



Fachhochschule Köln  
Cologne University of Applied Sciences

**Dokumentation (MS3.2)**  
im Rahmen der Veranstaltung  
„Entwicklungsprojekt interaktive Systeme“

Sommersemester 2015

**„Spiel 21“ (Twenty-One)**

**Projektmitglied**

Eduard König

**Projektbetreunde Professoren**

Prof. Dr. Gerhard Hartmann

Prof. Dr. Kristian Fischer

**Projektbetreuer**

Franz Jaspers, B. Sc.

Sheree Saßmannshausen, B. Sc.

## Inhaltsverzeichnis

1. Einleitung .....	5
1.1 Problemraum .....	5
1.2 Regeln .....	5
1.3 Lösungsansatz.....	6
2. Zielhierarchie .....	7
2.1 Strategische Ziele (langfristig) .....	7
2.2 Taktische Ziele (mittelfristig).....	7
2.3 Operative Ziele (kurzfristig).....	8
3. Marktrecherche (related-works).....	8
3.1 Wetterdienste .....	9
3.2 Nachrichtenaustausch (Messenger) .....	9
3.3 Navigation (GPS-Datenübertragung) .....	9
3.4. Alleinstellungsmerkmale .....	10
4 MCI Vorgehensweise .....	10
4.1 Wahl des Vorgehensmodells .....	10
4.1.1 User Centered Design .....	11
4.1.2 Usability-Methoden .....	11
4.2 Nutzungskontextanalyse .....	13
4.2.1 Stakeholderanalyse .....	14
4.2.2 User Profiles.....	16
4.2.3 Personas .....	17
4.2.4 Szenarien.....	17
4.3 Anforderungsermittlung .....	18
4.3.1 Produktperspektive .....	18
4.3.2 Einsatzkontext.....	18
4.4.2 Funktionale Anforderungen .....	19
4.4.3 Nichtfunktionale Anforderungen (qualitative) .....	19
4.5 Hierarchische Task Analysis (HTA) .....	20
5. Kommunikationsmodell .....	22
5.1 Deskriptives Modell .....	22

5.2 Präskriptives Modell.....	23
6. Risiken.....	24
6.1 Akzeptanz .....	24
6.2 Entwicklung.....	24
6.3 Datenschutz .....	24
6.4 Werbung .....	25
6.5 Technisch.....	25
6.6 Projektintern .....	25
7. Spezifikation der PoCs .....	26
7.1 Testen der Netzwerkstruktur .....	26
7.2 Nachrichtenaustausch (Chat Applikation).....	27
7.3 Standort.....	27
7.4 Wetter .....	27
7.5 Matchmaking .....	28
8. WBA-Modellierungen.....	29
8.1 Architekturdiagramm.....	29
8.2 Systemarchitektur .....	29
x.x Server.....	30
x.x Datenbank .....	30
x.x Endgeräte .....	31
8.2.3 Middleware .....	32
8.2.4 Drittanbieter .....	32
8.3 Fazit .....	33
9 Prototypen UI .....	33
9.x Anforderungsanalyse .....	33
9.x Papierbasierte Prototypen .....	36
9. x Erkenntnisse aus dem Evaluationsprozess .....	41
10. Installationsanleitung .....	42
Anhang .....	43
Fragebogen Spiel 21 .....	43
Auswertung des Fragebogen .....	45

Personas .....	46
Szenarien .....	47
Quellenverzeichnis .....	49
Literatur .....	49
Internetquellen .....	49

# 1. Einleitung

Das Konzept dient der Entwicklung der Anwendung „Spiel 21“, dies ist eine spezielle Variante des Basketballs. Die Anwendung soll als verteiltes System entwickelt werden und den Anforderungen der speziellen Domäne entsprechen. „Spiel 21“ stellt eine Anwendung dar, welche den Benutzern hilft andere Benutzer in der Umgebung zu finden und diese dann auf ein Spielchen einzuladen. Dieses Dokument beschreibt das Konzept, Prozess- sowie die Systemdokumentation der geplanten Anwendung. Dies umfasst die detaillierte Beschreibungen welche im Laufe des Projekts vorgenommen wurden. Außerdem werden während des Projektes die beiden wichtigen Module „Mensch Computer Interaktion“ und „Webbasierte Anwendungen 2“ verwendet und in der Praxis angewandt.

## 1.1 Problemraum

Einige Menschen halten sich gerne im freien auf und unternehmen zusammen verschiedene sportliche Aktivitäten, wie beispielsweise Laufen, Fußball oder auch Basketball. Und hierbei gibt es eine spezielle Variante des Basketballs „Spiel 21“. Dies ist ein Wettkampf Spiel, man kennt es vielleicht noch aus dem Sportunterricht. Es wird im freien zu zweit oder maximal bis zu vier Spielern gespielt, die Regeln werden weiter unter erläutert.

Das Spiel kann aber nur dann gespielt werden, wenn der öffentliche Basketballplatz nicht belegt ist und auch nur bei einem schönen Wetter. Es wird nicht gespielt wenn es regnet, weil:

- Der Basketball ständig nass ist und es dauert bis dieser richtig trocken ist.
- Die Spieler können schnell ausrutschen und Verletzungen hinzuziehen.
- Man findet eher keine Mitspieler beim regen.

Es ist nicht einfach Freunde oder Wildfremde in eigener Nähe zu finden, wenn man gerade in eine neue Ortschaft gezogen ist und diese dann zum Duell herauszufordern. Deshalb wäre es schön Gleichgesinnte beispielsweise nach dem baldigen Feierabend auf ein schnelles Spielchen einzuladen bzw. zu einem Duell herausfordern.

## 1.2 Regeln<sup>1</sup>

„Spiel 21 ist ein Basketball Wettkampf, der zu zweit oder maximal mit vier Spielern durchgeführt werden kann (bei mehr Spielern werden die Wartezeiten zu lange!). Für das Spiel benötigt man einen Basketball sowie einen Basketballkorb. Im Folgenden wird das Spiel mit zwei Spielern (A und B) erklärt.

A wirft den Ball von der Freiwurflinie aus auf den Korb. Trifft er, erhält er zwei Punkte und kann sogleich einen neuen Wurf tätigen. Trifft er nicht mehr, fängt B den Ball in der Luft oder nachdem er maximal einmal auf dem Boden aufgeprallt ist. B muss nun genau von dieser Position einen Korbwurf tätigen. Trifft er, erhält er einen Punkt (zwei Punkte erhält man nur, wenn der Ball von der Freiwurflinie aus geworfen wird) und darf den nächsten Wurf von der Freiwurflinie aus machen etc. [s. Abb.]

---

<sup>1</sup> Quelle: <http://www.sportunterricht.ch/lektion/spielen/spielen80.php> (Zugriff am: 17.06.2015)

Airball: Berührt ein Ball beim Wurf weder das Brett noch den Korb, so ist dies ein Airball. Airball bedeutet, dass der Spieler wieder bei 0 anfangen muss. Wer hat zuerst 21 Punkte, hat gewonnen!

**Für Fortgeschrittene:** Der letzte Wurf muss hinter der Dreipunktlinie aus erfolgen!

Natürlich wird das Spiel nicht nur in dieser Variante gespielt. Es ist jederzeit möglich die Regeln auf das Können oder den Raum etc. anzupassen.“

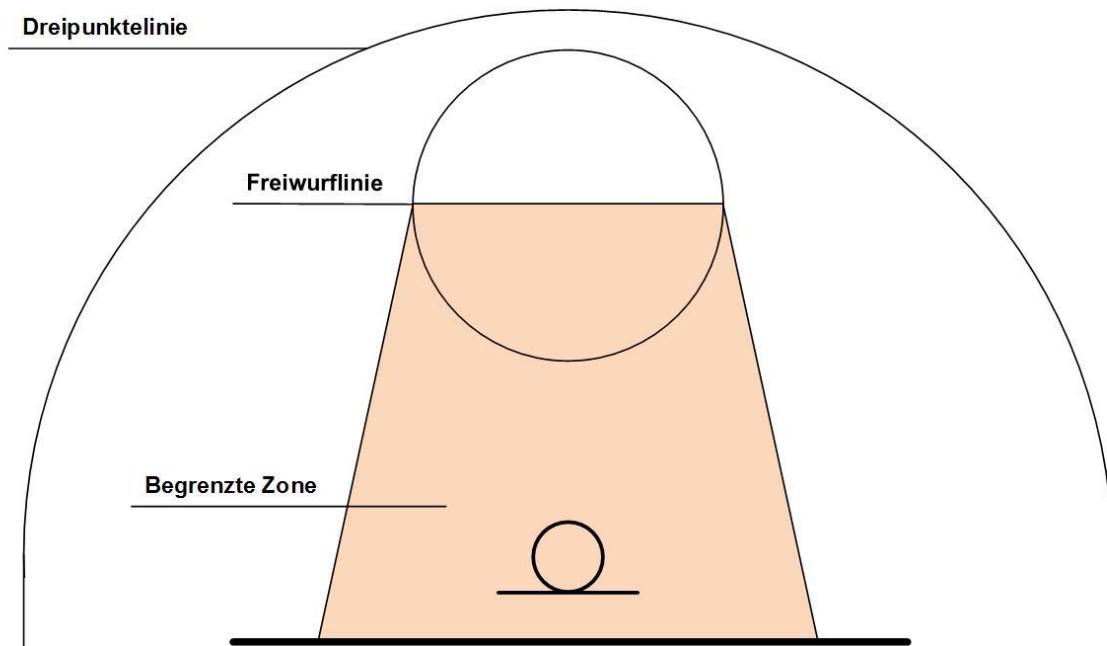


Abbildung: Schematische Darstellung

### 1.3 Lösungsansatz

Als Lösungsansatz wird ein System entwickelt, welches den Benutzern ermöglicht auf eine einfache Weise andere Benutzer zu finden und sich zu einem Spiel verabreden. Dabei werden die Wetterdaten von einem Wetterdienst abgegriffen und ausgewertet. Die andere Möglichkeit ist es die spielenden Benutzer zu protokollieren und für die anderen Benutzer sichtbar machen. Damit andere Benutzer auch wissen, dass zu gegebenen Zeitpunkt der Basketballplatz belegt ist und kann nur auf der anderen Hälfte des Spielfeldes gespielt werden.

Es soll das Versenden von Nachrichten unterstützen umso die Kommunikation zwischen den Benutzern zu erleichtern. Des Weiteren sollte das System die Benutzer zu den gewünschten Basketballplatz navigieren können.

Später erhebt die Anwendung Daten von den Benutzern nach dem Spielende, also die Spielergebnisse und kann diese für ein Matchmaking verwenden um zukünftig bessere bzw. gleichstarke Rivalen zu finden.

## 2. Zielhierarchie

Aus der formulierten Problemstellung werden verschiedene Ziele aufgestellt und näher erläutert. Im Folgenden erfolgt die Gliederung in strategische, taktische und operative Ziele, dies sind Maßnahmen ohne technischen Hintergrund.

### 2.1 Strategische Ziele (langfristig)

- 1) Eine Gemeinschaft für das „Spiel 21“ aufzubauen sowie die Gewinnung neuer Benutzer.
- 2) Eine digitale Plattform für das „Spiel 21“ etablieren.
- 3) Für Benutzer, die optimale Spielgegebenheiten kenntlich machen.
- 4) Die Manipulation von Spielergebnissen reduzieren um somit eine bessere Akzeptanz zu schaffen oder damit sich das Spiel besser etabliert.
- 5) Den beschränkten Basketballplatz optimal nutzen (max. 8 Spieler auf dem Basketballplatz).
- 6) In Kontakt treten mit anderen Benutzern.

### 2.2 Taktische Ziele (mittelfristig)

- (1) Benutzer durch virtuelle Erfolge, sogenannte Achievements belohnen umso mehr Motivation zwischen den Benutzern zu schaffen und dadurch die Gemeinschaft zu stärken.
- (1/6) Wildfremde Benutzer aus der Umgebung durch die Anwendung kennenzulernen und mit denen später in Kontakt treffen zu können.
- (2) Die Akzeptanz des Systems erhöhen.
- (2) Die Geschwindigkeit der Arbeit mit dem neuen System maximieren.
- (3) Wettervorhersage in der öffentlichen Umgebung angezeigt und einen Ratschlag gegeben ob es optimal wäre in der gewünschten Zeit zu spielen.
- (4) Durch die gegenseitige Kontrolle der Benutzer, kann man die Manipulation unterdrücken.
- (5) Durch das System sollte es möglich sein die Anzahl der Benutzer auf den öffentlichen Basketballplätzen zu identifizieren.
- (6) Den Benutzern erlauben Nachrichten zu versenden, um so mit anderen Benutzern in Kontakt treten zu können.
- (6) Die Zielvoraussetzung ist ein Korb und ein Ball pro Duell, im Duell sind jeweils 2 bis 4 Spieler.

### 2.3 Operative Ziele (kurzfristig)

- (1) Die Anwendung soll durch das Matchmaking sich positiv auf die Gemeinschaft auswirken.
- (1) Durch die Recherche kann festgestellt werden, wie viele Besitzer in der Domäne ein Smartphone mit Internetzugang haben.
- (2) Die Ausführung der notwendigen Schritte einer Funktion und die benötigte Zeit dafür minimieren.
- (2) Die Navigation sollte den Benutzern helfen einen bestimmten Basketballplatz zu finden.
- (2) Die Benutzer sollten den Weg zum Basketballplatz finden durch die Navigation oder durch ihr vorhandenes Wissen über die Umgebung. (Auto oder Fuß?)
- (3) Durch die Einbindung der Wetterinformationen kann das System genau vorhersagen ob ein Duell ausgetragen werden kann oder nicht, denn beim Regen wird nicht gespielt.
- (3/5) Durch die Anmeldung der Benutzer im System kann man bei den gewählten Basketballplatz die Vakanz einsehen (der reale Tatbestand bleibt jedoch unberührt).
- (3) Durch die Eingabe des Benutzers kann bestimmt werden, ob der Spieler ein Profi oder ein Anfänger ist.
- (4) Durch das Versenden von negativen Feedback über den Benutzer, kann man die Manipulation der Spielergebnisse reduzieren.
- (6) Den Benutzern eine Chatfunktion anbieten, um so den Nachrichtenaustausch zu gewährleisten.
- (6) Durch Befragungen in der Domäne kann festgestellt werden, wie viele Benutzer bereit wären mit Wildfremden Benutzer(n) zu spielen.
- (6) Die Benutzer sollten sich einigen beim Duell, wer den Basketball zum Platz bringt.

Um den Informationsfluss durch die Anwendung zu verdeutlichen kann ein Kommunikationsmodell entwickelt werden. Hierbei sollte sich ebenfalls eine Vorstellung der Nutzungsmotivation ergeben.

### 3. Marktrecherche (related-works)

Um die Alleinstellungsmerkmale der Anwendung zu bestimmen, wurde eine Marktrecherche durchgeführt, umso die positiven wie auch die negativen Teilaspekte der Konkurrenzprodukte zu kennen.

Im Rahmen des Systems werden die Anwendung im Google und Apple Applikation Store betrachtet, dabei wird nicht auf die Webanwendungen weiter eingegangen, da diese keine Relevanz für das jetzige System darstellen. Im Mittelpunkt stehen die Konkurrenzprodukte die eine Übermittlung von Wetterinformationen, Navigation sowie eine



Kommunikationsmöglichkeit zwischen den Benutzern bieten. Dies sind einige der Teilaspekte, die eine wichtige Rolle für das Projekt spielen.

### 3.1 Wetterdienste

Ein Wetterdienst wie beispielsweise [wetter.com](https://play.google.com/store/apps/details?id=com.wetter.androidclient)<sup>2</sup> und Wetter Online<sup>3</sup> bietet eine 7-Tage-Vorhersage für die verschiedenen Endgeräte sowohl aus dem Google als auch aus dem Apple Store. Natürlich wird hierbei auch das aktuelle Wetter angezeigt. Hierbei können die Orte entweder durch Namen, PLZ oder über die GPS Ortung erfolgen. Auch Regenwarnungen sind für Deutschland in der Anwendung vorhanden. Wetter.com nutzt automatisch denn zur Verfügung gestellten Netzwerk, entweder den mobilen oder einen stabile WLAN Datentarif.

Deutliche Nachteile von der Anwendung gibt es keine, es erfüllt jedoch nur eine Teilfunktionalität der Nutzerproblematik. Des Weiteren bietet der Dienst „wetter.com“ eine kostenlose Schnittstelle für nicht-kommerzielle Projekte. Daher ist es gut geeignet, zur Implementierung in das System als Drittanbieter.

### 3.2 Nachrichtenaustausch (Messenger)

Es gibt diverse Nachrichten-Apps dem sogenannten Messenger. Eines der Beliebtesten davon ist WhatsApp<sup>4</sup> und der Messenger<sup>5</sup> von Facebook. Der größte Vorteil solcher Messenger sind wohl die geringen Gebühren zum Vergleich für SMS/MMS. Zusätzlich bietet der Messenger eigene Profilverwaltung mit einem Namen und Profilbild. So können die Benutzer auf eine einfache Art und Weise in Kommunikation treten und Textnachrichten versenden.

Einige Nachteile solcher Dienste sind meist die AGB's, da diese den Datenschutz nicht genau wahrnehmen. Bei WhatsApp handelt es sich um eine kostenlose Testversion für ein Jahr, danach muss ein Abo abgeschlossen werden. Auch hier erfüllen die Anwendungen eine Teilfunktion für die Nutzerproblematik als solches nur die Kommunikation zwischen den Benutzern. Leider sind solche Lösungen von den Messenger eher weniger geeignet für das Projekt. Als Alternative kann man eigene Nachrichtenaustauschfunktion einbauen, wie beispielsweise einen Chat.

### 3.3 Navigation (GPS-Datenübertragung)

Die Navigationsprodukte unterscheiden sich meist in kostenlosen und kostenpflichtigen Variante, sowie offline und online Navigation. Einige der kostenpflichtigen sind beispielsweise NAVIGON select<sup>6</sup> und TomTom GPS-Navigation Traffic<sup>7</sup>. Diese bieten sehr viele Vorteile und sehr viele Funktionen wie beispielsweise die Echtzeit-Verkehrsinformationen. Einer der größten Nachteile ist wohl die fehlende Schnittstelle für das

---

<sup>2</sup> Quelle: <https://play.google.com/store/apps/details?id=com.wetter.androidclient> (Zugriff am: 20.04.15)

<sup>3</sup> Quelle: <https://play.google.com/store/apps/details?id=de.wetteronline.wetterapp> (Zugriff am 20.04.15)

<sup>4</sup> Quelle: <https://play.google.com/store/apps/details?id=com.whatsapp> (Zugriff am 20.04.15)

<sup>5</sup> Quelle: <https://play.google.com/store/apps/details?id=com.facebook.orca> (Zugriff am 20.04.15)

<sup>6</sup> Quelle: [https://play.google.com/store/apps/details?id=com.navigon.navigator\\_select](https://play.google.com/store/apps/details?id=com.navigon.navigator_select) (Zugriff am: 20.04.15)

<sup>7</sup> Quelle: <https://play.google.com/store/apps/details?id=com.tomtom.gplay.navapp> (Zugriff am 20.04.15)

angedachte System, deshalb wird hier nicht mehr weiter auf die kostenpflichtigen Produkte eingegangen. Eine Alternative sind die kostenlosen Produkte wie beispielsweise Maps<sup>8</sup> und MAPS.ME Offline Karte & Routen<sup>9</sup>. Bei Maps handelt es sich um eine online Navigation und ist unter Umständen mit hohen Datenaufkommen verbunden. Einer der größten Vorteile ist wohl die offene Schnittstelle für die Programmierer. Bei MAPS.ME dagegen handelt es sich um eine offline Navigation und hat den Vorteil dass es keine weiteren Karten geladen werden müssen. Einer der Nachteile wäre dann die Speichergröße der Anwendung bzw. der Karten die man vorher Installieren müsste und die fehlende Schnittstelle zu der Anwendung. In den Rahmen des Projekts kommt daher nur die online Navigation in Frage.

### 3.4. Alleinstellungsmerkmale

Es gibt viele Sportanwendungen auf dem Markt, doch aktuell gibt es keine Anwendung, die speziell für Basketballfreunde und in diesen Fall für das „Spiel 21“ gedacht ist. Anhand der erläuterten Konkurrenzprodukte kann man die verschiedenen Teilfunktionen betrachten. Viele der Anwendungen bieten vornerein gute Teilfunktionen zur Erfüllung des Nutzungsproblems, jedoch keinem identischen oder ähnlichen Konzept für das System. Als indirekte Konkurrenz gilt, jedoch aufzuführen die sozialen Netzwerke, wie beispielsweise Facebook oder Ähnliches. Hierbei lässt es sich nicht konkret sagen wie viele, wo und welche Gemeinschaften für das „Spiel 21“ existieren.

## 4 MCI Vorgehensweise

Nachdem im Folgenden die Zielhierarchie und Marktanalyse durchgeführt worden ist, werden nachstehende methodologische Herangehensweise im Bereich Mensch Computer Interaktion vorgestellt um für die Benutzer die Gebrauchstauglichkeit gewährleisten zu können. Die Überlegungen dienen später für das zu wählende Vorgehensmodell.

### 4.1 Wahl des Vorgehensmodells

Die Anwendung sollte vor allem Dingen für den Freizeitbereich entwickelt werden und sollte daher möglichst gut sein. Hierbei sollte einer hoher Maß an Gebrauchstauglichkeit aufgewiesen wird, damit es von der Gemeinschaft akzeptiert wird. Daher sind die Benutzer sehr wichtig für den Entwicklungsprozess.

Also sind vor allen die Merkmale der Benutzer, wie beispielsweise die Aufgaben und Ziele, Nutzungskontext, Wissenstand, Fertigkeiten und Fähigkeiten als zweckmäßigen Ausgangspunkt für das Konzept und die Entwicklung des interaktiven Systems zu betrachten. Hier werden die verschiedenen Ansätze gegeneinander abgewogen und später im Fazit festgelegt.

---

<sup>8</sup> Quelle: <https://play.google.com/store/apps/details?id=com.google.android.apps.maps> (Zugriff am 20.04.15)

<sup>9</sup> Quelle: <https://play.google.com/store/apps/details?id=com.mapswithme.maps.pro> (Zugriff am 20.04.15)

### 4.1.1 User Centered Design

Dies ist keine konkrete Arbeitsmethode, sondern eine Grundhaltung, die sich darin äußert, dass der Benutzer in allen Phasen der Entwicklung mit einbezogen wird. Dadurch wird bestrebt, eine hohe Gebrauchstauglichkeit (usability) zu erreichen.

Gründe die für die Einbeziehung der Benutzer sind meistens die verschiedenen Perspektiven. Der eine Entwickler hat nicht die gleichen Perspektiven wie ein Benutzer selbst. Deshalb wird ein Entwickler auch nicht schaffen die Perspektiven der Benutzer zu verbessern ohne seine Hilfe. Ein anderer Grund ist die Notwendigkeit, Produktqualität und Gebrauchstauglichkeit in einem Gesamtprozess als wichtige, aber eigenständige Aufgaben und Prozesse zu betreiben. So sollte der Benutzer mehr Aufmerksamkeit auf seine Tätigkeit werfen und nicht etwa von dem System abgelenkt werden.

Bei der Entwicklung der interaktiven Systeme gehört ein iteratives Vorgehen mit den folgenden Aktivitäten:

- Konzipieren (design).
- Erstellung der dazugehörigen Gestaltungsmöglichkeiten (prototyp)
- Evaluation der Prototypen (evaluation) und später wieder Design

Die Bewertung dieser Grundhaltung ist gut, da der Focus auf den Benutzern selbst liegt und nicht etwa auf dem Nutzen des Systems. Des Weiteren gibt es hierzu verschiedene Rahmenbedingungen, die das Projekt maßgeblich beeinflussen.

### 4.1.2 Usability-Methoden

Es stehen mehrere Modelle, mit unterschiedlichen Methoden und Techniken zur Auswahl, die den benutzungszentrierten Gestaltungsprozess unterstützen. Folgende Methoden kommen für das Projekt zum Einsatz:

#### Discount Usability Engineering

Dieser Ansatz stammt von Jakob Nielsen<sup>10</sup>. Dies stellt eine schnelle und kostengünstige Überprüfung zur Verbesserung der Gebrauchstauglichkeit mit den folgenden drei Techniken dar.

Bei den Szenarien (storyboard), stellt sich als Ziel, potenzielle Probleme mit der Usability zu identifizieren. Für das Projekt stellt sich als Nachteil heraus, dass keine Vollständigkeit gewährleistet wird und mit großen Zeitaufwand verbunden ist, deshalb werden diese verworfen.

Des Weiteren gibt es die Gestaltungslösungen (paperbased prototypes), diese haben eine geringe Vorläufigkeit und dient nur Auflistung der Anforderungen. Für das Projekt stellt es insgesamt eine gute Lösung dar, da hier die Daten iterativ gesammelt werden und somit wird ein erster visueller Eindruck über die Anwendung gewonnen. Hierbei existiert noch eine

---

<sup>10</sup> Quelle: <http://www.nngroup.com/articles/discount-usability-20-years/> (Zugriff am: 27.04.15)

vereinfachte Technik (think aloud), das „laute-Denken“. Diese Technik eignet sich auch sehr gut für das Projekt, da man die wichtigsten usability Probleme identifizieren kann.

Außerdem gibt es den letzten Punkt, Evaluation durch Heuristik (heuristic evaluation). Nach Nielson, können hier alle erfolgreich angewendet werden. Aber dies kann nicht von nur einer Person durchgeführt werden, da man sonst nie alle Fehler in einem System entdecken würde. Für das Projekt stellt dieses Modell eine mögliche Lösung dar, dennoch würde die Ausarbeitung der Szenariantypen viel Zeit kosten und hinsichtlich diese Ressource viele Nachteile in anderen Bereichen des Projekts mitbringen. Dennoch aufgrund der finanziellen und zeitlichen Rahmenbedingungen wird die Vorgehensweise des Discount Usability gewählt.

### **Benutzerprofile**

Die Benutzerprofile stellen eine Liste von Merkmalen, samt Merkmalausprägungen zur Verfügung welche für die Entwicklung benutzt werden. Der Entwickler kann sich so eine bessere Perspektive über weitere Benutzer schaffen. Hierbei wird iterativ geprüft und dokumentiert, wie der Benutzer mit der Anwendung zurechtkommen würde.

Für das Projekt stellt dieses Modell eine mögliche Lösung dar. Es bietet sich an, diesen Prozess also iterativ und über die gesamte Entwicklungsphase laufen zu lassen. Dennoch würde auch hier die Ausarbeitung und die empirische Datenerhebung viel Zeit kosten, beispielsweise durch Interviews, Umfragen, Feldstudien, Beobachtungen, etc. Dies könnte man bei dem Projekt begrenzen, beispielsweise auf Umfragen der Benutzer um somit die Zeitressource zu sparen.

### **DIN-Norm EN ISO 9241 (Teil 110/210)**

Die DIN-Norm mit dem Teil 110 stellt die Grundsätze der Dialoggestaltung dar, dies könnte bei dem Projekt sich als sehr nützlich Erweisen, beispielsweise bei der Entwicklung der Anwendung im Design Bereich. Diese sind in sieben Anforderungen gegliedert und stellen keine Einschränkungen mit den Projekt dar. Interessant sind hierbei die folgenden Grundsätze, wie beispielsweise die Selbstbeschreibungsfähigkeit, dies erlaubt das Anzeigen von Zustandsänderungen im System. Steuerbarkeit, in jedem Eingabefeld soll es eine Möglichkeit geben die letzte Eingabe rückgängig zu machen. Und Fehlertoleranz erkennt das System selbst die fehlerhafte Eingabe automatisch und teilt dem Benutzer mit, beispielsweise die falsch eingegebene E-Mail Adresse. Leider können aus zeitlichen Gründen nicht alle Anforderungen der DIN-Norm berücksichtigt werden.

Der Teil 210 stellt den Prozess zur Gestaltung gebrauchstauglicher interaktiver Systeme dar. Hierbei handelt es sich nur um einen Rahmenwerk. Nachteil für dieses Projekt ist, dass der Teil keine Techniken oder Methoden zur Verfügung stellt. Dies sollte möglichst mit anderen Modellen zur Vervollständigung beitragen. Dennoch bietet die Norm eine stabile Grundlage und sollte auf jeden Fall für das Projekt berücksichtigt werden.

## **4.2 Zusammenfassung**

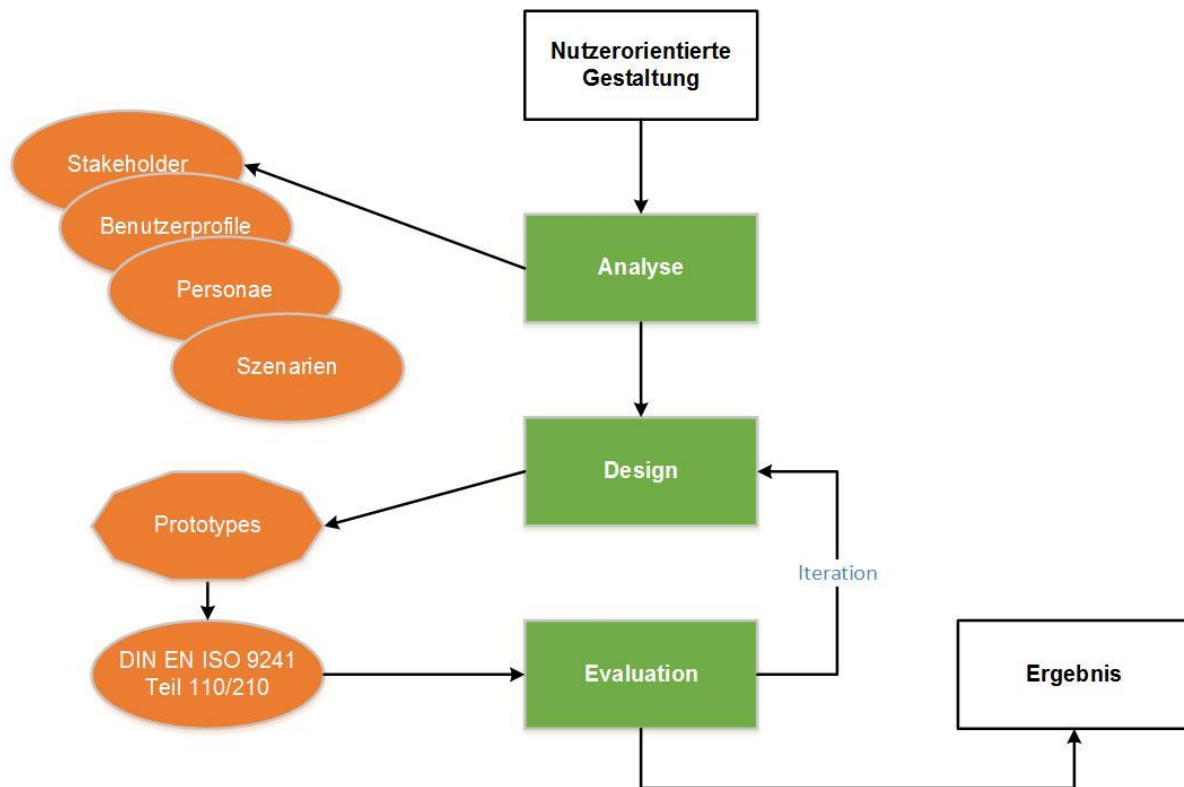


Abbildung: MCI Vorgehensweise

Eine benutzerorientierte Methodik sollte bereits in der Entwicklungsphase wichtige Lösungen aus der Benutzersicht liefern und später damit ein technisches Rahmenwerk schaffen. Als Grundhaltung sollte User Centered Design genommen werden mit der Kombination aus Discount Usability und der DIN-Norm EN ISO 9241 (Teil 110/210). Hier soll durch die Kombinatorik die Gebrauchstauglichkeit allgemein anerkannt werden. Für das Projekt wurden bereits die Evaluationsmethoden abgewogen und die besten Lösungsansätze für das Projekt genannt. Durch die Iteration sollte man später zu einem Ergebnis gelangen, welches ein hohes Maß an Gebrauchstauglichkeit aufweist um später auf ein Ergebnis zu gelangen.

## 4.2 Nutzungskontextanalyse

„Nutzungskontext sind Benutzer, Arbeitsaufgaben, Ausrüstung (Hardware, Software und Materialien) sowie die physische und soziale Umgebung, in der das Produkt genutzt wird.“  
[ISO 9241-210]

Um mehr Verständnis in der Domäne zu bekommen, wurde eine Internetrecherche durchgeführt. Leider ist die Domäne so speziell, dass es kaum Informationen dazu gibt. Die einzige Erkenntnis, die man daraus ziehen konnte sind die vielfältigen Regeln bei dem „Spiel 21“. **Leider war es nicht möglich, Spieler aus der Szene zu sprechen.**

**Um dennoch Daten von Personen zu bekommen, werden Fragebögen verteilt, um so die Erfahrungen und die Meinungen der Personen als Ergebnisse auswerten zu können [s. Anhang].**

Dabei ist der Entwurf des Fragebogens sehr wichtig für die Qualität und die Beschaffenheit der Aussagen. Die Ergebnisse der befragten fließen später in die Benutzermodellierung mit ein und begründen somit die wesentlichen Unterschiede zwischen den Benutzern.

Im Grunde wird hier die physiologischen Merkmale, Wohnortgegebenheiten sowie das Interesse an Freizeit abgefragt. Leider kann man zwischen einen Anfänger und Fortgeschrittene nicht unterscheiden, dies sollte im Verlauf der Analyse nicht auserhalb gelassen werden.

Die Analyse von Benutzer und Kontext hat ergeben, dass das Alter der befragten Personen im Durchschnitt bei 26,56 Jahren liegt und die meisten Personen waren männlich [s. **Anhang**]. Es stand nur ein begrenzter Befragtenkreis zur Verfügung, deshalb kamen die meisten befragten aus Gummersbach und sind Studenten. Hier ist aber interessant zu betrachten, dass die meisten Personen entweder länger als 5 Jahre in den selben Ort wohnen oder sind ganz neu hinzugezogen. Dabei spielt die Transportmöglichkeit eine wichtige Rolle und die meisten besitzen ein Auto.

Was die technischen Merkmale angeht, so hat jeder ein internetfähiges Endgerät, beispielsweise ein Smartphone und besitzt hierbei eine Flatrate. Des Weiteren hat schon jeder dritte nach Freizeitmöglichkeiten im Internet gesucht.

Im Durchschnitt haben die befragten Personen 26,15 Stunden in der Woche an Freizeit und sind sehr aktiv, deshalb kommt selten langweile auf. Wenn es um die Sportart geht, so gehen hier die Meinung weit auseinander und nur wenige beschäftigen sich mit Basketball, die Meisten trainieren lieber im Fitnessstudio. Überraschenderweise gibt es Leute die speziell das Basketballspiel 21 kennen, so haben 15 von 32 Personen mit ja geantwortet.

Die gewonnenen Erkenntnisse werden eingesetzt und dienen damit zur Ermittlung der Anforderungen und tragen zur neuen Lösungen bei. Im folgende werden zunächst die Benutzer identifiziert.

#### 4.2.1 Stakeholderanalyse

Stakeholder sind „Einzelperson oder Organisation, die ein Anrecht, einen Anteil, einen Anspruch oder ein Interesse auf ein bzw. an einem System oder an dessen Merkmalen hat, die ihren Erfordernissen und Erwartungen entsprechen.“ [ISO 9241-210]

Im Falle der Stakeholderanalyse konnten folgende Stakeholder anhand der Faktoren aus der DIN Norm infiziert werden:

- Benutzer: *Spieler.*
- Städte/Gemeinde: *Ansehen der Gemeinde.*
- Anwohner: *Lärmbelästigung.*
- Datenschützer: *sind besorgt um die missbräuchliche Datenverarbeitung.*

Im Weiteren werden die Stakeholder begutachtet und weiter veranschaulicht anhand der Merkmale.

Bezug Benutzer	Objektbereich der Beziehung
Anrecht	Privatsphäre
Anteil	Speicherung von Endergebnissen. Auswertung von Statistiken. Erstellung / Einladung eines Duells. Unterstützung der Anwendung durch Mundpropaganda.
Anspruch	Erhalt von Wettertipps. Erhalt von Basketballplatzinformationen, beispielsweise über den Zustand belegt (Ja/Nein). Erhalt von passenden Rivalen.
Interesse	Zum Basketballplatz zu gelangen, navigieren lassen.
Erfordernisse und Erwartungen	Nicht beim Regen zu spielen. Wissen, wie viele Benutzer auf den Basketballplatz sind. Mit anderen Benutzern in Verbindung treten, bspw. Chat. Herausfinden, wo der nächste Basketballplatz in der Nähe ist. Passende Spieler zu finden. Ein geeignetes Endgerät um die Anwendung nutzen zu können.

Tabelle: Bezug Benutzer

Bezug Städte/Gemeinde	Objektbereich der Beziehung
Anrecht	Ansehen der Gemeinde
Anteil	
Anspruch	
Interesse	Eine Gemeinschaft entstehen zu lassen, die das Ansehen der Umgebung bzw. des Stadtteils fördert.
Erfordernisse und Erwartungen	Steigerung des Freizeitangebotes. Friedlicher Ablauf.

Tabelle: Bezug Städte/Gemeinde

Bezug Anwohner	Objektbereich der Beziehung
Anrecht	Privatsphäre, beispielsweise keine Abbildung der nächstgelegenen Häuser
Anteil	
Anspruch	
Interesse	Sicherheit der Umgebung.

	Keine Lärmbelästigung.
<b>Erfordernisse und Erwartungen</b>	Sicherheit der gespeicherten Daten.

Tabelle: Bezug Anwohner

Bezug Datenschützer	Objektbereich der Beziehung
<b>Anrecht</b>	Privatsphäre
<b>Anteil</b>	
<b>Anspruch</b>	
<b>Interesse</b>	Sicherheit der Datenverarbeitung.
<b>Erfordernisse und Erwartungen</b>	Sicherheit der gespeicherten Daten.

Tabelle: Bezug Datenschützer

#### 4.2.2 User Profiles

Nachdem die Stackholder bekannt sind, müssen nun diese detaillierter charakterisiert werden. Hierbei werden die Benutzer in Jugendliche und Erwachsene unterteilt, da diese nicht die gleichen Merkmale aufweisen.

BENUTZER	Jugendlicher	Erwachsener
<b>Arbeitsaufgaben</b>		
<ul style="list-style-type: none"><li>- Prüfung der Wetterdaten in Kombination mit der Verfügbarkeit von den Basketballplätzen</li><li>- Herausforderung der anderen Benutzer</li><li>- Eintragen der Spielergebnisse</li></ul>		
<b>Ausrüstung</b>		
Hardware:	Smartphone mit Internetzugang und GPS Verbindung	
Software:	die Anwendung „Spiel 21“	
Materialien:	Basketballplatz, Basketball	
<b>Physische Umgebung</b>		
Alter:	14-18 Jahre	18-60 Jahre
Geschlecht:	männlich / weiblich	männlich / weiblich
Beförderung:	eingeschränkt (Fuß, Fahrrad)	uneingeschränkt (Fuß, Auto, o.ä.)
Körperlicher Zustand:	uneingeschränkt	uneingeschränkt
<b>Soziale Umgebung</b>		
Berufliche Status:	Schüler, Auszubildender	Erwerbstätig o. ä.
Herkunft:	uneingeschränkt	uneingeschränkt



Religion:	uneingeschränkt	uneingeschränkt
Psychologische Umgebung		
Geistiger Zustand:	uneingeschränkt	uneingeschränkt
Grundkenntnisse		
Umgang mit Smartphone:	Grundkenntnisse (Downloaden von Anwendungen, Installation und Konfiguration) erforderlich	
Motivation zur Nutzung		
<ul style="list-style-type: none"><li>- Wildfremde Leute kennenlernen</li><li>- Abfragen der Wetterinformationen</li><li>- Abfragen von nächstgelegenen Basketballplätze</li><li>- Abfragen von der Verfügbarkeit des Basketballplatzes</li></ul>		

*Tabelle: Nutzungskontextanalyse und deren Merkmale*

### 4.2.3 Personas

Bei den Personas handelt es sich um prototypische Benutzer und es stellt nur eine Maske dar, welche ihre unterschiedlichen Ziele, Verhalten und deren Eigenschaften darstellt. Diese Methodik wurde von den Alan Cooper eingeführt [**Cooper**].

Die Persona ist wichtig für spätere Produktdesign und stellt relevante Eigenschaften der Benutzer dar, deshalb ist es wichtig diese lebensecht wirken zu lassen. Im folgenden werden zwei Personas beschrieben, die für zwei unterschiedliche Benutzergruppen stehen [**s. Anhang**]. Während Markus ganz viel Freizeit hat und gerne was mit seinen Freunden unternimmt, arbeitet die Anna als Sachbearbeiterin in Vollzeit und hat daher weniger Freizeit.

Die Personas zeigen die wichtigen Eigenschaften und die Bedürfnisse der Benutzer in Hinblick auf das geplante System.

### 4.2.4 Szenarien

Um die Perspektive der Benutzer einnehmen zu können, wurden die Personas und Szenarien erstellt. Nun kann man aus deren Sicht diskutieren und aus deren Entwurf eine neue Lösung schlagen für das zukünftige System. Als erste Prototypen, können die Szenarien für das Projekt dienen. Im folgenden werden einige Problemszenarien und daraus die entstehenden User Needs beschrieben [**s. Anhang**].

Szenario 1: Geht hervor, dass es offensichtlich Personen gibt, die gerne den Tag sportlich abklingen lassen. Außerdem kommt Langweile auf durch die monotone Sportart. Die Anwendung könnte wildfremde Leute zusammen bringen und so eine neue Gemeinschaft entstehen lassen.

Szenario 2: Es gibt Leute, die sich über die Wettervorhersage nicht informieren und haben keinen Spaß haben beim schlechten Wetterbedingungen zu spielen. In der Anwendung könnten Warnung ausgegeben werden, beim schlechten Wetter. Außerdem gibt es Leute die

kein gutes Erinnerungsvermögen haben, dies könnte in der Anwendung unterstützt werden durch die persistente Speicherung.

Szenario 3 und 5: Es gibt offensichtlich Leute, die andere Leute für Basketballspielen begeistern wollen. Speziell die Jugendlichen kennen das Spiel 21 aus dem Sportunterricht. In der Anwendung könnten das unterstützt werden, dass man andere Leute zu dem Spiel einlädt und begeistert. Außerdem könnten die gesammelten Spielergebnisse und die laufenden Spiele für jeden sichtbar sein.

Szenario 4: Es gibt Leute, die relativ neu in der Umgebung sind und haben mangelnde Kenntnisse über die Umgebung selbst. Außerdem wollen einige sich Sportlich aktiv halten ohne gleich eine Mitgliedschaft in Vereinen oder sonstigen Einrichtung abschließen zu müssen. Die Anwendung könnte bei der Navigation den Benutzer unterstützen. Es sollten keine Verträge oder Verpflichtungen mit der Anwendung abgeschlossen werden

### **4.3 Anforderungsermittlung**

„Anforderung (requirement) – 1. Eine Bedingung oder Fähigkeit, die von einer Person zur Lösung eines Problems oder zur Erreichung eines Ziels benötigt wird. 2. Eine Bedingung oder Fähigkeit, die eine Software erfüllen oder besitzen muss, um einen Vertrag, eine Norm oder ein anderes, formell bestimmtes Dokument zu erfüllen.“ [IEEE 610.12-1990]

#### **4.3.1 Produktperspektive**

Es ist eine Anwendung „Spiel 21“ zu erstellen, dass die Planung und Durchführung von den Duellen sowie die Navigation zum bestimmten Basketballplatz unterstützen soll. Wetterinformationen werden ausgewertet und an die Benutzer weiter geleitet sowie weitere Information über die Belegung des Basketballplatzes. Hierbei wird ein Endgerät mit dem Android Betriebssystem benötigt und die passende Anwendung sollte auf dem System installiert sein. Es wird kein spezielles Fachwissen für die mobilen Endgeräte in der Domäne benötigt.

#### **4.3.2 Einsatzkontext**

Für das Projekt wird das System vorerst nur deutschlandweit eingesetzt als zu testende Anwendung, um die mangelnde Akzeptanz auszuschließen und um letztendlich die Ressourcen zu sparen. Folgende Ressourcen sollten betrachtet werden: Kosten für die Übersetzer, Zeit für die mehrsprachige Programmierung und die Relevanz der speziellen Domäne. Des Weiteren ist eine zentrale Datenhaltung vorgesehen um die Wartung des Systems zu erleichtern.

Zielgruppe des Produkts sind ausschließlich die Benutzer der Anwendung „Spiel 21“ selbst, die als Gemeinschaft agieren. Die Anwendung bezieht sich auf den Freizeitbereich und hat keinen geschäftlichen Hintergrund. Deshalb wird bei dem Projekt in Erwägung gezogen, das

Schalten von Werbung um sich dadurch zu finanzieren, dies wird weiter bei den Risiken verfolgt. Im Folgenden werden für die Anwendung „Spiel 21“ die Anforderungen identifiziert und erläutert.

#### 4.4.2 Funktionale Anforderungen

- F10** Der Benutzer muss die Möglichkeit haben sich im System identifizieren.
- F20** Der Benutzer soll die Wetterinformationen von seinem Bestimmungsort abrufen können.
- F30** Der Benutzer kann sich über die vorhandenen Basketballplätze im System informieren.
- F40** Der Benutzer kann andere Benutzer zum Duell herausfordern.
- F50** Das System soll Benutzern Routen zum Basketballplatz bieten.
- F60** Der Benutzer sollte mit anderen Benutzern in Kontakt treten können.
- F70** Der Benutzer kann den Verlauf der gespielten Spiele ansehen.
- F80** Der Benutzer kann die Spielergebnisse eintragen und diese einsehen.
- F90** Das System bietet die Möglichkeit in der Organisation des Spiels den Basketballträger zu bestimmen.
- F100** Das System soll den Benutzer automatisch passende Rivalen vorschlagen.
- F110** Das System soll den Benutzer automatisch warnen, wenn die Wahrscheinlichkeit auf Regen sehr hoch ist bei einem bevorstehenden Duell.
- F120** Das System sollte die Spielergebnisse der Benutzer persistent speichern.
- F130** Das System sollte den Standort des Benutzers lokalisieren.

#### 4.4.3 Nichtfunktionale Anforderungen (qualitative)

- Q10** Die Anwendung soll eine hohe Gebrauchstauglichkeit aufweisen.
- Q20** Die Anwendung soll stets über die aktuellen Daten verfügen bis zum nächsten Spiel.
- Q30** Gute Navigierbarkeit durch das GPS, soweit in der GPS Verarbeitung möglich ist.
- Q40** Die erhobenen Daten über die Benutzer sollten anonymisiert behandelt werden.
- Q50** Die Manipulation von Spielergebnissen reduzieren.

Weitere Anforderungen können im Laufe des Projektes ermittelt werden.

## 4.5 Hierarchische Task Analysis (HTA)

Auf das Projekt bezogen nimmt man die Aufgabe bzw. Absicht das Basketballspiel 21 zu spielen und betrachtet diese, um in Anschluss die Möglichkeiten für das System zu betrachten. Dieser Ansatz wurde von Shepherd definiert [s. **Shepherd**].

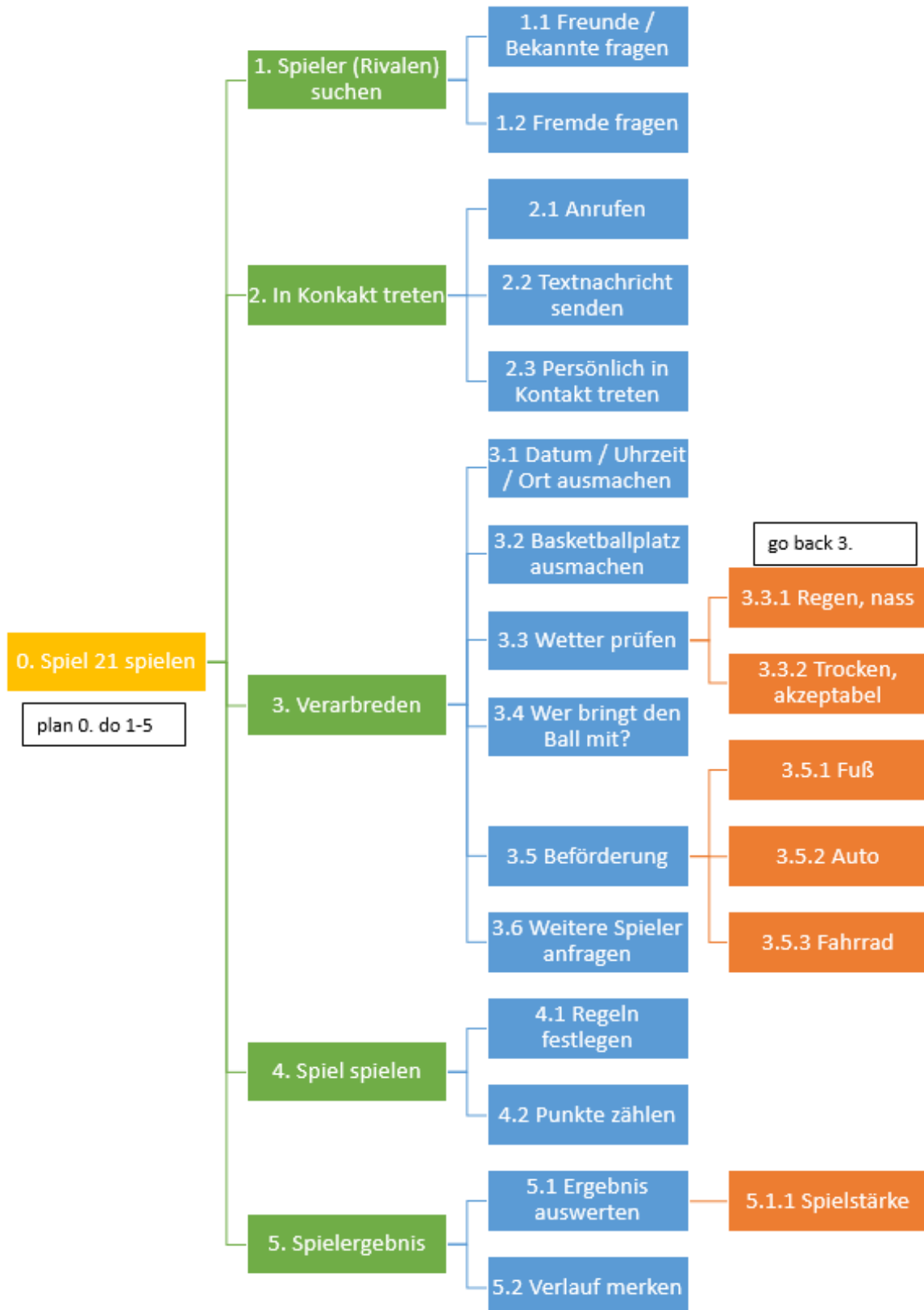


Abbildung: HTA-Dekomposition für den Task: Spiel 21 spielen

## 0. Spiel 21 spielen

1. Spieler (Rivalen) suchen
  - 1.1 Freunde / Bekannte fragen
  - 1.2 Fremde fragen
2. In Kontakt treten
  - 2.1 Anrufen
  - 2.2 Textnachrichten senden
  - 2.3 Persönlich in Kontakt treten
3. Verabreden
  - 3.1 Datum / Uhrzeit / Ort ausmachen
  - 3.2 Basketballplatz ausmachen
  - 3.3 Wetter prüfen
  - 3.4 Wer bringt den Ball mit?
  - 3.5 Beförderung
    - 3.5.1 Fuß
    - 3.5.2 Auto
    - 3.5.3 Fahrrad
  - 3.6 Weitere Spieler anfragen
4. Spiel spielen
  - 4.1 Regeln festlegen
  - 4.2 Punkte zählen
5. Spielergebnis
  - 5.1 Ergebnis auswerten
    - 5.1.1 Spielstärke
  - 5.2 Verlauf merken

Dies ist ein Beispiel und zeigt aus der HTA entstandenen Dokumentationen, wie die Durchfügung einer HTA mit den Aufgabendiagramm (Abbildung) und die dazugehörige Planliste.

### Pläne:

- Plan 0: mach 1 – 2 – 3 – 4 – 5 in dieser Reihenfolge
- Plan 1: mach 1.1 – 1.2 in beliebiger Reihenfolge
- Plan 2: mach 2.1 – 2.2 - 2.3 in beliebiger Reihenfolge
- Plan 3: mach 3.1 – 3.2 – 3.3 – 3.3.2 – 3.4 – 3.5 in dieser Reihenfolge,  
wenn gewünscht: mach 3.6, wenn 3.3.1 dann EXIT oder 3.1
- Plan 4: mach 4.1 – 4.2 in dieser Reihenfolge
- Plan 5: mach 5.1 – 5.2 in beliebiger Reihenfolge

Hier wurde die Aufgabe auf immer kleiner Teilaufgaben heruntergebrochen um ein detailliertes Verstehen von den Nutzungsarten und den Stufen zu erlangen.

## 5. Kommunikationsmodell

Die Anwendung „Spiel 21“ wirkt als verteiltes System mit verschiedenen Teilkomponenten. Im Folgenden wird der Ablauf der Kommunikation genauer erläutert, ohne auf die technischen Details einzugehen.

### 5.1 Deskriptives Modell

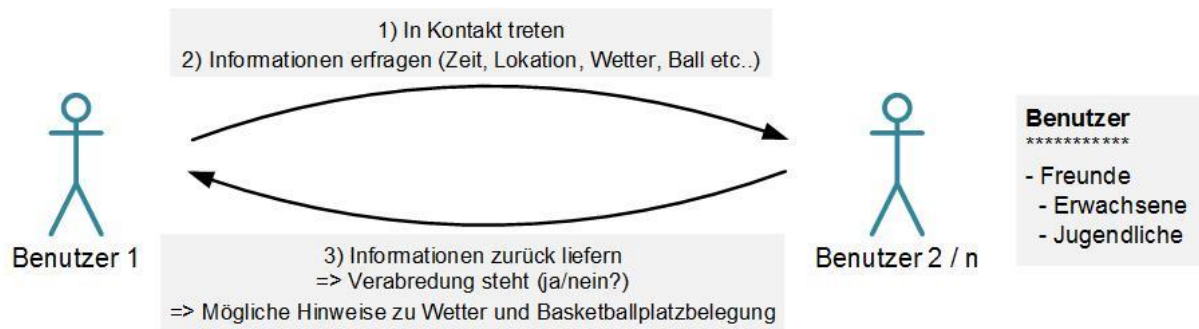


Abbildung 1: deskriptives Kommunikationsmodell

Die Abbildung 1 zeigt die momentane Situation zwischen den Benutzern. Derzeit verläuft die Kommunikation zum größten Teil über die Sprachebene, indem der Benutzer 1 mit dem Benutzer 2 in Kontakt tritt. Sie tauschen verschiedene Informationen untereinander aus. Beispielsweise wird erfragt:

- Wann und wo wird sich getroffen?
- Eventuell wird auch das momentane Wetter geprüft, regnet es?
- Wer bringt den Basketball zu den verabredeten Platz mit?

Des Weiteren bleibt jedoch die Problematik bestehen, ob ein Basketballplatz belegt ist oder nicht. Die Benutzer können verschiedene Personen sein, sind jedoch in den meisten Fällen nur Freunde aus der Umgebung die sich gut kennen.

## 5.2 Präskriptives Modell

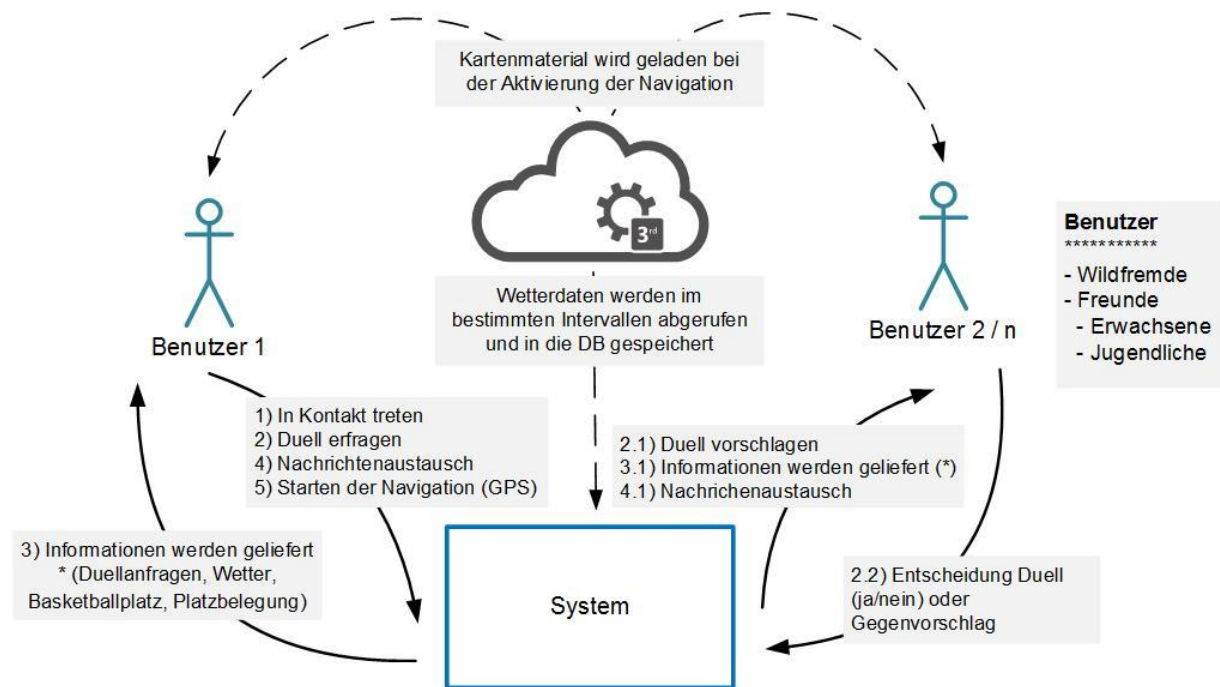


Abbildung 2: präskriptives Kommunikationsmodell

In der Zukunft sollte möglichst der größte Teil der Kommunikation über das System erfolgen. Ein Benutzer kann in Kontakt mit dem System treten und nach einem Duell anfragen. Daraufhin werden die möglichen Informationen, wie beispielsweise andere Duellanfragen, Wetterinformationen, Basketballplatz geliefert. Anhand der Informationen kann der Benutzer 1 mit anderen Benutzern, die sich vorher im System registriert haben in Kontakt treten. Hierbei findet ein Nachrichtenaustausch statt.

Sollte sich ein Match ergeben, so werden dem Benutzer 2 möglichen Duelle vorgeschlagen und die passenden Informationen geliefert, wie beispielsweise Uhrzeit, Datum, Lokation, etc.. Der Benutzer kann anhand dieser Informationen selbst entscheiden, ob er mit dem Vorschlag einverstanden ist oder einen Gegenvorschlag unterbreiten.

Hierbei kann die Kommunikation zwischen den verschiedenen Benutzern in verschiedene Richtungen laufen. Das präskriptive Kommunikationsmodell sollte nur einen Fallbeispiel zeigen, wie so was in der Anwendung ablaufen könnte.

Die Benutzer können verschiedene Personen sein, so auch Wildfremde, die man vielleicht nicht aus der Umgebung kennt. Dadurch könnten sich neue Freundschaften entstehen und Gemeinschaft bilden.

Ein Drittanbieter liefert die Wetterdaten, diese werden in bestimmten Intervallen abgefragt und daraufhin in der Datenbank gespeichert. Diese sind sowohl für das spätere Matchmaking als auch für Regenwahrscheinlichkeit von Bedeutung. Aber auch das Kartenmaterial wird von einem Drittanbieter zur Verfügung gestellt und in die Anwendung implementiert. Das Kartenmaterial wird erst bei Aktivierung der Navigation geladen, und zwar bei dem Benutzer selbst.

## **6. Risiken**

In den meisten Fällen gefährden die Projektrisiken den Erfolg des Projektes. Diese sind aber auch wichtig für die Planung und dürfen nicht nur mit technischer Natur in Verbindung gebracht werden. Ein wichtiges Mittel zur Bewertung von potenzieller Risiken stellt die Risikoanalyse dar, hiermit können diese frühzeitig erkannt und vermieden werden. Dies ist auch die Grundlage für die Projektplanung und wird im Allgemeinen zur Identifikation und Bewertung eingesetzt. Die typischen Risiken bestehen meist aus der mangelnden Akzeptanz der Benutzer und die spät entdeckten Anforderungen im System.

### **6.1 Akzeptanz**

Sollte die Anwendung von wenigen Benutzern genutzt werden, so entsteht ein hohes Risiko für die Anwendung selbst. Durch die fehlenden Daten, die zu qualitativ hochwertigen Statistiken und Rankingergebnissen führen könnten, kann kein Matchmaking stattfinden und somit keine geeigneten Rivalen für das „Spiel 21“.

Eine weitere Folge ist, dass die Gemeinschaft in dem Freizeitbereich erst gar nicht entsteht. Ein anderes Szenario wäre die Unzuverlässigkeit der Anwendung. Hier würde der Benutzer im schlechtesten Fall die Anwendung überhaupt nicht mehr benutzen.

### **6.2 Entwicklung**

Allein die Entwicklung des Systems verschlingt viel Zeit, Geld und Ressourcen. Dies sollte generell bei jedem internen Datenverarbeitungssystem in Betracht gezogen werden. Aber auch, die neuen und wenig bekannten Technologien bringen eine Herausforderung mit sich, beispielsweise die Einarbeitungsphase.

Eine mögliche Lösung ist, ein Konzept zu entwickeln, wo die ganzen Problematiken, Techniken und Methoden festgehalten werden um die Entwicklung des Systems zu ermöglichen oder eher eine bessere Erkenntnis davon zu ziehen.

### **6.3 Datenschutz**

Die Anwendung sollte in der Lage sein, den Benutzer zum gewünschten Basketballplatz navigieren zu können. Dabei greift die Anwendung auf die GPS Funktionalität des Smartphones zu und ermittelt somit die aktuelle Position des Benutzers. Hierbei könnte der Benutzer die Speicherung der Routen erwarten und dadurch „die totale Überwachung“ befürchten. Aber auch durch Sicherheitslücken im System oder auch in der Anwendung können Dritte an personenbezogene Daten herankommen und für eigene Zwecke missbrauchen.

Sollte man auf die GPS Funktionalität verzichten, dann kann das Navigationsfeature nicht benutzt werden. Die Sicherheitslücken entstehen durch den Entwicklungsprozess der Anwendung, wenn die Sicherheitsvorgaben nicht ausreichend berücksichtigt werden.



Eine mögliche Lösung ist das GPS auszuschalten, denn jeder Benutzer kann dies auf dem Smartphone steuern. Das Abspeichern und Verarbeiten der Informationen wird bei der Anwendung auf das Minimum reduziert. Bei den Sicherheitslücken im System oder auch der Anwendung müssen Updates durchgeführt werden um diese mit einem Patch bzw. Update zu schließen.

## **6.4 Werbung**

Viele der Benutzer meiden die Anwendungen mit einblendender Werbung, da diese oft als störend empfunden werden. Doch leider lässt es sich im einen „kostenlosen“ bzw. im Freizeitbereich wie so oft nicht vermeiden. Da dies meist eine finanzielle Grundlage ist für die Anwendung selbst, damit beispielsweise die Serverkosten gedeckt werden.

Eine mögliche Lösung ist es dem Benutzer zu erlauben die Werbung in den Einstellungen auszuschalten oder auf eine „Pro-Version“ umzusteigen mit einem geringen Entgelt. Die Werbung sollte auch nur dezent auf dem Bildschirm positioniert werden, also ohne nervige Pop-ups, wenn dafür Platz Vorhaben ist. Ohne dabei die Funktionalitäten der Anwendung zu beeinträchtigen. Ein weiterer Punkt, ist die Werbung mit im Kontext zu verwenden umso eine größtmögliche Annahme der Anwendung zu gewährleisten.

## **6.5 Technisch**

Auf dem Smartphone laufen verschiedene Anwendungen, die die Batterie des mobilen Endgerätes belasten. Besonders durch das Aktivieren der GPS Funktionalität, wodurch die Wetterinformationen und das Kartenmaterial für die Navigation geladen werden.

Eine Lösung ist es, den Benutzern zu erlauben durch die manuelle Eingabe ihren Standort auszuwählen, so können die Wetterinformationen empfangen werden, ohne auf die Navigationsfunktion zugreifen zu müssen. Dies ist für die Benutzer interessant, die sich in der Umgebung auskennen und wissen wo die jeweiligen Basketballplätze zu finden sind.

Außerdem kann ein Ausfall von Drittanbietern, die Funktionalität der Anwendung beeinträchtigen. So beispielsweise können keine Wetterinformationen abgerufen werden und auch keine Navigation.

Eine mögliche Lösung für diese Problematik wäre, dann die Daten des Wetterdienstes auf eigenen Server zwischen zu speichern und bei den Navigationsdienst auf einen alternativen Drittanbieter zu setzen.

## **6.6 Projektintern**

Durch die Bedrängnis der einzuhaltenden bevorstehenden Termine wird geradezu ein Zeitdruck aufgebaut. Außerdem könnte der Ausfall der Programmierer bzw. der Teammitglieder, beispielsweise durch eine Erkrankung o. ä. das Projekt zum Scheitern bringen.

## 7. Spezifikation der PoCs

Bei Proof of Concept (Machbarkeitsnachweis) lassen sich eventuell die Risiken ableiten und dieses müssen nicht nur ausschließliche der technischen Natur sein. Hier werden die Funktionalitäten des Systems geprüft und konkrete Bedingungen sowie konkrete Alternativen beschreiben und anschließen prototypisch umgesetzt.

### 7.1 Testen der Netzwerkstruktur

**Risiko:** Asynchrone Datenverarbeitung zwischen der Android Anwendung und Node.js Server. Anschließend sollten die Daten in die Datenbank gespeichert werden, beispielsweise ein einfaches Login.

**Ziel:** Es sollte eine Android Anwendung prototypisch geschrieben werden, welche mittels POST Methode die JSON Daten von der Android Anwendung an den Node.js Server sendet und diese dort in der Datenbank vergleicht, beispielsweise ein Login Bereich.

Folgende Schritte sollten ergriffen werden:

- Eine Android Anwendung sollte erstellt werden.
- Die Anwendung sollte auf einem Testgerät ausführbar sein.
- Der Server sollte JSON Daten empfangen und diese auch persistent speichern können.
- Allgemein: das Erlernen der Android Entwicklungsumgebung.

**Exit / Fail Kriterium:** Nur wenn der Ausbildungs- und und Lernaufwand in den Projektrahmen minimal gehalten werden und somit ein passables Ergebnis erzielt wird, gilt die Prüfung als bestanden.

**Alternative Fallback:** Leider führt hier kein Weg vorbei, man sollte sich die Zeit nehmen und sich mit den verschiedenen Entwicklungstools beschäftigen. Als Informatiker sollte man sich mit den neuen Technologien aneignen und sich weiter entwickeln. Die Ressourcen stehen frei zur Verfügung und die Wahrscheinlichkeit ist hier sehr gering.

**Status:** Die Entwicklungsumgebung wurde eingerichtet. Die ersten Schritte wurden ergriffen und es ist gelungen die Kommunikation zwischen den Node.js Server und der MongoDB Datenbank herzustellen und dabei die Daten auszutauschen in beide Richtungen. Anschließend wurde eine prototypische Android Anwendung geschrieben und auf dem Testgerät erfolgreich getestet.

**Termin:** sollte bis zum 08.05.2015 realisiert werden.

## 7.2 Nachrichtenaustausch (Chat Applikation)

**Risiko:** Es besteht zur Zeit keine Möglichkeit mit den Benutzern in Verbindung zu treten um unvorhersehbare Ereignisse zu diskutieren oder den Benutzer kennenzulernen. Man sollte die Netzwerkverbindung zwischen zwei Clients aufbauen.

**Ziel:** Schaffung offener Verbindung, die zum Nachrichtenaustausch zwischen den Benutzern dient, beispielsweise durch einen Chat.

**Exit / Fail Kriterium:** Nur wenn eine ständig offenen Verbindung zwischen den Client und einem anderen Client über den Server besteht und diese miteinander kommunizieren können, so gilt die Prüfung als bestanden. Ausnahme ist, wenn der Client nicht kommuniziert wie gewünscht mit dem Server und es findet kein Nachrichtenaustausch statt.

**Alternative Fallback:** Eine mögliche Alternative wäre eine Handynummer im Profil des angemeldeten Benutzers einzutragen und somit die persönliche Kommunikation zwischen den Benutzern erlauben. Dies würde dann als Pflichtfeldeingabe im Datenmodell benutzt werden.

**Status:** Es wurde eine Chat Anwendung mithilfe von Node.js und Socket.io realisiert. Später sollte die Anwendung in die Android Anwendung implementiert werden.

**Termin:** sollte bis zum 10.05.2015 realisiert werden.

## 7.3 Standort

**Risiko:** Die Benutzer verlieren das Interesse an dem System, wenn schon zu Beginn viele Abfragen gestellt werden, beispielsweise über den aktuellen Standort.

**Ziel:** Um für die Benutzern nicht unnötig mit Standortfrage zu belästigen, sollte eine die Lokalisierung so einfach wie möglich gehalten werden. Mittels GPS sollte der Standort des Benutzers ermittelt werden.

**Exit / Fail Kriterium:** Nur wenn die GPS Position mittels der Android Anwendung bestimmt werden kann, so gilt die Prüfung als bestanden, ansonsten greift man auf die Alternative zurück.

**Alternative Fallback:** Die Alternative sieht vor eine manuelle Eingabe des Standorts.

Status:

**Termin:** sollte bis zum 13.05.2015 realisiert werden.

## 7.4 Wetter

**Risiko:** Wenn die Spieler auf dem Basketballplatz spielen (Duell) und es fängt an zu regnen. Es ist vorher nicht bewusst den Spielern, dass es im Laufe des Tages regnen könnte. Würden die Spieler richtig die Wetterlage einschätzen können, so würden sie das Spiel verschieben.

**Ziel:** Wetterinformationen sollen die Benutzer vor Regen warnen.

**Exit / Fail Kriterium:** Wenn die Wetterinformationen bei den Benutzern in seiner Umgebung (manuelle Standorteingabe) korrekt als Warnungen ausgegeben werden können. Ausnahme ist eventuell die Anbindung der Schnittstelle bei den Drittanbietern.

**Alternative Fallback:** Es gibt mehrere Dienstleister am Markt die, die Wetterinformationen und die passenden APIs zur Verfügung stellen. Sollten die Wetterwarnungen nicht funktionieren kann man einfach die Informationen für die nächsten 7 Tage zur Verfügung stellen ohne die Warnungen.

**Status:** Das Vorgehen hat gezeigt, das die Daten durch einen HTTP Request durchgeführt werden können. Hier wird auf den „openweathermap<sup>11</sup>“ Wetterdienst zugegriffen und nicht wie etwa auf „wetter.com“, da man sonst zwei Wochen warten müsste, bis ein Antrag für die Schnittstelle gewilligt wird. Des Weiteren wird hier über XML und wie etwa über JSON kommuniziert, da die Fragmente und die ganzen Values unter Android Studio als XML dargestellt werden. Die Warnungen im Regenfall konnten leider noch nicht umgesetzt werden und deshalb gilt der PoC als fehlgeschlagen. Als Alternative wurde die Ausgabe von den nächsten 7 Tagen zur Verfügung gestellt.

**Termin:** sollte bis zum 10.05.2015 realisiert werden.

## 7.5 Matchmaking

**Risiko:** Ohne Matchmaking Algorithmus wird das System für den Benutzer uninteressant und bietet daher kaum einen Mehrwert in der Domäne.

**Ziel:** Implementierung der Bewertungsgrundlagen anhand der Spielergebnisse schaffen, dann mit Pseudodaten anreichern und testen. Matchmaking verknüpfen mit den Wetterinformationen und Basketballplätzen.

**Exit / Fail Kriterium:** Nur wenn der Algorithmus einwandfrei funktioniert und die Funktionalitäten der Benutzer bestmöglich erfüllt so gilt es als erfolgreich. Ausnahme ist, wenn der Algorithmus für jeden Benutzer, der die gleichen Ergebnisse liefert oder der Algorithmus bezieht alle Ergebnisse der Benutzer mit ein, auch diejenigen die nicht in der Umgebung leben. (Was ist der beste Radius für die Spielumgebung oder sollte man den selbst definieren können?)

**Alternative Fallback:** Es gibt kaum eine Alternative dafür, außer man macht es in einer kleinen Domäne, wo jeder Jeden kennt.

**Status:** Das Matchmaking wird zu diesen Zeitpunkt nur als Pseudocode entwickelt, da hier das Testen der Netzwerkstruktur Vorrang hat und muss zuerst als erfolgreich abgeschlossen werden um später die Implementierung im System zu gestatten.

**Termin:** sollte bis zum 20.05.2015 realisiert werden.

---

<sup>11</sup> Quelle: <http://openweathermap.org/api> (01.05.15)

## 8. WBA-Modellierungen

### 8.1 Architekturdiagramm

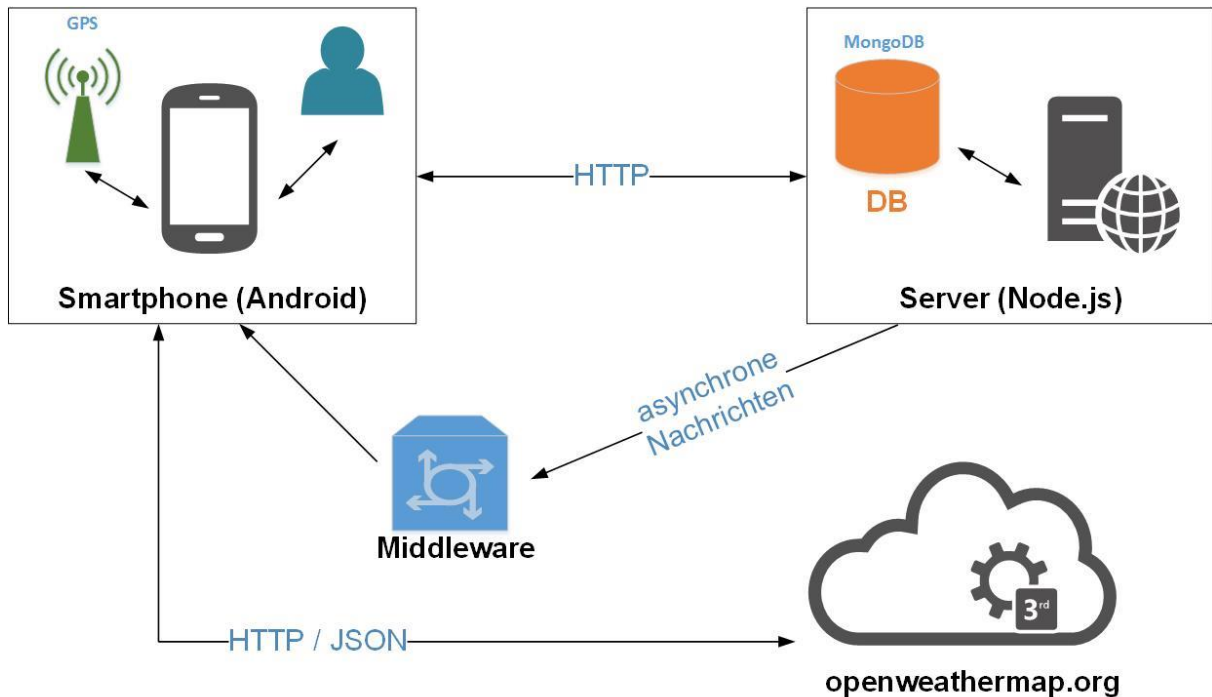


Abbildung 3: Systemarchitektur

### 8.2 Systemarchitektur

Die Systemarchitektur sollte über einen Server mit angebundener Datenbank kommunizieren. Der Client besteht aus einer Präsentations- und Anwendungslogik. Bei dem Endgerät spielt die GPS Funktion eine wichtige Rolle um beispielsweise die Wetterinformationen abrufen zu können oder sich zum Basketballplatz navigieren zu lassen. Hierbei wird auf die meteorologischen Wetterinformationen von einem Drittanbieter (openweathermap.org) zugegriffen. Des Weiteren sollte das System in der Lage sein, die Nachrichten zwischen den Benutzern in einer asynchronen Verbindung auszutauschen. Das ganze System sollte zusammenwirken um als eins dargestellt zu werden, obwohl es eine Menge voneinander unabhängiger Computer sind, so erscheinen diese für den Benutzer als ein einzelnes System. Im Folgenden beschreibt die Abbildung der Systemarchitektur, ohne in die Implementierungsdetails zu gehen.

## **x.x Server**

Auf dem Server sollen verschiedene Aufgaben durchgeführt werden, unter anderem das Matchmaking. Durch den Einsatz von Drittanbietern können Wetterinformationen abgefragt und weiter auf dem Server verarbeitet werden. Für das jetzige Projekt bietet sich die Implementierung von „node.js“<sup>12</sup> an. Dies ist eine Plattform, basierend auf JavaScript Laufzeitbibliotheken, welches einen schnellen Bau von Netzwerkanwendungen erlaubt. Ein weiterer Punkt, der bei der Entwicklung der Anwendung als geeigneter Aspekt erscheint ist die asynchrone Kommunikation bei Ein- und Ausgabe der Daten.

Im aktuellen Projekt werden die Daten über die REST Grundprinzipien versendet und empfangen [Tilkov]<sup>13</sup>. Hier kann die Anwendung POST Anfragen an die URI Ressourcen des Servers senden und anschließend wertet der Server die Anfragen aus. So werden beispielsweise bei der Registrierung die Daten im JSON Format abgelegt und anschließend wird die Client Anwendung darüber informiert ob der Vorgang erfolgreich war oder nicht.

### **x.x.x Datensicherheit**

Leider findet die derzeitige Kommunikation des Prototypen im Klartext über http

Daten unverschlüsselt auch in der Datenbank

HTTPS Anzustreben, verschlüsselte Form

### **x.x.x Datenschutz**

Google (Maps) amerikanische Firma unterliegt nicht den deutschen Datenschutzbestimmungen. Ob und welche Daten werden erhoben?

## **x.x Datenbank**

Für die persistente Datenhaltung wird die Datenbank „mongoDB“<sup>14</sup> verwendet. Diese bietet eine flexibles Datenmodell und erlaubt Daten in jeglicher Struktur zu speichern, darunter auch beispielsweise das JSON Repräsentationsformat. Dies ist ein kompaktes Datenformat in einer einfachen lesbaren Textform für Mensch und Maschine. Weiterhin bietet die Datenbank hohe Skalierbarkeit und Leistung. Die relationale Datenbank ist in diesen Fall eher kontraproduktiv, da diese nur eine eingeschränkte Skalierbarkeit bietet.

---

<sup>12</sup> Quelle: <https://nodejs.org/> (Zugriff am 21.04.15)

<sup>13</sup> Quelle: REST und HTTP: Entwicklung und Integration nach dem Architekturstil des Web

<sup>14</sup> Quelle: <https://www.mongodb.org/> (Zugriff am 21.04.15)

Im aktuellen Projekt verwaltet die Datenbank drei Collections. Die erste Collection „users“ ist für die Benutzer gedacht, welche die wichtigen Informationen von den Personen beinhaltet. Die zweite Collection „courts“ ist für die Basketballplätze gedacht, welche die wichtigen Informationen über die Plätze und deren Standort beinhaltet. Die letzte Collection „duel“ ist für die Spielergebnisse zuständig.

*Abbildung: Collection „users“*

*Abbildung: Collection „courts“*

*Abbildung: Collection „duel“*

### Vorläufiges Datenbankmodell

users	courts	duel
_id	_id	_id
gender	name	enemy
first	street	score
last	code	date
email	city	time
username	country	courts_id
pass	latitude	users_id
phone	longitude	
pic		
win		
lose		

## x.x Endgeräte

In der heutigen Zeit gibt es viele Technologien und auch viele verschiedene mobile Endgeräte mit den verschiedenen Betriebssystemen. Ein der beliebtesten ist ein Smartphone das einiges an Computerfunktionalität bietet als ein herkömmliches Mobilfunktelefon. Deshalb wird der Fokus der Entwicklungsarbeiten auf ein Smartphone gesetzt. Hierbei stellt sich die Frage zwischen den verschiedenen Betriebssystemen. Ein der beliebtesten auf dem Markt ist das Android OS aus dem Hause Google. Es hat zurzeit den größten Marktanteil gegenüber den anderen Systemen, wie beispielsweise iOS aus dem Hause Apple und Windows Mobile aus dem Hause Microsoft. Die Geräte mit dem Android Betriebssystem werden im Durchschnitt günstiger auf dem Markt angeboten als die Konkurrenz. Aufgrund dessen wurde auf der Benutzerseite entschieden, eine mobile Anwendung für das Smartphone mit Android Betriebssystem zu entwickeln.

### x.x.x Android Anwendung

Das Dilemma der Entwicklung für die mobilen Endgeräte besteht darin, dass es sehr viele verschiedene Geräte mit verschiedenen Auflösungen existieren. Deshalb sollte man so flexibel wie möglich sein und relative Größen definieren. Die Größen werden in dp (dop per inch) definiert, dies ist wichtig, da man die Elemente mit dem Finger anklickt und nicht wie etwa auf einem Computer mit der Maus.

Finger haben einen bestimmten Durchmesser, deshalb gibt es auch seit neuesten auch Richtlinien für die Metrics und Grids aus dem Hause Google<sup>15</sup>. Beispielsweise sollte man 48 dp (7-10 cm) Vorgabe beachten und mindestens 8 dp zwischen den einzelnen Elementen. Android bietet zwei verschiedene Themes an, dunkel und hell. Dunkel sollte man für die Einstellungen und hell sollte für das normale Lesen am Bildschirm verwendet werden, da dies durch das Lesen am Papier in unseren Köpfen geprägt ist. Das Konzept von Android sieht vor, keine Hardgecodeten Strings zu verwenden und sorgt dafür das man zentrale Stelle für die Texte hat. Dies schreibt auch die Internationalisierung für die Softwareentwicklung vor, abgekürzt mit i18n<sup>16</sup>. Für das Projekt wird dieser Punkt nicht berücksichtigt, dies wurde bereits in den Einsatzkontext beschrieben.

Beispielsweise wird bei der erstmaligen Ausführung der Anwendung, die Login Eingabemaske gestartet. Sollte sich der Benutzer schon mal angemeldet haben so wird dieser automatisch eingeloggt in das System. Ist der Benutzer neu im System, so muss er sich zuerst registrieren, dies geschieht über die Registrierungsmaske. Nach erfolgreichen Absenden der Daten wird der Benutzer zum Login Eingabemaske geleitet, dort kann er sich anschließend einloggen und wird darauf hin in das Navigationsmenü der Anwendung weitergeleitet.

[Code Schnipsel]

Wie werden die Nachrichten versendet?

Android-APK -> Server -> DB -> Server -> Android-APK

### 8.2.3 Middleware

Die Middleware stellt eine Komponente dar, die neben der verteilten Anwendung auch auf verschiedenen Systemen laufen kann. Bei der Middleware fällt die Wahl auf das „node.js“ Modul „express“ zurück. Dies ist ein kleines und flexibles „node.js“ Webframework, dessen Funktionsumfang sich mit ergänzenden Modulen je nach Bedarf erweitern lässt.

### 8.2.4 Drittanbieter

Um die Wetterinformationen zu bekommen, wird versucht auf die Drittanbieter der meteorologischen Dienste durch eine Schnittstelle zu greifen. Einige der Anbieter, wie beispielsweise „wetter.com“<sup>17</sup> bieten einer kostenlosen Programmierschnittstelle an, diese ist momentan auf 10.000 Zugriffe pro Monat limitiert. Die Anfragen an die Schnittstelle erfolgen über ein einfaches REST Format. Als Ausgabeformat wird JSON mit geringen Overhead und damit geringen Übertragungszeiten unterstützt. Die Systemkommunikation zwischen den Drittanbieter soll von dem Server ausgehen, dass die Anwendungslogik zum Abrufen der Wetterinformationen auf der Benutzerseite stattfindet.

<sup>15</sup> Quelle: <https://developer.android.com/design/style/index.html> (Zugriff am 22.05.15)

<sup>16</sup> Quelle: <http://www.debian.org/doc/manuals/intro-i18n/> (Zugriff am 22.05.15)

<sup>17</sup> Quelle: [http://www.wetter.com/apps\\_und\\_mehr/website/api/](http://www.wetter.com/apps_und_mehr/website/api/) (Zugriff am 21.04.15)



Aber auch das Kartenmaterial kommt von einem Drittanbieter, wie beispielsweise Google Maps<sup>18</sup>. Hier kann man für seine Mobilgeräte die Standortdienste und Karten hinzufügen. Dies basiert auf einer JavaScript API und ist von Google sehr gut auf der Webseite dokumentiert. Durch die GPS Funktion wird der Standort ermittelt und auch Wunsch die Route berechnet.

### 8.3 Fazit

Für das Projekt wurde entschieden den Server zentral zu verwalten um somit die Wartungsarbeiten so niedrig wie möglich zu halten. Durch die Auszeichnung der Einfachheit wird Node.js als Server und MongoDB als Datenbank verwendet.

## 9 Prototypen UI

Nach dem die Anforderungen für das System erstellt wurden, folgen nun die vorläufigen Prototypen.

### 9.x Aufgabenmodellierung

#### 9.x Anforderungsanalyse

##### Ermittlung von Use Cases

Aus den Anforderungen werden die Use Case ermittelt. Diese werden später bei der Gestaltungslösungen mit einbezogen. Von wen stamm dieses Schema?

Use case ID:	#1
Description:	Start der Anwendung
Pre conditions:	Die Anwendung wurde installiert und gestartet
Standard flow:	Der Benutzer möchte die Anwendung nutzen. Hier stehen zwei Optionen zur Verfügung: 1. Login 2. Registrierung
Post conditions:	1. Besteht bereits ein Account so kann der Benutzer erfolgreich sich einloggen und landet im Menü (use case #3) 2. Besteht kein Account, so muss sich der Benutzer sich vorher registrieren (use case #2)

Use case ID:	#2
Description:	Registrierung des Benutzers

<sup>18</sup> Quelle: <https://developers.google.com/maps/mobile-apps?hl=de> (Zugriff am 24.04.15)

Pre conditions:	Der Benutzer trägt seine persönlichen Daten (Name, Geschlecht, Email, Passwort, etc) ein.
Standard flow:	Stimmen die Daten überein und sind dieses vollständig, so können diese dann abgesendet werden.
Post conditions:	Benutzer sendet die eigenen Daten ab und wird wieder zum Login (use case #1) weitergeleitet.

Use case ID:	#3
Description:	Navigation und Menü
Pre conditions:	use case #1 muss erfüllt sein.
Standard flow:	Benutzer kann die verschiedenen Menüpunkte ansteuern. Die Überschrift zeigt an, wo sich der Benutzer zur Zeit befindet.
Post conditions:	Trifft der Benutzer eine bestimmte Auswahl im Menü, so wird er zu den jeweiligen Punkt navigiert (uses case #4-#12)

Use case ID:	#4
Description:	Spielstream, Spielergebnisse
Pre conditions:	use case #1 muss erfüllt sein.
Standard flow:	Der Benutzer kann die zuletzt gespielten Spiele und deren Spielergebnis einsehen.
Post conditions:	Eine Detailansicht der zuletzt gespielten Spiele wird geladen.

Use case ID:	#5
Description:	Herausforderungen
Pre conditions:	use case #1 muss erfüllt sein.
Standard flow:	<ol style="list-style-type: none"> <li>1. Der Benutzer wählt das Datum und die Uhrzeit aus.</li> <li>2. Zu den ausgewählten Zeitpunkt werden Wetterinformationen angezeigt.</li> <li>3. Benutzer wählt ein Basketballplatz aus (use case #7)</li> <li>4. Benutzer wählt seine Gegenspieler aus (use case #6) oder lässt sich durch das System automatisch welche vorschlagen.</li> <li>5. Anschließend fordert er die Spieler heraus.</li> </ol>
Post conditions:	Die Daten wurden korrekt eingegeben und die herausforderten Spieler werden benachrichtigt (#use case 11)

Use case ID:	#6
Description:	Gegenspieler wählen
Pre conditions:	use case #5 muss erfüllt sein.
Standard flow:	Der Benutzer kann einen Gegenspieler auswählen oder einen von System sich vorschlagen lassen.
Post conditions:	Eine Detailansicht über die jeweiligen Spieler im System werden angezeigt.

Use case ID:	#7
Description:	Basketballplatz auswählen
Pre conditions:	use case #5 muss erfüllt sein.
Standard flow:	Der Benutzer kann einen Basketballplatz auswählen oder selbst einen auf der Karte hinzufügen.
Post conditions:	Das ausgewählte Basketballplatz wird angezeigt.

Use case ID:	#8
Description:	Nachrichten versenden
Pre conditions:	use case #1 muss erfüllt sein.
Standard flow:	Der Benutzer kann mit den anderen Spielern über den Chat kommunizieren.
Post conditions:	Der Benutzer sieht die ein und ausgehende Textnachrichten.

Use case ID:	#9
Description:	Wetter
Pre conditions:	use case #1 muss erfüllt sein.
Standard flow:	Der Benutzer kann Wetterinformationen über seinen Standort abrufen, diese kann auch automatisch über GPS lokalisiert werden.
Post conditions:	Eine Detailansicht mit den Wetterinformationen.

Use case ID:	#10
Description:	Einstellungen
Pre conditions:	use case #1 und #3 muss erfüllt sein.

Standard flow:	Der Benutzer kann sich in den Einstellungen <ol style="list-style-type: none"> <li>1. Hilfe: Informationen anzeigen lassen, beispielsweise die Regeln für das Spiel.</li> <li>2. Profil: Informationen über seinen Account.</li> <li>3. Wetter: Hier kann der Benutzer den Standort manuell oder automatisch durch GPS bestimmen und die Vorhersage für 1 Woche wählen.</li> </ol>
Post conditions:	Die Daten werden korrekt angezeigt und gegebenenfalls gespeichert, beispielsweise manueller Standort.

Use case ID:	#11
Description:	Popup Benachrichtigung
Pre conditions:	use case #1, #5, #6 und #7 muss erfüllt sein.
Standard flow:	Ein Benutzer im System hat einen anderen herausgefordert, nun kann dann die Einladung annehmen oder ablehnen.
Post conditions:	Popup wird korrekt dargestellt und bei den Gegenspieler wiedergegeben.

Use case ID:	#12
Description:	Spielergebnis eintragen
Pre conditions:	use case #1, #5, und (#11 wenn angenommen) muss erfüllt sein.
Standard flow:	Ein Benutzer im System hat einen anderen herausgefordert, nun kann dann die Einladung Annehmen oder Ablehnen.
Post conditions:	Popup wird korrekt dargestellt und bei den Gegenspieler wiedergegeben.

## 9.x Papierbasierte Prototypen

Bei einem papierbasierten Prototyp wird mit einfachen Mitteln nur grob eine Skizze gezeichnet und dient bei der Entwicklung neuer Produktkonzepte. Einer der wichtigen Vorteile ist, dass es keine Grenzen bei der Kreativität durch Technik oder sonstiges gesetzt werden. Hier wird der Focus auf die vereinfachte Darstellung und leichte Zugänglichkeit für den Benutzer gelegt. Die einzelnen Anforderungen werden hier in Abhängigkeit gezogen und in Iteration umgesetzt. Weiterhin sollten die Prototypen als Screendesign umgesetzt werden.

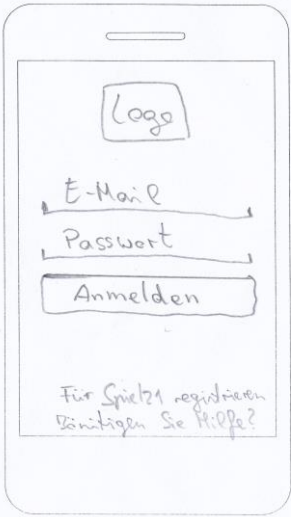
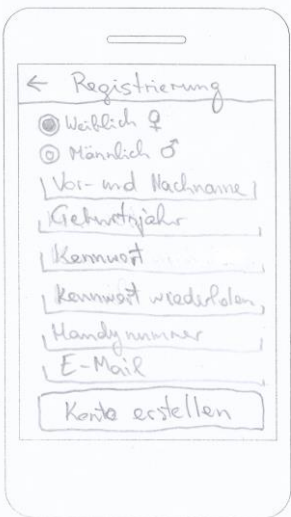

 <p><i>Abbildung 1: uc#1 Start der Anwendung</i></p>	<p>Oben sollte das Logo des Spiels erscheinen, um speziell die Domäne zu erkennen. Zwei Optionen sollten auf dem Startbildschirm zu Verfügung gestellt werden. Die Anmeldung (use case #1) mit E-Mail und Passwort, dafür gibt es einen Button. Nach erfolgreichen Einloggen sollte der Benutzer zum Menü weiter geleitet werden. Ein anderer Punkt ist die Registrierung (use case #2), dafür gibt es einen Text zum anklicken.</p>
 <p><i>Abbildung 2: uc#2 Registrierung des Benutzers</i></p>	<p>Bei der Registrierung werden die wichtigen Daten von den Benutzer abgefragt, wie beispielsweise Name, Geschlecht, Passwort, etc. und anschließend nach dem Ausfüllen aller Felder werden die Daten persistent in der Datenbank gespeichert. Um das Konto zu erstellen gibt es einen entsprechenden Button dafür. Nach erfolgreicher Registrierung sollte der Benutzer eingeloggt werden (use case #1) und zum Menü (use case #3) weiter geleitet werden.</p> <ul style="list-style-type: none"> <li>- Selbsterklärend</li> <li>- Daten eintragen</li> <li>- Button klicken</li> </ul>
 <p><i>Abbildung 3: uc#3 Nav. und Menü</i></p>	<p>Nach dem erfolgreichen Einloggen sollte der Benutzer das Navigationsmenü zu sehen bekommen und darüber die wichtigen Menüpunkte (use case #4, #5, #7, #8, #9 und #12) ansteuern können.</p>



Abbildung 4: uc#4 Spielstream

Im dem Menüpunkt Spielstreams hat der Benutzer die Ansicht mit den gespielten Spielen. Hier sind hierarchisch die gespielten Duelle zu sehen zwischen den einzelnen Benutzer mit dem entsprechenden Spielergebnis und den Basketballplatz. Des Weiteren sollte oben eine ActionBar zu sehen sein um somit die Navigation zu erleichtern und um weitere Einstellungen (use case #10) vornehmen zu können.



Abbildung 5: uc#5 Herausforderungen

Die Überschrift zeigt immer wo sich zur Zeit der Benutzer befindet. Hier kann der Benutzer andere Spieler zu einem Duell herausfordern, dabei kann er das Datum und die Uhrzeit bestimmen, daraufhin wird automatisch das passende Wetter für den spielenden Tag angezeigt. Weiterhin wählt er einen Basketballplatz (use case #7) und weitere Spieler (use case #6) aus. Das System kann die Spieler auch automatisch vorschlagen durch das Matchmaking.



Abbildung 6: uc#6 Gegenspieler wählen

Im Menü Herausfordern kann der Benutzer manuell seinen Gegenspieler wählen. Durch das Plus-Symbol wird im eine Itemliste vorgeschlagen mit den registrierten Spielern im System, dort kann er seinen Gegenspieler auswählen und kehrt anschließend wieder in das Herausforderungen Menü (use case #5) zurück.



Abbildung 7: uc#7 Basketballplatz auswählen

Im Menü Herausforderungen (use case #5) kann man auf einer Karte einen Basketballplatz auswählen, seinen eigenen Standort bestimmen, sich zu den Basketballplatz navigieren lassen und einen neuen hinzufügen. So kann hier beispielsweise ein Basketballplatz gewählt werden und man kehrt anschließend wieder in das Herausforderungen Menü (user case #5) zurück. Die Kartenansicht sollte auch über das Menü (use case #3) direkt aussteuerbar sein.



Abbildung 8:uc#8 Nachrichten versenden

Im Menü Nachrichten können die Benutzer verschiedenen Textinformationen austauschen und andere Benutzer kennen lernen. Hier kann beispielsweise ausgemacht werden, wer den Basketball zum Platz bringen soll.

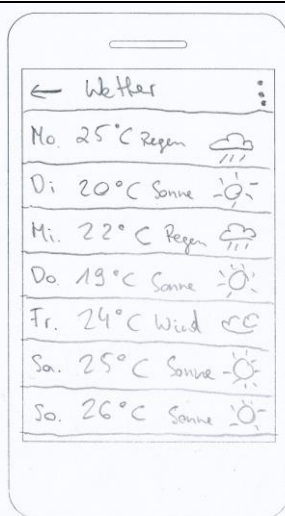


Abbildung 9: uc#9 Wetter

Einer wohl der wichtigsten Nutzungsprobleme bei dem Spiel ist wohl die Wetterlage. Im Menü Wetter wird eine Vorhersage des Wetters bekannt gegeben und der Benutzer kann sich über die Wetterlage informieren und kann somit besser mit anderen Benutzern verabreden.



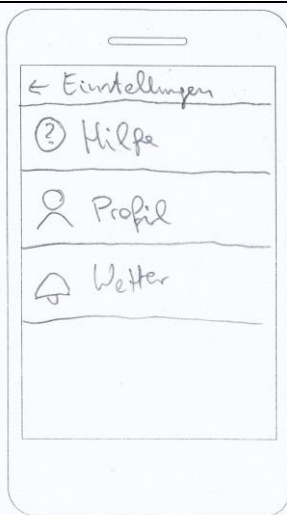


Abbildung 10: uc#10 Einstellungen

Die Option Einstellungen wird nach den erfolgreichen einloggen sichtbar und beinhaltet Auskunftsinformationen über das Spiel unter dem Punkt „Hilfe“, dort kann der Benutzer sich beispielsweise über die Regeln des Spiels informieren. Unten den Punkt „Profil“, kann der Benutzer seine registrierten Informationen im System einsehen. Letzter Punkt ist das „Wetter“, dort kann der Benutzer den Standort manuell bestimmen oder auch durch das GPS automatisch bestimmen lassen.



Abbildung 11: uc#11 Popup Benachrichtigung

Die Popup Benachrichtigung taucht dann auf, wenn ein Benutzer einen anderen zum Spiel herausfordert (use case #5). Hier werden die Grundlegende Informationen (Name, Datum, Uhrzeit, Wetter) für den Benutzer gezeigt. Hier kann der Benutzer selbst die Entscheidung treffen, ob er die Herausforderung annimmt oder ablehnt.



Abbildung 12: #uc12 Spielergebnis eintragen

Sollte es zu einer Herausforderung (use case #5) kommen und diese wurde auch angenommen (use case #11). So kann der Benutzer nach dem gespielten Spiel sein Spielergebnis eintragen, der Gegenspieler muss das gleiche tun. Das System verifiziert dann die Ergebnisse und zeigt diese später in den Spielstream (use case #3) an.



**9.x Erkenntnisse aus dem Evaluationsprozess**

Die Beurteilung der verschiedenen Prototypen hat ergeben, dass das Design überarbeitet werden sollten weil.

9.x Screendesign

Feedback hier. Später Screenshots etc..

## **10. Installationsanleitung**

### **Anforderungen**

#### **Serverseitig**

- Server, Datenbank

#### **Clientseitig**

- Android Anwendung

## Anhang

### Fragebogen Spiel 21

**WICHTIG! Der Fragebogen wird anonym ausgewertet.**

Wie alt sind Sie?

\_\_\_\_\_ Jahre                      ( ) weiblich                      ( ) männlich

Was machen Sie beruflich?

\_\_\_\_\_

Wo wohnen Sie?

\_\_\_\_\_

Wie lange wohnen Sie da?

( ) weniger als 1 Jahr                      ( ) 1 Jahr                      ( ) 5 Jahre                      ( ) länger

Haben Sie ein eigenes Auto?

( ) ja                      ( ) nein

Besitzen Sie ein internetfähiges Endgerät (bspw. Smartphone)?

( ) ja                      ( ) nein

Wenn ja, habe Sie einen Volumentarif oder Flatrate?

( ) Volumen                      ( ) Flat

Schätzen Sie, wie viel Freizeit habe Sie in der Woche?

\_\_\_\_\_ Stunden

Haben Sie schon mal Ihr Endgerät zum suchen von Feizeitmöglichkeiten verwendet?

( ) ja                      ( ) nein

Haben Sie am Wochenende Langeweile?

☐ nie      ☐ selten      ☐ oft      ☐ immer

Nutzen Sie Angebote von Sportklubs oder ähnlichen Einrichtungen?

☐ ja      ☐ nein

Wie wichtig ist für Sie Sport?

☐ nicht wichtig      ☐ wenig wichtig      ☐ wichtig      ☐ sehr wichtig

Machen Sie Sport in ihrer Freizeit?

☐ ja      ☐ nein

Wenn ja, welche Sportart

☐ Fitnessstudio      ☐ Fußball      ☐ Basketball  
☐ Tischtennis      ☐ Schwimmen      ☐ Volleyball  
☐ Andere \_\_\_\_\_

Kennen Sie das Basketballspiel 21 (Twenty-One)?

☐ ja      ☐ nein

**Vielen Dank, dass du dir die Zeit genommen hast, den Fragebogen auszufüllen.**

**Text für die Online Umfrage:**

***Umfrage im Rahmen der Veranstaltung „Entwicklungsprojekt interaktive Systeme“.***

*Im Rahmen der Veranstaltung „Entwicklungsprojekt interaktive Systeme“ möchte ich gerne mithilfe Eurer Meinung wichtige Informationen zu meinem Projekt dazu ermitteln.*

*Ich würde mich sehr freuen, wenn ihr Euch etwa 10 Minuten für die Umfrage zeitnehmen könntet. Die Befragung erfolgt anonym und es sind keinerlei Rückschlüsse auf einzelne Personen möglich.*

*Zur Umfrage gelangt ihr über diesen Link: <http://goo.gl/forms/vziaD5ebwb>*

*Über Eure Unterstützung würde ich mich sehr freuen!*

*Vielen Dank im Voraus.*

*Viele Grüße, Eduard König*

## Auswertung des Fragebogen

Fragen	Zeitstempel: 14.05-17.05.2015	Befragtenanzahl: 32 Personen
Wie alt sind Sie?	Weiblich: 27,85 Männlich: 27,85 Durchschnitt: 26,56	
Welches Geschlecht?	Weiblich: 8 Männlich: 24	
Was machen Sie beruflich?	Es wurden meistens Studenten gefragt	
Wo wohnen Sie?	Die meisten befragten kommen aus Gummersbach	
Wie lange wohnen Sie da?	weniger als 1 Jahr: 3 1 Jahr: 10 5 Jahr: 6 Länger: 13	
Haben Sie ein eigenes Auto?	Ja: 19 Nein: 13	
Besitzen Sie ein internetfähiges Endgerät (bspw. Smartphone)?	Ja: 32 Nein: 0	
Wenn ja, habe Sie einen Volumentarif oder Flatrate?	Flatrate: 20 Volumen: 12	
Schätzen Sie, wie viel Freizeit habe Sie in der Woche (in Stunden)?	Durchschnitt: 26,15	
Haben Sie schon mal Ihr Endgerät zum suchen von Feizeitmöglichkeiten verwendet?	Ja: 23 Nein: 9	
Haben Sie am Wochenende Langeweile?	Nie: 5 Selten: 25 Oft: 2 Immer: 0	
Wenn ja, welche Sportart?	Basketball: 2, Rest: 30 Am meisten: Fitnessstudio	
Machen Sie Sport in ihrer Freizeit?	Ja: 27 Nein: 5	
Kennen Sie das Basketballspiel 21 (Twenty-One)?	Ja: 15 Nein: 17	

*Tabelle: Auswertung des Fragebogens*

## Personas

	<p><b>Markus</b></p> <p>Markus ist 17 Jahre alt, ein Jugendlicher und arbeitet bei der Fa. Saturn in Gummersbach.</p> <p>In seiner Freizeit spielt er gerne Fußball mit seinen Freunden. Dennoch würde auch gerne Basketball spielen, sowie früher das Spiel 21, kennt er noch aus der Realschule.</p> <p>Er versucht immer wieder seine Freunde zu überreden aber hat leider keinen Erfolg. Dabei würde er so gerne ein paar Körbe schmeißen.</p> <p>Er hat Abend immer viel Zeit und geht gerne raus beim schönen Wetter, wenn es nicht regnet. Dann kann man am besten das Wetter genießen bei einem Spielchen unter Freunden.</p> <p>Am liebsten verabredet er sich mit seinen Freunden über das Smartphone.</p>
<p>Alter: 17          Beschäftigung: Azubi          Status: single          Sport: Basketball, Fußball          Mobilität: Fahrrad</p>	

*Tabelle: Persona Markus*

	<p><b>Anna</b></p> <p>Anna ist eine Mutter mit einem Kind im Alter von 6 Jahren.</p> <p>Sie hat einen neuen Job als Sachbearbeiterin in Gummersbach bei der Stadt. In der Freizeit beschäftigt sie sich gerne mit Sport.</p> <p>Sie versucht sich selbst immer zum Sport zu animieren, doch es fällt leider schwer die Sportlichen Aktivitäten alleine auszuüben. Den einer muss immer auf das Kind auf aufpassen, deshalb kann auch der Ehermann nicht mitkommen.</p> <p>Am liebsten spielt sie Basketball. Doch leider kann sie keine Freunde finden in der Umgebung. Nach dem Umzug kennt sie sich in Gummerbach noch nicht so gut aus. Die Zeit ist knapp und sie würde gerne neue Leute kennenlernen die genau soviel Spaß haben an sportlichen Aktivitäten wie sie.</p> <p>Sie benutzt ihr Smartphone um mit Ihren Freunden aus der alten Umgebung in Kontakt zu bleiben.</p>
<p>Alter: 34          Beschäftigung:          Sachbearbeiterin          Status: verheiratet          Sport: Jogging, Basketball          Mobilität: Auto, Fahrrad</p>	

*Tabelle: Persona Anna*

## **Szenarien**

### **Szenario 1: Langweile**

Es ist 15:00 Uhr und Henry D. macht heute früher Feierabend. Er würde gerne noch irgend eine sportliche Aktivität machen, bevor er komplett für heute Abend abschaltet. Er langweilt sich nur immer, das seine Kollegen immer wieder nur Fußball spielen wollen. Dabei gibt es doch auch so viele andere Sportarten. Er würde gerne was neues ausprobieren, wie beispielsweise Basketball.

### **Szenario 2: Wechselhafte Wetterbedingungen**

Daniel und Markus sind beste Freunde und spielen gerne das Basketballspiel 21 zusammen. Durch das wechselhafte Wetter in letzter Zeit haben die beiden Jungs mittlerweile kein Lust zum spielen und bleiben lieber gerne zu Hause und spielen Computerspiele. Des Weiteren kommen immer wieder Streitereien zwischen den beiden, da sie sich den Punktestand nicht merken, wer zuletzt gewonnen hat. Dabei wünschen sich die beiden nicht sehnlicher als eine Anwendung, wo sie ihren Punktestand aufschreiben können und durch den bevorstehenden Regen gewarnt werden.

### **Szenario 3: Begeisterter Basketballspieler**

Peter M. ist ein begeisterter Basketballspieler. Schon seit längeren möchte er gerne mehr Aufmerksamkeit auf die Sportart lenken und versucht die jungen Leute zu begeistern. Eines seiner Konzepte ist das Organisieren von Wettkämpfen und das sammeln von Statistiken aus den Spielbegegnungen. Dabei würden einige der Spieler und Zuschauer auch begrüßen, wenn die Ergebnisse des Spiels öffentlich zugänglich wären.

### **Szenario 4: Neu in der Umgebung**

Maria S. studiert seit zwei Semestern an der FH-Köln und ist relativ neu in der Umgebung. Sie wohnt in einer WG und hat ab Freitag bis Sonntag Freizeit und würde gerne sportlich aktiv sein, ohne sich in ein Sportverein einschreiben zu müssen. Seit dem Umzug hat sie leider immer noch keine Kenntnis von der Umgebung und kennt sehr wenige Leute aus der Umgebung. Dabei würde sie so gerne neue Leute kennen lernen, ohne irgendwelche Fitness-Verträge abschließen zu müssen.

### **Szenario 5: Sportunterricht**

Die Lisa S. ist die Klassenbeste im Sportunterricht. Sie liebt es Basketball zu spielen, und kennt noch das Spiel 21 aus dem Sportunterricht. Dies findet allerdings leider äußerst selten statt, da das jetzige Sportprogramm dies nicht mehr vorsieht. Ebenso kann sie ihre Klassenkameraden nur schwer für das spezielle Variante des Basketballspiel begeistern. Nur

unter größerem Aufwand und viel Überredungskunst ist es jedoch spontan möglich, einige Partien unter den Schülern zu starten.



## Quellenverzeichnis

### Literatur

- [ISO 9241] Deutsches Institut für Normung (2010) Ergonomie der Mensch-System-Interaktion: Teil 210: Prozess zur Gestaltung gebrauchstauglicher interaktiver Systeme (ISO 9241-210:2010). DIN Deutsches Institut für Normung, 2010.
- [IEEE 610.12-1990] IEEE Standard Glossary of Software Engineering Terminology (1990) 610.12-1990. IEEE Computer Society.
- [Cooper] Alan Cooper, (2009). About Face: Interface und Interaction Design, Bonn: Mitp-Verlag.
- [Shepherd] Andrew Shepherd, Hierarchical Task Analysis, Taylor & Francis, London and New York, 2001
- [Tilkov] Stefan Tilkov, REST und HTTP: Entwicklung und Integration nach dem Architekturstil des Web, dpunkt.verlag, 2015

### Internetquellen

- [Spiel 21 Regeln] <http://www.sportunterricht.ch/lektion/spielen/spielen80.php>  
(Zugriff am: 17.06.2015)
- [Wetter.com APP] <https://play.google.com/store/apps/details?id=com.wetter.androidclient>  
(Zugriff am: 20.04.15)
- [Wetter Online APP] <https://play.google.com/store/apps/details?id=de.wetteronline.wetterapp> (Zugriff am 20.04.15)
- [Whatsapp APP] <https://play.google.com/store/apps/details?id=com.whatsapp>  
(Zugriff am 20.04.15)
- [Facebook APP] <https://play.google.com/store/apps/details?id=com.facebook.orca>  
(Zugriff am 20.04.15)
- [Navigon APP] [https://play.google.com/store/apps/details?id=com.navigon.navigator\\_select](https://play.google.com/store/apps/details?id=com.navigon.navigator_select)  
(Zugriff am: 20.04.15)
- [TomTom APP] <https://play.google.com/store/apps/details?id=com.tomtom.gplay.navapp>  
(Zugriff am 20.04.15)
- [GoogleMaps APP] <https://play.google.com/store/apps/details?id=com.google.android.apps.maps> (Zugriff am 20.04.15)
- [MAPS.ME APP] <https://play.google.com/store/apps/details?id=com.mapswithme.maps.pro>  
(Zugriff am 20.04.15)
- [Node.js] <https://nodejs.org/> (Zugriff am 21.04.15)
- [MongoDB] <https://www.mongodb.org/> (Zugriff am 21.04.15)
- [Wetter.com API] [http://www.wetter.com/apps\\_und\\_mehr/website/api/](http://www.wetter.com/apps_und_mehr/website/api/)  
(Zugriff am 21.04.15)

**[GoogleMaps API]** <https://developers.google.com/maps/mobile-apps?hl=de>  
(Zugriff am 24.04.15)