurrenzprodukte – icow

iCow ist eine App, welche afrikanische Bauern bei der Viehzucht unterstützen soll. Der Nutzer registriert jede seiner Kühe und bekommt Haltungs- und Nutzungsvorschläge im Bezug auf jedes einzelne Tier. Termine werden gesetzt und der Landwirt rechtzeitig an diese erinnert. Außerdem werden Tipps im Bereich Fütterung und Milchproduktion regelmäßig zur Verfügung gestellt. Damit die Bauern einen konstanten Bestand sichern können, werden auf die Themen Fortpflanzung und insbesondere künstliche Befruchtung eingegangen, um leistungsfähige Tiere zu züchten.

Diese App passt sich an seinen Nutzer an und so muss man weder lesen können, noch ein Smartphone besitzen. Alle Informationen werden mittels Sprachnachricht übermittelt.

iCow hat viele positive Rezensionen, denn die Landwirte konnten nachweislich die Sterberate der Kühe senken und gesundes Vieh züchten. Folglich erzielten die Landwirte einen deutlich gestiegenen finanziellen Ertrag.

|  |  |
| --- | --- |
| **Vorteile** | **Nachteile** |
| Nutzer können einzelne Tiere registrieren und bekommen daher spezifische Tipps. | Die App ist nicht frei zugänglich. |
| Es können Termine gesetzt werden an die, die Landwirte rechtzeitig erinnert werden. | Es werden ausschließlich über Audio Inhalte vermittelt. |
| Allgemeine Tipps zur Fütterung, Milchproduktion, Fortpflanzung etc. werden regelmäßig dem Benutzer mitgeteilt. |  |
| Der Nutzer muss nicht lesen können. Alle Informationen werden mittels Sprachnachricht übermittelt. |  |
| Icow hat bereits positive Rezensionen und es konnte nachweislich die Sterberate der Kühe gesenkt werden. Daraus resultierte ein höherer finanzieller Ertrag für die Viehzüchter. |  |

## Konkurrenzprodukte – e-Soko

e-Soko ist ein Dienst, welcher Bauern über aktuelle Marktpreise Ihrer Produkte informiert. Diese Informationen erreichen die Landwirte per SMS. Dies ermöglicht es den Bauern ihre Waren rechtzeitig auf dem Markt zu einem guten Preis anzubieten. So wird ein Handel ohne Zwischenhändler unterstützt, welche meist einen großen Abschlag des Erlöses für den Verkauf erhalten haben. Außerdem werden Daten zum Wetter, Erinnerungen, sowie hilfreiche Tipps im Bezug auf Pflügen, Säen, Düngen und Ernten mitgeteilt.

|  |  |
| --- | --- |
| **Vorteile** | **Nachteile** |
| Fördert die Landwirte nachhaltig zu wirtschaften , um einen größtmöglichen Gewinn zu erzielen. | Nicht für Analphabeten geeignet, da es ausschließlich auf schriftlichen Informationen basiert. |
| Es werden Daten zum Wetter und Tipps zur Feldbearbeitung, Ernten etc. mitgeteilt. | Nicht frei zugänglich. |

## Konkurrenzprodukte – Irmgard

IRMGARD ist eine App mit der Jugendliche und Erwachsene Lesen und Schreiben lernen können. Geboten werden neun Level mit 100 verschiedenen Übungen um das Lesen und Schreiben zu trainieren. Es sind keine Kenntnisse nötig, um die App nutzen zu können. Die App ist nur auf Android-Geräten nutzbar und multimedial ausgestattet. Die Vermittlung der Inhalte wird durch Audio, Video und Animationen unterstützt. Die Betroffenen können ganz bequem von Zuhause aus lernen und empfinden kein Schamgefühl. Ein großer Nachteil ist jedoch, dass die Installationsdatei der Applikation sehr groß ist und daher viel Speicher in Anspruch nimmt.

|  |  |
| --- | --- |
| **Vorteile** | **Nachteile** |
| Bietet Jugendlichen und Erwachsenen die Möglichkeit Lesen und Schreiben in verschiedenen Leveln zu lernen. | Nur für Android-Geräte verfügbar. |
| Ein schamfreier Lernvorgang kann gewährleistet werden. | Beansprucht stark den Akku. |
| Es sind keine Kenntnisse nötig, um die App nutzen zu können. | Es wird viel Speicher benötigt, da die Installationsdatei sehr groß ist. |
| Informationen werden angepasst auf die Nutzer präsentiert in Form von Audio, Video und Animationen. |  |
| Kann kostenlos runter geladen werden. |  |

## Fazit

Insgesamt kann festgehalten werden, dass es bisher kein System gibt, welches sich mit unserem Nutzungsproblem beschäftigt. Es gibt Systeme die decken den Bereich Landwirtschaft ab und es gibt Systeme, welche sich mit dem Überwinden der Wissensbarriere im Bezug auf Lesen und Schreiben beschäftigen. Eine Kombination kann nur die App iCow bieten, welche sich jedoch eine andere Domäne abdeckt.

# Alleinstellungsmerkmale

Die Marktrecherche ergab vor allem, dass das geplante System keine direkten Konkurrenzprodukte hat, die sich unmittelbar in der gleichen Domäne befinden. Aktuell gibt es keine Systeme auf dem Markt, die sich mit der didaktischen Wissensvermittlung im Bereich des Ackerbaubetriebes beschäftigen, was an sich schon ein Alleinstellungsmerkmal ist.

Eins der wichtigsten Alleinstellungsmerkmale ist es, dass das System die Wissensbarriere zwischen den Menschen in Entwicklungsländer und denen der Industrieländer schließt. Dies geschieht durch eine maßgeschneiderte Vermittlung von Informationen für Analphabeten, welche Ackerbau betreiben. Außerdem werden nicht nur die Menschen vor Ort angesprochen, sondern auch die Helfer, welche die Menschen im Bereich Landwirtschaft aufklären möchten. Daher ist dieses System für zwei Zielgruppen ausgerichtet. Des Weiteren wird das System kostenlos und frei zur Verfügung stehen.

Das System sollte zudem die Bauer langfristig unterstützen, in dem die Informationen zur lokalen Wetterbedingungen in bestimmten Zeitabschnitten und langjährigen Klimaänderungen zur Verfügung gestellt werden, so dass der Ackerbau nachhaltig getrieben werden kann.

# Stakeholderanalyse

Im Folgenden Kapitel wird die Stakeholderanalyse durchgeführt. Dabei werden möglichst alle relevanten Stakeholdergruppen berücksichtigt, deren Erfordernisse identifiziert und Bezug zum System bestimmt. Die Analyse ergab verschieden Gruppen, deren Erfordernisse und Erwartung sich in bestimmten Teilen des Systems überschneiden können. Grundsätzlich lassen sich die Stakeholder in primäre und sekundäre aufteilen. Zu den primären Stakeholdern gehören Bauer und Helfer, die unmittelbar mit dem System interagieren. Zu den sekundären gehören unter anderem die Endverbraucher der Anbauprodukte, Hilfsorganisationen, Händler usw., da diese nicht direkt mit dem System interagieren.

| # | Bezeichnung | Beziehung zum System | Objektbereich der Beziehung | Erfordernis, Erwartung |
| --- | --- | --- | --- | --- |
| 1 | Bauer | Anrecht | Teilsystem | Der Benutzer möchte Informationen zum Ackerbau erhalten. |
|  |  | Anspruch | Teilsystem | Der Benutzer möchte die Informationen in Form von Visualisierung und Sprachnachrichten für Analphabeten erhalten. |
|  |  | Anspruch | Teilsystem | Der Benutzer möchte spezielle Informationen zu kurzfristigen Wetterereignissen erhalten, die den Anbau beeinflussen. |
|  |  | Anspruch | Teilsystem | Der Benutzer möchte spezielle Informationen zur langfristigen Vorausschau von Vegetationsveränderungen erhalten. Welche Pflanzen kann man auch in 2 Jahren noch mit hohen Ertragen anbauen? |
|  |  | Interesse | Teilsystem | Der Benutzer möchte, dass das System möglichst wenige Daten über das Internet austauscht, damit sein mobiles Datenvolumen nicht zu stark belastet wird. |
|  | Familienbetriebe | Anspruch | Teilsystem | Familienbetriebe möchten ihre Erträge steigern, um mehr Produkte vermarkten zu können, so dass sich ihre Lebensumstände verbessern. |
| 2 | Mitarbeiter bzw. Helfer | Anrecht | Teilsystem | Der Benutzer möchte Lehrmaterial zur Unterstützung der Lehrinhalte erhalten. |
|  |  | Anspruch | Teilsystem | Der Benutzer möchte vom System aktiv bei der Vermittlung unterstützt werden. |
|  |  | Interesse | Teilsystem | Der Benutzer möchte, dass das System möglichst wenige und kleine Daten über das Internet austauscht, damit sein mobiles Datenvolumen nicht zu stark belastet wird. |
| 3 | Endverbraucher der Anbauprodukte | Anspruch | Teilsystem | Der Endverbraucher möchte weiterhin seine Produkte konsumieren können, was eine bestimmte Höhe von Ernteerträgen voraussetzt. |
| 4 | Hilfsorganisationen | Anrecht | System | Der Benutzer möchte die Informationen für die Bauern auf dem aktuellsten Stand halten. |
|  |  | Anrecht | System | Der Benutzer möchte die Gesamten Lebensumstände der Menschen in Entwicklungsländern verbessern. |
| 5 | Händler | Interesse | Teilsystem | Händler möchten, dass Landwirte höhere Erträge erzielen, damit sie mehr Waren zum Handel zu Verfügung haben. |
|  | Saatguthändler | Interesse | Teilsystem | Saatguthändler möchten ihre Produkte an die Landwirte vermarkten, um ihren Umsatz zu steigern. |
|  | Produkthändler | Interesse | Teilsystem | Produkthändler möchten Produkte von den Landwirten erwerben, um sie auf den lokalen Märkten weiter zu verkaufen. |
|  |  | Interesse | Teilsystem | Produkthändler möchten den Produzenten Produkte verkaufen, welche sie selbst nicht anbauen. |
|  | Düngemittel-Händler | Interesse | Teilsystem | Händler möchten ihre Düngemittel vermarkten, um höhere Erträge zu erzielen und den Umsatz zu steigern. |
| 6 | Regierung | Interesse | Teilsystem | Die Regierung möchte die Wirtschaft in ihrem Land nachhaltig unterstützen und verbessern, um die Lebensumstände der Bevölkerung zu verbessern. |
| 7 | Kreditinstitute | Interesse | Teilsystem | Kreditinstitute möchten Kredite an die Bauern vergeben um Einnahmen zu erzielen. |
|  |  | Interesse | Teilsystem | Kreditinstitute möchten, dass Bauern ihr Geld bei der Bank anlegen, welches ihr dann zur Verfügung steht. |
| 8 | Spender/Investoren | Anrecht | System | Spender und Investoren möchten die Lebensumstände von Menschen in Entwicklungsländern nachhaltig verbessern. |
| 9 | Telekommunikationsanbieter | Interesse | Teilsystem | Telekommunikationsanbieter möchten mit der Regierung kooperieren, um das Datenvolumen für dieses System nicht zu belasten. In ihrem Interesse ist es die Landwirtschaft zu fördern und die Lebensbedingungen der Menschen zu verbessern. |

# Zielhierarchie

Im Folgenden werden die Entwicklungsziele erläutert. Dabei werden diese in strategische, taktische und operative Ziele gegliedert.

## Operative Ziele

• Der Projektplan muss erstellt und gepflegt werden. Dabei muss der Workload von 600 Stunden verteilt werden.

• Ein Rapid-Prototype muss entstehen, um mindestens einen PoC zu demonstrieren.

• Im präskriptiven Kommunikationsmodell müssen die mit dem System direkt interagierenden Stakeholder dargestellt werden.

• Es muss sich auf ein Vorgehensmodell beschränkt werden, welches auf das Projekt skaliert wird. Dazu müssen alle notwendigen Artefakte erstellt werden.

• In der Domänenrecherche müssen alle wichtigen Punkte zur weiteren Entwicklung erfasst werden.

• Bei der Spezifizierung der Risiken sollen bereits Gegenmaßnahmen für das Auftreten eines Risikos festgelegt werden.

• Es müssen Anforderungen an das System spezifiziert werden. Diese sollen in funktionale, organisationale, qualitative und technischen Anforderungen gegliedert werden.

• Es soll ein Style Guide entwickelt werden, welcher die Entwicklung des User Interfaces bestimmt.

• Jeder Projektfortschritt muss in der Dokumentation festgehalten werden.

• Das Projekt soll so konzeptioniert sein, dass eine gute Basis geschaffen wird um das System mit weiteren Features ausstatten zu können

## Taktische Ziele

• Der Projektplan soll eingehalten werden.

• Proof of Concepts sollen getestet und umgesetzt werden.

• Ein Prototyp des Systems soll erstellt werden, um die Usability zu testen.

• Der Prototyp soll sich durch Einhaltung von Style Guides übersichtlich gestaltet werden.

• Die Kommunikationswege zwischen den Stakeholdern sollen analysiert und visualisiert werden.

• Der methodische Rahmen muss in Form eines Vorgehensmodells festgelegt werden.

• Um das Projekt erfolgreich abzuschließen, soll lösungsorientiert entwickelt werden. Das System gilt als abgeschlossen, wenn alle Funktionalitäten implementiert sind.

## Strategische Ziele

• Durch die Kombination möglichst vieler Daten, sollen Benutzer möglichst präzise und individuelle Anbauempfehlungen erhalten. Dabei soll jeder Bauer mindestens eine Anbauempfehlung erhalten.

• Die Applikation muss zur Lösung des Eingangs beschriebenen Nutzungsproblem verwendet werden können.

• Die Kommunikation zwischen den Stakeholdern soll verbessert werden.

• Die Applikation soll Verwendung im Alltag finden.

• Die Benutzung der Applikation soll nicht als Aufwand wahrgenommen werden.

• Die Verwendung der Applikation soll für Benutzer möglichst kostenfrei sein und daher soll ein hoher Datenverbrauch vermieden werden. Ziel ist es eine Übertragungszeit von 5 Sekunden nicht zu überschreiten.

# Risiken

# POCs

Die POCs beschreiben wesentliche Funktionen und Alleinstellungsmerkmale des Systems, von denen schließlich die Realisierbarkeit und Usability des Systems abhängen. Dabei beziehen sich einige direkt auf die ermittelten Risiken.

## Benutzer anlegen

**Beschreibung**: Helfer und Bauer müssen jeweils Profile erstellen können, um Zugriffsrechte zu erhalten. Dabei werden verschieden Daten der Benutzer als JSON – Objekt an Server geschickt und gespeichert.

**Exit**: Nach Eingabe erforderlicher Daten wird der User in Datenbank angelegt. Nur Mitarbeiter der Hilfsorganisationen können ein Helfer – Profil erstellen.

**Fail**: Der Benutzer hat keine Berechtigung um Helfer – Profil zu erstellen. 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

## Eintrag mit Ackerdaten erstellen

**Beschreibung**: Der User muss einen Eintrag erstellen können, wo Relevante Daten der Anbaufläche eingegeben werden zum weiteren Speichern und Verarbeiten auf dem Server. Die Einträge mit Ackerdaten werden in einer Liste im Userprofil gespeichert. Die Daten werden auf dem Client interaktiv mit Grafiken, Animationen und Bildern dargestellt.

**Exit**: Eintrag wird erstellt und in der Datenbank gespeichert, wenn alle für die Verarbeitung relevante Daten eingegeben wurden. Anhand der evaluierten Daten wird die Visualisierung angepasst.

**Fail**: Der Benutzer ist nicht berechtigt zum Erstellen der Einträge oder die Daten nicht vollständig sind.

**Fallback**: Die Daten sollen bei Eingabe auf dem Client validiert werden.

## 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.

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

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

**Fallback**: Stattdessen wird die Mobiltelefonnummer des Users eingegeben, damit die Kommunikation über ein Mobiltelefon sattfinden kann.

## Effiziente Datenübertragung

**Beschreibung**: Die Daten Übertragung zwischen Systemkomponenten muss effizient realisiert werden, so das im JSON – Format nur für Verarbeitung relevante Daten übertragen werden, und keine Bilder, Audio oder Video Dateien.

**Exit**: Es wird geringer Datenvolumen zwischen Komponenten transferiert. Die Daten zur Visualisierung der Informationen werden lokal auf dem Client – Gerät gespeichert und abgerufen.

**Fail**: Datenübertragung wird stakt beeinträchtigt oder abgebrochen.

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

## Auswertung der Bodendaten

**Beschreibung**: Die aufgehobenen Daten werden auf dem Server analysierte und darauf aufbauend werden individuelle Szenarien zum Ackerbau für Benutzer erstellt. Dazu sollen auch informationen zur aktuellen Wetterbedingungen bezogen werden. Die rechenaufwändigen Algorithmen werden dadurch nicht auf dem Client realisiert und beeinträchtigen nicht die Performance des Endgeräts.

**Exit**: Die Daten werden ausgewertet und zur weiteren Verarbeitung an den Client geschickt.

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

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

## Effiziente Erstellung interaktiver Anleitungen

**Beschreibung**: Es Werden dynamisch personalisierte Anleitungen 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.

**Fail**: Der Algorithmus arbeitet fehlerhaft oder die lokalen Daten werden zu lange geladen oder können nicht gelesen werden. Dadurch wird der Client stakt belastet und die Anleitungen werden nicht angezeigt.

**Fallback**: Eine in textueller Form gestaltete Anleitung anzeigen. Auf die interaktive Visualisierung wird verzichtet.

## 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.

**Exit**: 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.

## Erstellung der Anleitungen für Mobiltelefone

**Beschreibung**: Falls ein Benutzer kein Smartphone besitz, soll er die Empfehlungen in vereinfachter Form zeitnah auf seinem Mobiltelefon per SMS erhalten.

**Exit**: Aktuelle Informationen werden 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 wird verzichtet. Die Helfer informieren die Ackerbauer vor Ort informieren.

## Didaktische Darstellung der Informationen

**Beschreibung**: Die Informationen sollen didaktisch sinnvoll Visualisiert oder in anderen Formen dargestellt werden, damit auch die analphabeten Benutzer sie nachvollziehen können.

**Exit**: Die dargestellten Informationen sind klar dargestellt und sind für alle Stakeholder verständlich.

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

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

## Fazit

Zu den wichtigsten POCs gehören unter anderem die 8.4, 8.5, 8.9, da diese die Kernfunktionalitäten des Systems beschreiben und den wesentlichen Teil der Entwicklungszeit in Anspruch nehmen werden.

# Methodischer Rahmen

Im Folgenden werden die Vorgehensmodelle analysiert, im Rahmen derer der Entwicklungsprozess des Systems abgewickelt wird.

## Auswahl des methodischen Rahmens

Da das System sich mit einer komplexen Domäne beschäftigt, ist wichtig die Aufgaben der Benutzer detailliert zu analysieren und den Verwendungszweck zu verstehen. In diesem Sinne eignet sich optimal  **Usage centered design** als methodischer Rahmen. Beim Usage centered design wird Funktionalität des Systems in den Vordergrund gestellt. Damit ein hoher Grad der Usability eines interaktiven Systems erreicht wird, werden bei diesem Ansatz die Aufgaben der User und Ziele dieser Aufgaben im gegebenen Nutzungskontext analysiert und spezifiziert. Dies erleichtert vor allem das Verständnis vom Verwendungsweck des Systems für Entwickler und verbessert die Kommunikation mit den Benutzern.

Anderseits liegt der Fokus des Systems zum größten Teil auf Wissensvermittlung für die bestimmte Stakeholder-Gruppe. Aus dieser Perspektive ist das Vorgehen nach Usage centered design nicht geeignet, da es sich mehr auf die Benutzung des Systems konzentriert. Das geplante System wird von den Benutzern mit extrem unterschiedlichen Merkmalen (Helfer und Ackerbauer in den Entwicklungsländern) benutzt. Daher ist es wichtig die Benutzermerkmalle(Aufgaben, Ziele, Wissenstand, Fähigkeiten…) der verschieden Stakeholder zu berücksichtigen. Der **User centered design** – Ansatz, bei dem die Benutzermerkmalle im Fokus der Konzeption der Entwicklung stehen, sorgt dafür, dass die Anforderung der verschiedenen User erfüllt werden und dadurch die Gebrauchstauglichkeit des System erhöht wird.

Aus der Analyse folgt, dass im Rahmen des Projekts eine geeignete Kombination aus **Usage centered design** und **User centered design** zur Entwicklung eines gebrauchstauglichen Systems verwendet wird. Auf diese Weise wird erreicht, dass sowohl der Verwendungszweck des Systems detailliert verstanden wird, als auch die unterschiedlichen Anforderung und Merkmale der User berücksichtigt werden.

## Auswahl des Vorgehensmodells

In diesem Kapitel werden die Vorgehensmodelle abgewogen.

Usability engineering lifecycle

Das Modell von Deborah Mayhew stellt ein komplexes und umfangreiches Werkzeug dar, mit dem man den Entwicklungsprozess der gebrauchstauglichen Systeme in wesentlichen Aktivitäten gestalten kann. Dieses skalierbare Vorgehensmodell hat eine klar detaillierte Struktur mit Iterationen in allen Stufen der Entwicklung. Durch die Anforderungsanalyse im ersten Prozess-Bestandteil werden sowohl User als auch Ihre Aufgaben modelliert. Somit vereint Usability engineering lifecycle in sich die Ansätze von Usage centered design und User centered design. Anzumerken ist jedoch, dass das Modell sehr Umfangreich ist und einen gewissen Zeitaufwand fordert.

DIN EN ISO 9241 Teil 210

Das Vorgehen nach ISO 9241 Teil 210 ist ebenso skalierbar und iterativ. Es bietet vordefinierte Struktur, ist aber im Vergleich zu Usability engineering lifecycle übersichtlicher und ermöglicht eine frei Wahl der Techniken. Aus diesem Grund ist dieses Modell weniger Zeitaufwendig.

Szenario based usability engineering

Das Szenario based usability engineering ist ein aufwändiges narratives Vorgehen, das auf den Szenarien basiert, die das menschliche Handeln beschreiben und modellieren. Es ist ebenso ein iterativer Prozess, ist aber nicht auf die Komplexität des Projekts skalierbar und dadurch entsteht ein unangemessener Aufwand bei Erstellung der Szenarien.

Discount usability engeniering

Discount usability engeniering ermöglicht einen möglichst schnellen und kostengünstigen Entwicklungsprozess. Doch wegen fehlenden Kernaktivitäten der Entwicklung interaktiver Systeme wie Anforderungsanalyse oder Evaluation kann mit dieses Modell hohe Usability nicht erreicht werden.

## Fazit

Für dieses Projekt kommt hauptsächlich ein Vorgehensmodelle in Frage – Usability engineering lifecycle. Das Modell ist ziemlich aufwändig und braucht längere Einarbeitungszeit, bietet jedoch konkrete effektive Techniken zum Vorgehen an. Da die Zeit im Rahmen des Projekts eine sehr knappe Ressource ist, muss das Modell auf die Projektgröße skaliert werden. In den Entwicklungsphasen werden bestimmte schritte angepasst und Techniken aus anderen Vorgehensmodellen verwendet, die ähnliche Ergebnisse liefern. Somit wird der Entwicklungsprozess nicht strikt in einem Vorgehensrahmen stattfinden, sondern es wird eine Kombination von Techniken, Methoden und Vorgehensmodellen verwendet, was im Kontext des Systems und des Entwicklungsprozess durchaus sinnvoll ist.

# Kommunikationsmodell

## Deskriptives Modell

## Präskriptives Modell

# Architektur

In diesem Kapitel wird die Architektur des Systems skizziert und einzelne Komponenten werden erläutert. Das System ist eine verteilte Anwendung bietet damit größere Performance und Skalierung. Es ist besonders wichtig im Kontext des Projekts, da durch das verteilte System z.B. der Client keine aufwändigen Anfragen zu den externen Diensten schicken muss.

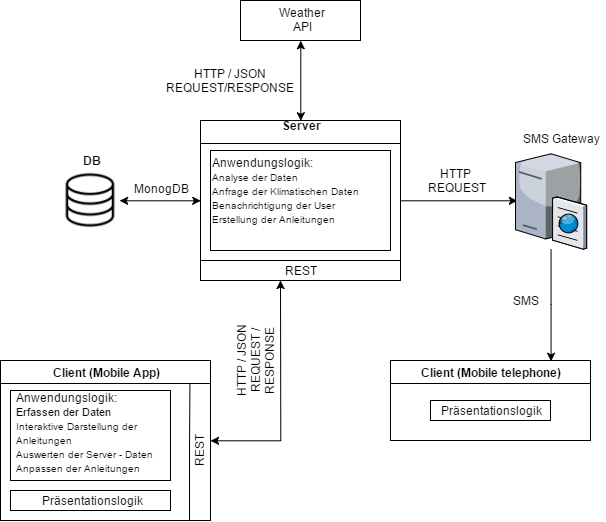


Abb. 1 Architekturmodell

**Client-Server Kommunikation**

Die einzelnen Komponenten Kommunizieren nach dem Client-Server Paradigma, wodurch eine zentralisierte Datenhaltung und strikte Verteilung der Anwendungslogik erreicht wird.

**Clients**

Als Clients agieren mobile App und Mobiletelefon, die jeweils für die Präsentationslogik zuständig sind. In Mobilen App wird auch Teil der Anwendungslogik ausgeführt.

**Server**

Server ist eine Node.js App, die vor allem rechenaufwändige Aufgaben übernimmt und mit externen Diensten kommuniziert. Zur Speicherung der Einträge und User wird vom Server die Datenbank verwaltet.

**SMS Gateway**

Externer Dienst wird Benutzt, um SMS an die Clients zu verschicken. Dabei wird ein API benutz, um die Nachricht an Gateway zu schicken, die dann an den Client weitergeleitet wird.

**Zielplattform**

Als Zielplattform wurde Android Smartphone gewählt. Android hat den größten Anteil am Weltmarkt[1]. Mit diesem Betriebssystem werden aber auch sehr günstige Geräte angeboten, was gerade für die Entwicklungsländer ein großer Vorteil ist. So gibt es z.B. ein Smartphone in Indien schon für 3,28 Euro angeboten[2].

**Datenformat**

Als Datenformat wird JSON bevorzugt. Dieses Format eignet sich optimal für eine effiziente Datenübertragung zwischen Komponenten. Kommunikation mit externen Diensten wird ebenso mit JSON abgewickelt.

**Protokolle**

Für die grundlegende Übertragung wird HTTP verwendet, da es mit Node.js und REST optimal arbeitet und JSON-Daten problemlos übertragen werden. Benachteiligung des Clients wird mit FCM (Firebase Cloud Messaging) realisiert, was sowohl Android als auch IOS Plattformen unterstützt.

**Synchrone und Asynchrone Kommunikation**

Die Speicherung von Daten in DB und auch Erstellen und Schicken der Anleitungen zum Client sollte Synchron ablaufen. Abfragen von Wetterdaten und Benachrichtigung der Clients wird dagegen asynchron realisiert.

# Quellenverzeichnis

Günstigstes Smartphone: <https://www.welt.de/wirtschaft/webwelt/article152327016/Das-ist-das-guenstigste-Smartphone-der-Welt.html> (11.05.17)

Android Marktanteil: <https://de.statista.com/statistik/daten/studie/184332/umfrage/marktanteil-der-mobilen-betriebssysteme-in-deutschland-seit-2009/> (11.05.17)