

UNIVERSITÄT POTSDAM
INSTITUT FÜR INFORMATIK

PROJEKT

ROUTEME

LEVEL 3

Kora Haedge, Stefanie Lemcke

Zusammenfassung

In der vorliegenden Arbeit wird die Weiterentwicklung des pervasiven Lernspiels *RouteMe* beschrieben. Pervasive Lernspiele bilden eine gute Alternative für die traditionelle Wissensvermittlung und werden immer häufiger eingesetzt, um Lerninhalte zu vermitteln. RouteMe soll dabei die Konzepte im Bereich mobiler Routing-Protokolle, speziell für den Algorithmus AODV, spielerisch darstellen. Das Spiel umfasste bereits zwei Level mit erhöhter Unterstützung für den Spieler. Aufbauend auf diesem Stand wurde ein weiteres Level mit verringerter Assistenz konzipiert und umgesetzt. Das entwickelte Konzept sowie die dazu notwendigen Erweiterungen des bestehenden Systems werden diskutiert. Weiterhin werden kurz die durchgeführten Nutzertest geschildert und die daraus hervorgegangene Ergebnisse und Problem sowie mögliche Erweiterungen umrissen.

Inhaltsverzeichnis

| | |
|--|-----------|
| Abbildungsverzeichnis | 4 |
| 1. Einleitung | 5 |
| 2. Ausgangssituation | 5 |
| 2.1. Bisherige Umsetzung | 6 |
| 2.1.1. Backend | 7 |
| 2.1.2. Kommunikation | 7 |
| 2.1.3. Web-Ansichten | 7 |
| 2.1.4. Mobile Client | 7 |
| 3. Konzeption | 8 |
| 3.1. erforderliche Erweiterungen | 8 |
| 4. Umsetzung | 10 |
| 4.1. Backend | 10 |
| 4.1.1. Algorithmus | 11 |
| 4.1.2. Punktevergabe | 13 |
| 4.2. Kommunikation | 14 |
| 4.3. Web-Ansichten | 15 |
| 4.4. Mobiler Client | 16 |
| 4.4.1. Levelunterscheidung | 16 |
| 4.4.2. Pakete und Routing | 16 |
| 4.4.3. Routing | 17 |
| 5. Nutzeranleitung | 18 |
| 5.1. Installation | 18 |
| 5.1.1. Entwicklungsumgebung | 18 |
| 5.1.2. Server | 19 |
| 5.1.3. Mobile-Client | 20 |
| 5.2. Kurzanleitung | 20 |
| 5.2.1. Spielstart und -verwaltung | 20 |
| 5.2.2. Client | 21 |
| 5.2.3. Mobile | 21 |
| 5.3. Spielverlauf | 21 |
| 5.3.1. Indoor | 21 |
| 5.3.2. Mobile | 22 |
| 5.4. Spielende | 22 |
| 6. Fazit/Ausblick | 22 |
| 6.1. Fazit | 22 |
| 6.2. Nutzertest | 23 |
| 6.3. Ausblick | 23 |
| 6.3.1. GPS | 23 |
| 6.3.2. Pervasivität | 24 |
| 6.3.3. Karte | 24 |
| 6.3.4. Spielfeld | 24 |
| 6.3.5. Administration | 25 |
| 6.3.6. Paketwahl (Indoor) | 25 |
| 6.3.7. Verwerfen von Paketen (Outdoor) | 25 |

| | |
|------------------------|-----------|
| Literatur | 26 |
| A. Fragebogen | 27 |
| B. Daten | 30 |
| C. Auswertung | 32 |
| C.1. Indoor | 32 |
| C.2. Outdoor | 35 |
| C.3. Gesamt | 38 |

Abbildungsverzeichnis

| | |
|--|----|
| 1. Architektur [DG13] | 6 |
| 2. Mobilclient mit Paketauswahl und Levelanzeige | 11 |
| 3. Indoor-Ansicht mit Paketauswahl und Levelanzeige Variante 1 | 12 |
| 4. Indoor-Ansicht mit Paketauswahl und Levelanzeige Variante 2 | 12 |
| 5. Änderung Spielerlevel | 15 |
| 6. Paket und Routenwahl Indoor-Spieler | 15 |
| 7. Header der Applikation | 16 |
| 8. Sequenzdiagramm zum Senden eines Pakets | 17 |
| 9. Anzeige der Pakete eines Spielers | 17 |
| 10. Gesamtansicht der Applikation | 18 |
| 11. Erstellen eines Spiels | 20 |
| 12. Erstelltes, noch pausiertes Spiel | 21 |
| 13. Daten aller Teilnehmer | 30 |
| 14. Daten der Indoor-Teilnehmer | 30 |
| 15. Daten der Outdoor-Teilnehmer | 30 |
| 16. Indoor: Auswertung | 32 |
| 17. Indoor: Konsistenz | 33 |
| 18. Outdoor: Auswertung | 35 |
| 19. Outdoor: Konsistenz | 36 |
| 20. Gruppe: Auswertung | 38 |
| 21. Gruppe: Konsistenz | 39 |

1. Einleitung

Das Prinzip des Lernens durch Spielen wird bereits lange Zeit von Bildungseinrichtungen genutzt. Seit es Personal Computer in fast jeden Haushalt geschafft haben, gibt es auch Lernsoftware für Kinder und Erwachsene. Durch den Vormarsch mobiler Geräte wird das Prinzip dieses Lernens noch um die gegebene Mobilität ergänzt. Während das Lernen mit Software vorher nur vor dem Bildschirm sitzend möglich war, gibt es jetzt die Möglichkeit, sich während der Nutzung eines Spiels auf einem mobilen Gerät (z.B. einem Smartphone oder Tablet) drinnen oder draußen frei zu bewegen oder sogar die Umgebung in den Lernvorgang einzubeziehen. Diese Allgegenwart von Computern und -programmen wird als Pervasive Computing bezeichnet.

Ein Beispiel für ein pervasives Lernspiel stellt die in dieser Arbeit erweiterte Software „RouteMe“ dar, die in mehreren aufbauenden Projekten und Abschlussarbeiten von Studierenden des Instituts für Informatik und Computational Science¹ der Universität Potsdam entwickelt wurde. Das Spiel dient dem Verständnisszuwachs bezüglich des Routing-Protokolls AODV, indem der Spieler verschiedene „Rollen“ übernimmt, welche die Akteure eines Routing-Vorgangs symbolisieren.

AODV ist ein Protokoll, das für mobile ad-hoc Netzwerke (MANETs) entwickelt wurde und dynamisch bei Bedarf nach Routen über verschiedene mobile Knoten sucht. Gehen bestehende Routen verloren, wird ein Paket, das gerade unterwegs ist, nicht verworfen sondern es werden neue Routen generiert [Net03]. In RouteMe können Spieler entweder die Rolle des Senders übernehmen, der Pakete in einem Netz von einem Knoten zu einem anderen schicken möchte oder die des mobilen Knoten, der sich frei bewegt und über Nachbarschaftsbeziehungen Pakete weitersendet. Im Vorfeld wurden bereits ein einführendes und ein vom Schwierigkeitsgrad leicht erhöhtes Level entwickelt. In der vorliegenden Arbeit wurde das Spiel um ein weiteres Level erweitert, das die Kenntnisse der vorigen Spielstufen voraussetzt und eine hohe Initiative und Denkarbeit erwartet, wenn der Spieler Punkte verdienen möchte.

2. Ausgangssituation

Die eingangs erwähnten Gruppen von Spielern werden bei RouteMe als „mobile Knoten“ und als „Indoor-Spieler“ bezeichnet. Ein mobiler Knoten ist ein Spieler, der sich mit einem mobilen Endgerät – z. B. einem Smartphone – auf einem definierten Spielfeld bewegt. Die Indoor-Spieler agieren als Sender von Nachrichtenpaketen und spielen von einem Laptop oder Desktop-PC, während die mobilen Knotenspieler sich außerhalb eines Gebäudes mit Smartphones bewegen. Die Indoor-Spieler müssen Pakete von einem Start- zu einem Zielknoten routen. Hierzu werden die mobilen Spieler als Start-, Ziel- und Zwischenknoten verwendet.

Das Lernspiel wurde zunächst von Tobias Moebert in seiner Diplomarbeit konzipiert und umgesetzt. Die Implementation für die Spiellogik und die Webfrontends erfolgte dabei in PHP und Javascript. Als Datenbank wurde MySQL gewählt [DG13].

Im Folgenden wurde diese Arbeit von Julian Dehne und Hendrik Gefner refaktorisiert. Dabei wurde der Code nach Java portiert. Für die Entwicklung der verschiedenen Webfrontends wählte man das Google Web Toolkit (GWT). Aus Performancegründen wurde auf eine eingebettete H2-Datenbank gewechselt, wobei Hibernate als Framework zur Persistierung dient. Zusätzlich entwickelte man eine Rest-Schnittstelle, welches aus der Entscheidung resultierte, Teile des JavaScript-Codes von Tobias Moebert zu übernehmen [DG13].

Auf dieser Grundlage fertigte Hendrik Gefner seine Bachelorarbeit an. Es wurde die Spiellogik um ein weiteres Level erweitert, sowie der Mobile-Client für die Outdoor-Spieler für Android-Systeme umgesetzt [Ges13]. Die entstandene Rest-Schnittstelle wurde erweitert, um Funktionen für die Kommunikation mit diesem neuen Client bereitzustellen.

¹<http://www.cs.uni-potsdam.de>

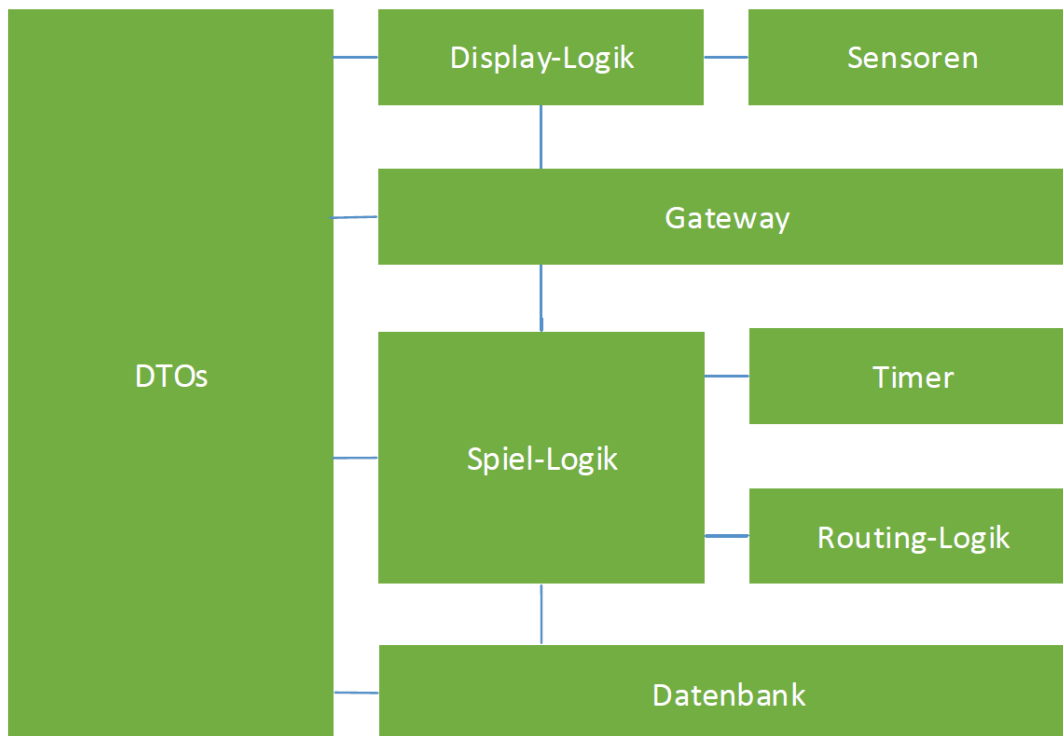


Abbildung 1: Architektur [DG13]

Im Ausgangszustand sind folgende Level umgesetzt:

In **Level 1** werden die Spieler sehr stark unterstützt. Es können alle Knoten von den Indoor-Spielern als Start- und Endpunkt ausgewählt werden und das Routing erfolgt automatisch. Die mobilen Spieler müssen lediglich Batterien und andere Gegenstände einsammeln, sich aber nicht um das konkrete Routing kümmern.

Level 2 erfordert mehr Initiative und Nachdenken der Spieler. Mobile Spieler müssen durch Senden einer HELLO-Nachricht ihren Nachbarn ihre Existenz signalisieren, woraufhin diese dann Pakete an sie senden können. Der Paketversand selbst erfolgt weiterhin automatisch. Es werden zufällig mögliche Routen generiert und dem Nachrichtensender angezeigt. Dieser muss selbstständig günstige Routen erkennen und auswählen.

2.1. Bisherige Umsetzung

Für die Weiterentwicklung wurde die Bachelorarbeit von Hendrik Geßner gewählt, welche auf Github (<https://github.com/ItsNotYou/RouteMeBA>) verfügbar ist. Ausgangspunkt bildet nach Rücksprache mit Hendrik Geßner der Master-Branch (Stand: 23. Oktober 2013).

Abbildung 1 zeigt die logische Architektur der Implementierung, deren Komponenten in den kommenden Kapiteln vorgestellt werden.

Insgesamt besteht das System aus 5 Teilprojekten:

- LocationLibrary
- NexplorerGWT: umfasst die Indooransichten sowie die Implementierung des Routingalgorithmus
- RouteMeMobile: beinhaltet den Mobile-Client

- RouteMeMobileTest: stellt Tests für die Funktionalitäten des Mobile-Clients zur Verfügung
- RouteMeRestLibrary: definiert Objekte für den Datenaustausch zwischen Backend und Mobile-Client

2.1.1. Backend

Das Backend umfasst die Komponenten DTOs, Spiellogik, Routinglogik, Timer und Datenbank und ist für den Spielkontrolle und die Durchführung des Routingalgorithmus verantwortlich. Mittels Timer wird dabei das automatische Senden von Paketen, RoutingRequest- und RoutingError-Nachrichten gesteuert.

Das System wurde in Java umgesetzt und ist zustandslos. Die zu speichernden Daten werden in einer eingebetteten H2-Datenbank gehalten, wobei für die Persistierung und Kommunikation mit dieser das Hibernate-Framework eingesetzt wurde. Hibernate ist ein Open Source Object Relational Mapping (ORM) Framework, welches für die Persistierung von Daten in relationalen Datenbanken via JDBC genutzt wird. Es läuft unter der LGPL v2.1 Lizenz [red].

2.1.2. Kommunikation

Für die Kommunikation des Mobile Clients mit dem Server steht eine Rest-Schnittstelle zur Verfügung. RESTful Webservices bieten die Möglichkeit Daten über http-Aufrufe zu übermitteln, zu lesen oder zu löschen [Elk08].

Als Austauschformat dienen JSON-Objekte². Mittels des Jackson-Frameworks³ kann anhand von Annotationen entschieden werden, welche Objekte innerhalb einer Klasse in JSON-Objekte konvertiert werden sollen, bzw. welche nicht.

Für die Kommunikation aus den in GWT⁴ implementierten Webansichten mit dem Backend wurde die bereits bestehende JavaScript-Schnittstelle verwendet.

2.1.3. Web-Ansichten

Als Indooransichten werden die Indoorview für die Spieler, welche die Pakete senden, und die Adminview für die Spielkontrolle angesehen. Für die Displaylogik wurde GWT eingesetzt. GWT ist ein Open Source Framework, welche für die Entwicklung cross-Browser Webanwendungen verwendet werden kann. Die Implementierung erfolgt dabei in Java und wird anschließend nach JavaScript kompiliert [GWT]. Für die Kommunikation mit dem beibehaltenen JavaScript-Code wird das JavaScript Native Interface (JSNI)⁵ verwendet.

2.1.4. Mobile Client

Der mobile Android-Client baut sein User Interface (UI) größtenteils aus XML-Dateien auf. Das UI wird in einem festgelegten Zeitintervall aktualisiert, welcher derzeit auf 5000ms festgelegt ist. Elementar in der Entwicklung des nativen Clients war die Einbeziehung der Hardware (des Smartphones). Dies wird besonders deutlich in der Möglichkeit, Gegenstände durch ein Schütteln des Smartphones einzusammeln, was außerdem eine kurze Vibration als Feedback hervorruft. Auch ist hier die auf Google Maps basierende Kartendarstellung, welche sich je nach Neigung des Smartphones dem Blickwinkel anpasst, zu erwähnen.

Unter Android muss für den Mobilclient eine Umwandlung des zur Kommunikation mit dem Backend verwendeten JSON-Formats in Java-Objekte erfolgen, was über Google Gson⁶ realisiert

²<http://json.org/>

³<http://jackson.codehaus.org/>

⁴<http://www.gwtproject.org/>

⁵<http://www.gwtproject.org/doc/latest/DevGuideCodingBasicsJSNI.html>

⁶<https://code.google.com/p/google-gson/>

ist. Außerdem wird die Bibliothek „Spring für Android“⁷ verwendet, welche das Implementieren von Callbacks ermöglicht.

3. Konzeption

Ziel der Weiterentwicklung ist es ein neues Level bereitzustellen. Dabei wird die Unterstützung für die Spieler weiter verringert, um einen erhöhten Schwierigkeitsgrad und somit ein tieferes Verständnis für den Algorithmus zu schaffen.

Für die Umsetzung ergeben sich neue Anforderungen für die verschiedenen Personengruppen Admin, Indoor-Spieler und Outdoor-Spieler.

Zu Beginn muss zunächst eine Auswahlmöglichkeit für Level 3 geschaffen werden. Zudem muss es möglich sein, jedem Spieler vor Spielstart ein Level zuzuweisen. Diese Level können unterschiedlich sein. Während eines Spiels bleiben diese Level konstant.

Da mit verschiedenen Levels gespielt werden kann, müssen dem Admin sowie den In- und Outdoor-Spielern ihre Level angezeigt werden, damit der Spielablauf transparent bleibt. Level anderer Spieler sind einem Spieler nicht bekannt.

Die Routenauswahl für die Indoor-Spieler wird analog zu Level 2 (d.h. Auswahl aus einer Liste mit zufällig ausgewählten Routen) umgesetzt. Alternativ wäre auch denkbar, wie in Level 1 den Start- und Endpunkt auswählen zu lassen, jedoch ist bei dieser Variante eine höhere Assistenz gegeben, welche in Level 3 nicht gewünscht ist.

Einer der wichtigsten Aspekte an Level 3 ist, dass der Indoor-Spieler verschiedene Paketarten senden kann. Hierzu wird dem Spieler zunächst ein konkretes Paket vorgegeben, welches er zu versenden hat. Darauf aufbauend wird in einem späteren Entwicklungsstand ein bestimmtes Kontingent an Paketen vorgegeben, welches eine feststehende Anzahl jeder Paketart beinhaltet. Erst wenn diese Pakete alle erfolgreich verschickt wurden, bekommt der Spieler neue Pakete zugewiesen. Dadurch erhält der Spieler die Möglichkeit, gute Pakete auf potentiell guten Routen zu senden und so mehr Punkte zu bekommen. Die Anzahl der gewonnen Punkte nach erfolgreichem Routing eines Pakets hängt von der Art des Pakets ab.

Für den Outdoor-Spieler ist in Level 3 die größte Änderung zu den vorigen Levels, dass er mehr oder weniger dringende Pakete zum Routing gegeben bekommt. Er muss sowohl die Art des Pakets als auch die bereits verstrichene Zeit, die das Paket auf ein Routing wartet, mit in seine Entscheidungen einbeziehen können. Zudem ist nach dem Routing ein Feedback in Form einer Bewertung des Vorgangs (gut oder schlecht) sinnvoll, sodass der Spieler aus Entscheidungen lernen kann.

3.1. erforderliche Erweiterungen

Im Folgenden werden die konkretisierten Anforderungen an die verschiedenen Client-Sichten tabellarisch (Admin-, Indoor- und Outdoor-Ansicht) aufgeführt sowie die geplante Umsetzung skizziert. Dabei bildet das Design der vorhandenen GUIs die Grundlage und wird lediglich um neue Funktionen erweitert.

| <i>Sicht</i> | <i>Anforderung</i> | <i>geplante Umsetzung</i> |
|---------------------|---------------------------------------|--|
| Admin-Client | <i>Auswahlmöglichkeit für Level 3</i> | Erweiterung der Startansicht Level für einzelne User bei Spielstart festlegen |

⁷<http://projects.spring.io/spring-android/>

| | | |
|-----------------------|--|--|
| | | (konstante Level während eines Spiels) |
| | <i>Anzeige der Userlevel</i> | Spielerstatus in Adminansicht um Level erweitern |
| Indoor-Client | <i>Levelanzeige</i> Routenauswahl analog zu Level 2 <i>zu sendende Pakete</i> zunächst: Vorgabe, welches Paket als nächstes gesendet werden muss darauf aufbauend: vorgegebenes Paketkontingent <i>Punktevergabe</i> abhängig von Paketart Anzeige der Punkte nicht geroutete Pakete | Spielerstatus erweitern durch Icons, wie in Outdoor-Client beschrieben unverändert in GUI (Lvl 2) und Logik Einfache Info in Gui (Abb. 3) Paketkontingent wird erst nach erfolgreichem Versand aller Pakete erneuert Anzeige und Auswahl der Pakete in der GUI analog zu den Routen(Abb. 4). Es werden verschiedene Paketarten wie VoIP, Chatnachrichten, E-Mails, ...implementiert, die entsprechend ihrer Dringlichkeit mehr Punkte bringen. Welche Paketart welche Punktzahl bringt, wird dem Indoor-Spieler in einer Legende angezeigt. die Punktevergabe für nicht erfolgreich geroutete Pakete erfolgt analog zu Level 2 |
| Outdoor-Client | <i>Levelanzeige</i> | Die Anzeige des Levels erfolgt in der Statusleiste, in welcher Batteriestand, Punkte etc. angezeigt werden mithilfe von goldenen/silbernen/bronzefarbenen 1/2/3 Icons. |

| | |
|---------------------|---|
| <i>Paketauswahl</i> | Die zu routenden Pakete werden in der Box für Hinweise am unteren Bildschirmrand angezeigt (siehe Abb. 2) Der Pakettyp ist erkennbar durch das Design des Pakets. Die Pakete sind absteigend nach Wartezeit sortiert und durch die Box kann gescrollt werden (siehe Abb. 2(a)). |
| Routing | Nachbarknoten werden wie in Level 2 durch das Senden von Hello-Nachrichten ausfindig gemacht. Pakete können auf verfügbare Nachbarn „gezogen“ werden, um sie an diese zu senden (siehe Abb. 2(b)). |
| Punktevergabe | Punkte werden abhängig von der verstrichenen Wartezeit sowie der Art des erfolgreich gesendeten Pakets vergeben. Nach erfolgreichem Routing erscheint ein Feedback (farblich (rot/gelb/grün) oder good/bad) auf der Karte. |

4. Umsetzung

Grundlage für die Weiterentwicklung von RouteMe bildete die Bachelorarbeit von Hendrik Geßner. Die bisher verwendeten Konzepte und die Implementierung der Level 1 und 2 wurden dabei weitestgehend unverändert gelassen. Die Umsetzung für Level 3 orientiert sich am vorhandenen Code und fügt sich somit in das bisherige Projekt ein. Lediglich die Funktionen, die für das dritte Level nicht mehr relevant oder in anderer Form genutzt werden mussten, wurden so geändert, dass sie nur für die unteren Level verfügbar sind.

Kernpunkt des neuen Levels ist das selbstständige Routen der Pakete von einem Knoten zum nächsten durch die mobilen Spieler, sowie das Zuweisen von Pakettypen und somit eine Priorisierung des Pakets durch die Indoor-Spieler. Es stehen insgesamt fünf Paketarten zur Verfügung, die häufig in mobilen Netzen gesendet werden. Die Punktevergabe erfolgt anhand der Dringlichkeit.

- Mail (1 Punkt)
- Html (2 Punkte)
- Chat (3 Punkte)
- Navigation (4 Punkte)
- VoIP (5 Punkte)

Im Folgenden werden die erforderlichen und umgesetzten Erweiterungen für das neue Level beschrieben.

4.1. Backend

Datenbankseitig kam es nur zu kleinen Änderungen. Die verwendete H2-Datenbank wurde beibehalten und auch die umgesetzte Tabellenstruktur blieb unverändert. Lediglich die Tabelle



(a) Scrollen durch verfügbare Pakete per Swipe-Geste (b) Ziehen der Pakete auf verfügbare Nachbar-knoten per Drag&Drop

Abbildung 2: Mobilclient mit Paketauswahl und Levelanzeige

`aodv_data_packets` wurde um das Feld `type` erweitert und somit den Paketen eine Paketart zugewiesen.

4.1.1. Algorithmus

Der Routingalgorithmus wurde um das manuelle Senden der Pakete durch die Outdoor-Spieler erweitert. Dazu wurde eine neue Funktion `sendPacket` über die Restschnittstelle bereitgestellt. Zu beachten ist hier, dass Pakete nur dann gesendet werden, wenn der Knoten auch tatsächlich eine Route zum Zielknoten besitzt. Dies wird serverseitig geprüft.

Für das Füllen seiner `RoutingTable` bleibt der Outdoor-Spieler weiterhin selbst zuständig. Dies wurde aus der Umsetzung von Level 2 übernommen, sodass der Knoten mit Absetzen eines Pings seine Nachbarn benachrichtigt und in dessen `RoutingTable` eingetragen wird. Das Ausbauen der Routen über mehrere Knoten hinweg wird anschließend über `RoutingRequests` gelöst.

Das Senden der `RoutingRequests` (RREQ) und `RoutingErrors` (RERR) wird weiterhin durch den Server unterstützt, sodass hier eine Intervention seitens der Spieler nicht notwendig ist.

Auch das Setzen des Routingstatus eines Pakets, welche für eine Information für den Indoor-Spieler dienen, bleibt durch den Server unterstützt. Folgende Stati sind derzeit verfügbar:

- `DATA_PACKET_STATUS_UNDERWAY`
- `DATA_PACKET_STATUS_ARRIVED`

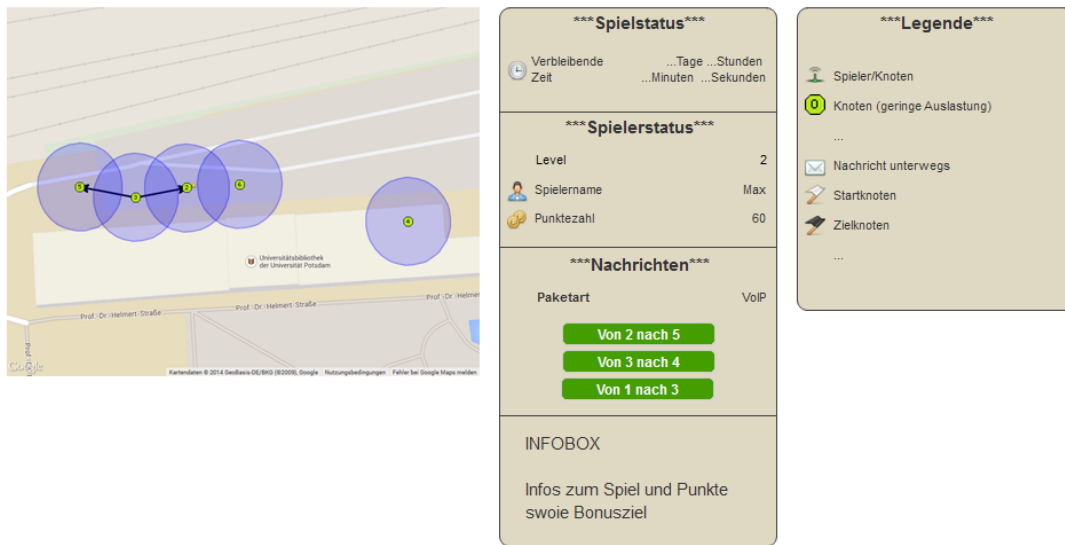


Abbildung 3: Indoor-Ansicht mit Paketauswahl und Levelanzeige Variante 1

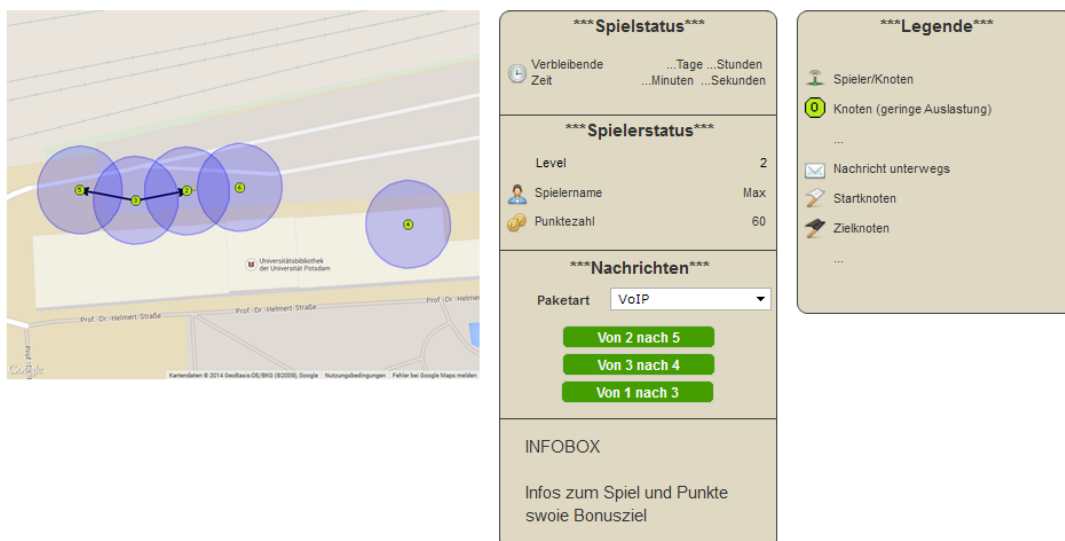


Abbildung 4: Indoor-Ansicht mit Paketauswahl und Levelanzeige Variante 2

- DATA_PACKET_STATUS_ERROR
- DATA_PACKET_STATUS_WAITING_FOR_ROUTE
- DATA_PACKET_STATUS_NODE_BUSY
- DATA_PACKET_STATUS_CANCELLED

In den Level 1 und 2 ändert sich diesbezüglich nichts, da hier das komplette Routen der Pakete überwacht und durchgeführt wird.

Für Level 3 wird der Status zunächst auf DATA_PACKET_STATUS_UNDERWAY gesetzt. Sollte innerhalb von 15 Sekunden das Paket nicht weitergesendet worden sein, wird der Status auf DATA_PACKET_STATUS_WAITING_FOR_ROUTE. Das ist jedoch noch keine Wertung bezüglich der Punktevergabe für Indoor- bzw. Outdoor-Spieler.

Für die automatisierten Algorithmusfunktionen wird vorab geprüft, welches Level der Spieler besitzt und die Funktion gegebenenfalls nicht ausgeführt.

Für den Fall, dass sich Spieler auf unterschiedlichen Level befinden, musste eine Lösung für das Setzen des Pakettyps gefunden werden. Relevant wird dieser Aspekt, wenn sich die Indoor-Spieler auf niedrigeren Level befinden und Outdoor-Spieler das dritte Level spielen, da diese anhand der Priorität der Pakete entscheiden müssen, welches als erstes gesendet werden soll. Diesem Problem wurde mit dem randomisierten Setzen von Paketarten begegnet, da es sich zunächst als einfachste Lösung herausstellte.

4.1.2. Punktevergabe

Die Rückmeldung für die Spieler über das erfolgreiche Routen von Nachrichten wird über Punkte realisiert. Somit ist ein direkter Vergleich der Spieler innerhalb einer Gruppe möglich. Die Punktevergabe für Level 3 setzt auf der Umsetzung für die unteren Level auf und berücksichtigt zusätzlich die vergebene Paketart.

Indoor:

Der Berechnung der Punkte für ein erfolgreich abgeschlossenes Routing liegen die Anzahl der durchgeführten Hops (*hopCount*), das Erreichen eines Bonusziels (*bonus* $\in \{0, 1\}$) und für Level 3 die Paketart (*packetType*) zugrunde:

$$\begin{aligned}
 points = & \left(\sum_{n=1}^{hopCount} n \right) * 10 \\
 & + \left(\sum_{n=1}^{hopCount} n \right) * 10 * 0,5 * bonus \\
 & + hopCount * packetType
 \end{aligned}$$

Outdoor:

Im Gegensatz zum Indoor-Spieler, bei dem für die Berechnung die komplette Route relevant ist, kann für die Outdoor-Spieler lediglich der aktuelle Versand des Pakets betrachtet werden. Dazu werden für jedes erfolgreiche Weiterleiten eines Pakets Basispunkte vergeben. Für Level 3 ist zusätzlich die Paketart (*packetType*) und die benötigte Zeit (*time*) für das Senden relevant.

Zu Beginn wird der 20-fache Paketwert angesetzt. Ausgehend von der Zeit, welches das Paket bei einem Knoten verweilt, wird ein Malus vergeben. Der Spieler hat 60 Sekunden Zeit das Paket zu versenden ohne Strafpunkte zu erhalten. Danach werden alle 20 Sekunden der 4-fache Paketwert

abgezogen:

$$\begin{aligned} points &= 100 \\ &+ 20 * packetType \\ &+ \lfloor (time - 60) / 20 \rfloor * 4 * packetType \end{aligned}$$

Somit wird sichergestellt, dass das schnelle Versenden von Paketen mit hoher Priorität vorteilhaft ist. Das Ignorieren dieser hingegen wird härter bestraft als bei Paketen mit geringerer Dringlichkeit.

Die Berechnung der Punkte orientiert sich hier an der Umsetzung in den vorhergehenden Levels und wurde so erweitert, dass sie sich sinnvoll in diese integriert.

4.2. Kommunikation

Für die Realisierung des manuellen Sendens durch die Outdoor-Spieler wurde die bereits vorhandene Rest-Schnittstelle genutzt. Zum einen wurde die neue Funktion **send_Packet** als post-Methode implementiert und verlangt Formular-Parameter (**@FormParam**) als Eingabe. Die benötigten Werte sind **packetId**, für das zu sendende Paket und **nextHopId** für den Knoten an den das Paket gesendet werden soll. Weitere Parameter sind nicht erforderlich, da der **SourceNode** und **DestinationNode** als Informationen im Paket mitgeführt werden.

Zum anderen wurde **get_game_status** erweitert, um die Vorraussetzung im Mobile Client zu schaffen. Diese Funktion liefert alle relevanten Informationen für den aktuellen Spielstatus sowie konkrete Fakten zur derzeitigen Situation des Knotens. Die Informationen werden als JSON-Response (siehe Listing 1) zur Verfügung gestellt, welcher um die Daten aus der **RoutingTable** und die Liste der aktuellen Nachbarn eines Knotens erweitert wurde. Weiterhin werden nun die zu sendenden Pakete übergeben.

Listing 1: JSON-Response zu **get_game_status**

```
{
  "node": {
    "id": 6,
    "name": "Mobile1",
    ...
  },
  "neighbours": {
    "7": {
      "id": 7,
      "name": "Mobile2"
      ...
    }
  },
  "packets": {
    "1": {
      "id": 1,
      "status": 4,
      "messageDescription": {
        "sourceNodeId": 6,
        "destinationNodeId": 7,
        "ownerId": 11
      },
      "type": 4,
      "ownerId": 11
      ...
    }
  },
  ...
  "routingTable": [
    {
      "nodeId": 6,
      "destinationId": 7,
      "nextHopId": 7,
      "hopCount": 1
    }
  ]
}
```

Auch die Javascript-Schnittstelle wurde an die neue Funktionalität angepasst. Diese wurde aus der Diplomarbeit übernommen und stellt Funktionen bereit um Events aus den GUIs an das

Backend weiterzuleiten. Dabei wurde eine neue Funktion implementiert, welche zusätzlich zur gewählten Route auch die Paketart weiterleitet.

4.3. Web-Ansichten

In der **Admin-Ansicht** wurde die Möglichkeit geschaffen, das Level der Spieler zu Ändern (siehe Abbildung 5). Somit besteht die Möglichkeit, die Spieler, je nach Vorwissen, im selben Spiel auf unterschiedlichen Level spielen zu lassen.

The screenshot shows a web interface for changing a player's level. It has a light blue header with the title 'Level'. Below the header, there are two dropdown menus: 'Spieler:' with 'Steffi' selected, and 'Schwierigkeitsgrad:' with 'einfach (geringe Anforderungen / starke Assistenz)' selected. To the right of these menus is a button labeled 'Level ändern'.

Abbildung 5: Änderung Spielerlevel

Zu beachten ist dabei, dass die Level nicht während eines laufenden Spiels geändert werden sollten, da sich die Ansicht für die Indoor-Spieler, speziell die Routenauswahl nicht aktualisiert. Dies wurde bewusst so umgesetzt, um weiterhin eine Vergleichbarkeit innerhalb der Spieler eines Levels zu gewährleisten. Zudem könnte die automatische Umstellung ohne vorherige Ankündigung beim Spieler für Verwirrung sorgen.

The screenshot shows a game status interface. At the top, it says '***Spielstatus***' and shows 'Verbleibende Spielzeit' as '0 Tage 0 Stunden 9 Minuten 45 Sekunden'. Below this is '***Spielerstatus***' showing 'Spielerlevel' as 3, 'Spielername' as Steffi, and 'Punktzahl' as 0. Underneath is '***Nachrichten***' with a 'Paketart:' dropdown set to 'Mail'. A list of 15 packets is shown in a green box with white text, each starting with 'Von' followed by a number and 'nach' followed by another number. At the bottom, there is a small information icon and a text block explaining the bonus goal node.

Abbildung 6: Paket und Routenwahl Indoor-Spieler

Die bedeutendste Änderung für die **Indoor-Ansicht** ist die Hinzunahme der Paketauswahl (siehe Abbildung 6). Da der Indoor-Spieler in Level 3 die Paketpriorisierung übernehmen muss, wurde mittels einer Dropdownliste eine Paketauswahl geschaffen. Diese umfasst insgesamt 15 Pakete, somit von jedem Typ drei, und ist hierarchisch geordnet. Sollte der Spieler also nicht eingreifen, wird immer das niedrigste verfügbare Paket gesendet. Um eine vergleichbare Situation zu schaffen,

werden die gesendeten Pakete aus der Liste entfernt und diese erst neu erstellt, wenn der Spieler alle angezeigten Pakete verwendet hat. Mit dieser Vorgehensweise werden die Spieler belohnt, welche schnell ihre Pakete versenden konnten.

Die gesamte Umsetzung des User Interfaces basiert auf der Implementierung von Level 2 und wurde bis auf die Paketauswahl und das Anzeigen der Spielerlevel nicht verändert. Für die Routenauswahl selbst wurde weiterhin die Realisierung von Level 2 verwendet. Aufgrund einer besseren Übersicht, wird die Anzeige nun aufsteigend anhand der IDs der Knoten sortiert. Diese Neuerung wurde auf Anregung von Testpersonen umgesetzt.

4.4. Mobiler Client

4.4.1. Levelunterscheidung

Da mit Level 3 auch die Möglichkeit implementiert wurde, verschiedenen Spielern eines Spiels unterschiedliche Level zuzuweisen, sollte für den Spieler erkennbar sein, auf welchem Level er spielt. Aufbauend auf dem erweiterten **GameStatus**-Objekt, welcher nun unter anderem das Level des Spielers enthält, wurde hierzu der Header des Mobilen UI um eine Anzeige des aktuellen Spielerlevels erweitert (vgl. Abbildung 7). Das Level kann vom Spieler selbst nicht verändert, sondern nur eingesehen werden.

An verschiedenen Stellen im Code wurde die Abfrage des Spiel-Levels durch die des Spieler-Levels ersetzt und ein Fall für Level 3 hinzugefügt.



Abbildung 7: Header des Mobilen UI: Spielerlevel, Punktestand, Anzahl der bekannten Nachbarn, verbleibende Spielzeit und aktueller Batteriestand

4.4.2. Pakete und Routing

Abbildung 8 stellt auf die wesentlichen Funktionen reduziert das Anzeigen und Senden von Paketen in der Applikation dar.

Der Client bezieht in 1. zunächst über die Rest-Schnittstelle das **gameStatus**-Objekt, in welcher Informationen zum Spiel – Routingtabelle, Spielzeit und Ähnliches sowie dem Spieler aktuell zugewiesene Pakete (im Feld **packets**) – enthalten sind. Besagte Pakete werden über das **PacketFooterFragment** im GUI dargestellt (vgl. Abbildung 9), wobei jedes Paket als eigener **ImageButton** implementiert ist. Die Pakete sind aufsteigend nach dem Zeitpunkt ihres Eintreffens bei dem Knoten sortiert. In den Paketen ist auch der jeweilige Zielknoten angegeben. Damit der mobile Knoten Pakete nicht wahllos weitersendet, ist serverseitig nur das Senden an Nachbarknoten erlaubt, die entweder Zielknoten des Pakets sind oder eine Route zu diesem kennen. Anhand seiner eigenen Routing-Tabelle werden für einen mobilen Knoten – sobald der Spieler einen Paket-Button antippt (3.) – alle erlaubten Nachbarknoten extrahiert (4.1) sowie die Karte in den Sendemodus versetzt. Das bedeutet, dass nun potentielle Nachbarknoten hervorgehoben (5.) werden (dies geschieht bei jedem Update des GameStatus erneut, also alle 300ms, solange ein Paket ausgewählt ist). Während der Sendemodus aktiv ist, wird ein Tippen auf den Bildschirm (6.) nicht als Ping, sondern als Sendeversuch des Pakets verarbeitet. Dies wird dem Spieler über einen textuellen Hinweis am unteren Bildschirmrand angezeigt (vgl. Abbildung 10). Beim Tippen auf die Karte wird der nächste gültige Nachbar (im Umkreis von 1m) vom angestippten Punkt aus berechnet. Daraufhin wird das Routen des Packets initiiert und bei Erfolg eine Feedback-Nachricht (*good*, *ok* oder *bad*) als Toast eingeblendet. Außerdem werden dem Spieler die erreichten Punkte gutgeschrieben.

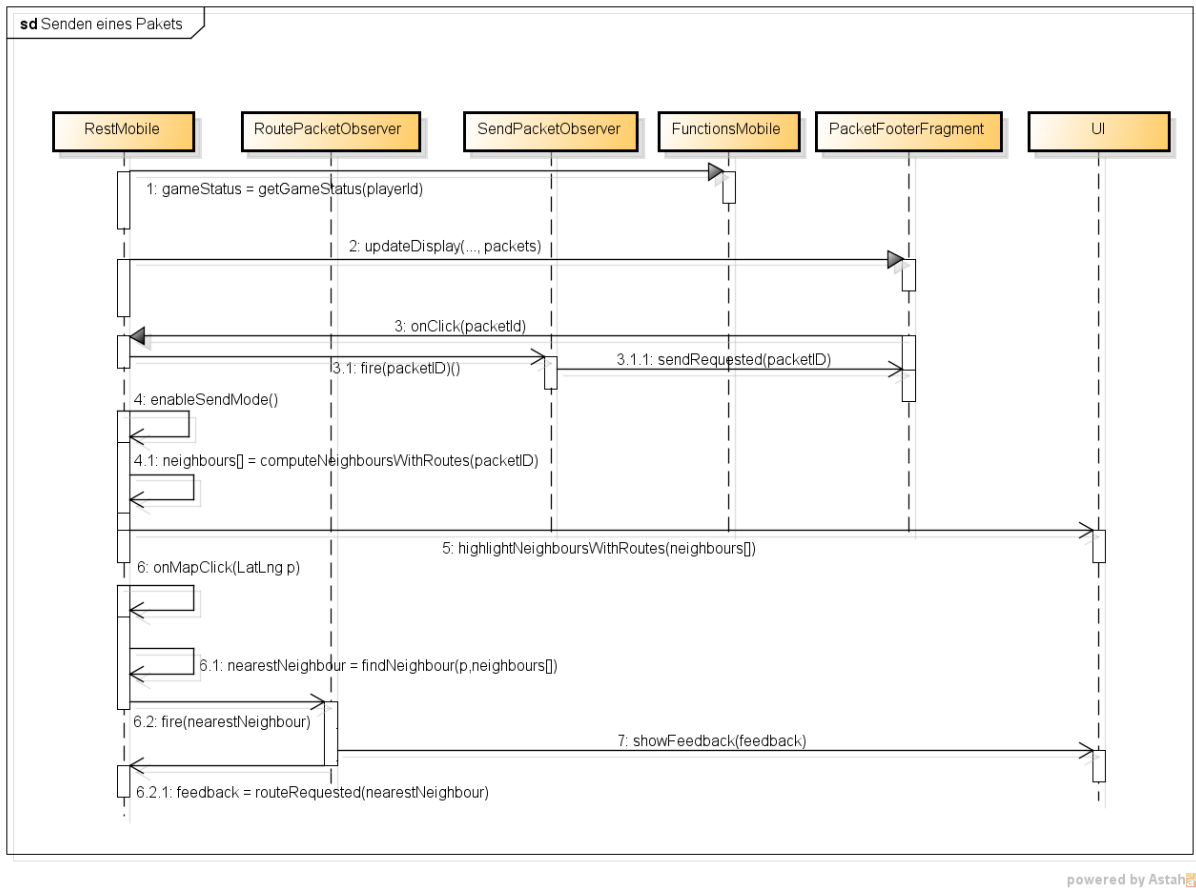


Abbildung 8: Sequenzdiagramm zum Senden eines Pakets



Abbildung 9: Anzeige der Pakete eines Spielers

4.4.3. Routing

Es war zunächst angedacht, das Routen von Paketen so zu realisieren, dass die Pakete per Swype-Geste⁸ auf den gewünschten Nachbarn bewegt werden (vgl. Kap. 3). Konkret sollten Pakete am Finger „kleben“, sobald sie angewählt wurden. Daraufhin sollten die Nachbarknoten, die Routen zum Ziel besitzen, hervorgehoben werden. Das Senden des Pakets wäre nach „Loslassen“ des Pakets auf einem gültigen Ziel initiiert worden. Dies erwies sich einerseits in der Umsetzung als schwierig, da die Google-Maps API keine direkte Funktion bietet, ein auf die Karte gezogenes Objekt zu erkennen. Andererseits war in einigen Versuchen auch erkennbar, dass durch die Geste mit dem Finger ein großer Teil der Karte verdeckt würde – gegebenenfalls könnte so sogar leicht ein Zielknoten übersehen werden.

⁸Ziehen mit dem Finger von einem Punkt des Displays auf einen anderen

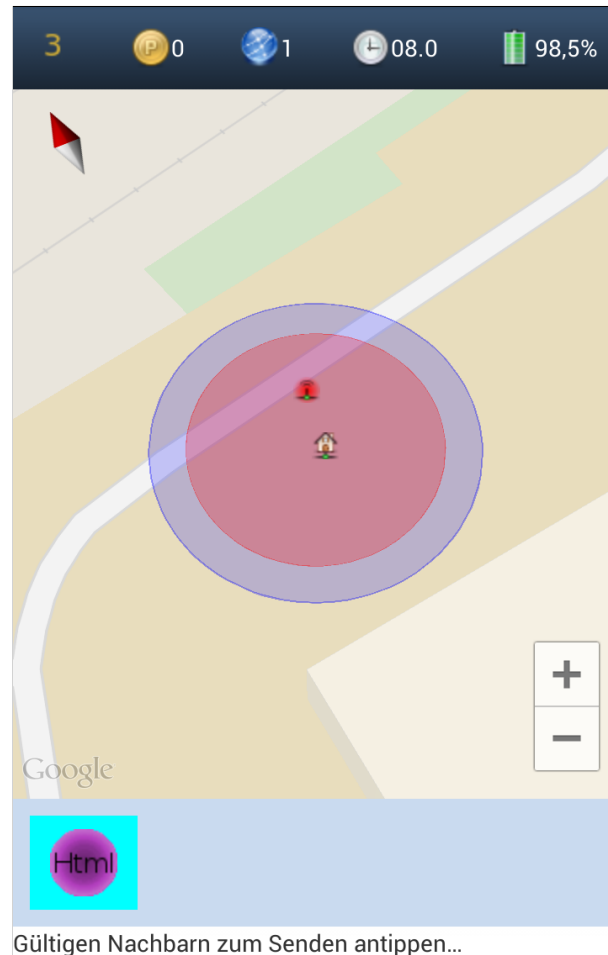


Abbildung 10: Ausgewähltes Paket, textueller Hinweis am unteren Bildschirmrand und hervor-
gehobener Nachbarknoten

5. Nutzeranleitung

5.1. Installation

Dieser Abschnitt erläutert kurz die Installation des Projektes. Das umfasst einerseits das Aufsetzen einer Entwicklungsumgebung und andererseits die Inbetriebnahme des Systems auf dem Servers sowie der Installation des Mobilen Clients auf den Endgeräten.

5.1.1. Entwicklungsumgebung

Für die Entwicklung kann **Eclipse Kepler (4.3)**⁹ als IDE verwendet werden. Dieses steht unter <https://www.eclipse.org/downloads/packages/release/Kepler/SR2> zum Download bereit.

Google Web Toolkit (2.6.0)¹⁰ existiert als Plugin für Eclipse, derzeit jedoch nur bis Version 4.3 (Kepler). Hier stehen alle Funktionen für die GWT und Android-Entwicklung zur Verfügung, wodurch keine zusätzlichen Tools benötigt werden.

Die Installation in Eclipse erfolgt über *Help* → *Install New Software*. Als Repository muss <http://dl.google.com/eclipse/plugin/4.3> eingetragen werden. Die im Folgenden angezeigte Auswahlmöglichkeiten sollten markiert werden:

⁹<https://www.eclipse.org/>

¹⁰<http://www.gwtproject.org/>

- Developer Tools
- Google App Engine Tools for Android
- Google Plugin for Eclipse
- GWT Designer for GPE
- SDKs

Der Quellcode steht als Projekt auf **Github** bereit und kann unter <https://github.com/slemcke/RouteMeBA/> genutzt werden. Nach erfolgreichem Download wird dieser über *File* → *Import...* → *General* → *Existing Projects into Workspace* in die eingerichtete Eclipse IDE importiert. Alternativ funktioniert auch der Import über *Git* → *Projects from Git*. Dabei sind die Projekte

- LocationLibrary
- NexplorerGWT
- RouteMeMobile
- RouteMeMobileTest
- RouteMeRestLibrary

zu wählen. Eventuell müssen bei einigen Projekten die Pfade der verwendeten Bibliotheken angepasst werden. Dies ist über die *Properties* der Projekte durchzuführen.

Das Projekt kann als WAR-Datei exportiert werden. Für die Erstellung kann das Eclipse-internen Ant genutzt werden. Alternative ist eine Installation von **Apache Ant**¹¹ notwendig. Der Download und eine Dokumentation ist auf der Homepage von Apache Ant verfügbar.

Für die Entwicklung wurde eine **PostgreSQL-Datenbank**¹² genutzt, um die Tabellen einzusehen. Die Konfiguration ist in `hibernate.cfg.xml` im Projekt NexplorerGWT hinterlegt und kann bei Bedarf eingebunden werden. Es ist jedoch möglich auch andere Datenbanken sowohl für den Test-, als auch für den Produktivbetrieb zu nutzen. Dazu muss lediglich die vorhandene Konfigurationsdatei angepasst werden.

5.1.2. Server

Für die Installation des Servers muss zunächst das Projekt **NexplorerGWT** mittels *Google* → *GWT compile* kompiliert werden. Die übersetzten Dateien befinden sich anschließend im Ordner `war`.

Eclipse bietet die Möglichkeit, das Projekt als Web Applikation auf den mitgelieferten Jetty-Server zu deployen. Die Indooransichten können dabei über folgende URLs aufgerufen werden:

- Admin: <http://localhost:8888/NexplorerGWT.html>
- Indoor: <http://localhost:8888/IndoorGWT.html>

Für die Nutzung externer Server kann über das bereitgestellt build-Script (`build.xml`) eine WAR-Datei (`R00T.war`) erzeugt werden.

Für einen Produktiveinsatz wurde der Server **routeme.cs.uni-potsdam.de** bereitgestellt. Die

¹¹<http://ant.apache.org/>

¹²<http://www.postgresql.org/>

Applikation kann über den installierten Tomcat¹³ verfügbar gemacht werden. Dazu muss die erzeugte Datei `R00T.war` in den Ordner `/usr/share/apache-tomcat-7.0.52/webapps` eingefügt werden und der Application-Server gegebenenfalls gestartet werden. Anschließend sind die User Interfaces über folgende URLs erreichbar:

- Admin: `http://routeme.cs.uni-potsdam.de:8080/NexplorerGWT.html`
- Indoor: `http://routeme.cs.uni-potsdam.de:8080/IndoorGWT.html`

5.1.3. Mobile-Client

Nach Herunterladen der APK-Datei über den Browser des Smartphones oder Kopieren der APK auf ein an einen PC angeschlossenes Smartphone kann diese installiert werden, indem gegebenenfalls über einen Dateibrowser an den entsprechenden Ort navigiert und die Datei angetippt wird. Hierfür ist es notwendig, vorher unter *Einstellungen* → *Sicherheit* → *Unbekannte Herkunft* die Option *Installation von Apps aus anderen Quellen als dem Play Store zulassen* zu aktivieren.

5.2. Kurzanleitung

Im Folgenden wird das Erstellen sowie die Verwaltung und das Spielen eines RouteMe-Spiels auf dem in der vorliegenden Arbeit behandelten Schwierigkeitsgrad dargestellt. Vorausgesetzt wird eine Installation des Servers, die unter **[Server-Url]** erreichbar ist.

5.2.1. Spielstart und -verwaltung

Nach Aufruf der URL **[Server-URL]/NexplorerGWT.html** in einem Browser können diverse Spielparameter angepasst (vgl. Abbildung 11) und unter „Schwierigkeitsgrad“ die Stufe „erhöht“ ausgewählt werden.

Einstellungen:

| | | |
|--------------------------------------|---|---------|
| Spieldauer: | <input type="text" value="10"/> | Minuten |
| Basisreichweite Knoten: | <input type="text" value="9"/> | Meter |
| Reichweite "Gegenstände einsammeln": | <input type="text" value="7"/> | Meter |
| Max. Anzahl Batterien: | <input type="text" value="19"/> | |
| Max. Anzahl Booster: | <input type="text" value="20"/> | |
| Schwierigkeitsgrad: | erhöht (erhöhte Anforderungen / minimale Assistenz) ▾ | |
| Routing-Protokoll | Ad-hoc On-Demand Distance Vector (AODV) ▾ | |

Erweiterte Einstellungen:

| | | | |
|-------------------------------|---|---|---|
| Spielfeld obere linke Ecke: | <input type="text" value="52.394317333573106"/> | : | <input type="text" value="13.129833250863612"/> |
| Spielfeld untere rechte Ecke: | <input type="text" value="52.39410782938052"/> | : | <input type="text" value="13.131571322305263"/> |

Intervallzeiten *:

| | | |
|---|-----------------------------------|---------------|
| Senden der Knotenposition an Server (deprecated): | <input type="text" value="500"/> | Millisekunden |
| Aktualisierung der Benutzeroberfläche: | <input type="text" value="500"/> | Millisekunden |
| Dauer eines Pings (AODV Level 2): | <input type="text" value="1000"/> | Millisekunden |

* Achtung! Intervallzeiten geben an mit welcher Frequenz versucht wird Anfragen an den Server zu senden. Anfragen können verworfen werden, wenn der Server oder die Netzwerkverbindung ausgelastet sind.

Abbildung 11: Erstellen eines Spiels

Nach Betätigung des Buttons *Spiel erstellen* ist das Spiel erstellt und im pausierten Modus. Der Ersteller hat nun einen Überblick über das Spielfeld, die teilnehmenden Spieler und vorhandene Gegenstände (vgl. Abbildung 12). Im Anschluss können sich (In- und Outdoor-) Spieler in dem Spiel anmelden.

¹³<http://tomcat.apache.org/>

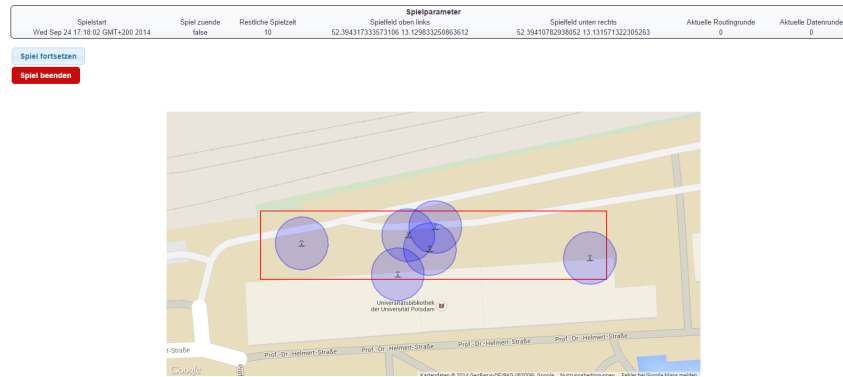


Abbildung 12: Erstelltes, noch pausiertes Spiel

Sobald sich ein Spieler angemeldet hat, kann der Administrator sein Spiellevel anpassen. Das Anpassen des Levels nach Spielstart kann zu Konflikten führen und wird nicht empfohlen. Sind alle erwarteten Spieler dem Spiel beigetreten, kann das Spiel über *Spiel fortsetzen* gestartet werden.

5.2.2. Client

Nach Aufruf der Oberfläche über die URL `[Server-URL]/NexplorerGWT.html` in einem Browser (Indoor) oder Start der App `RouteMeMobile` erscheint eine Login-Maske, wenn bereits ein Spiel vom Spielleiter erstellt wurde (vgl. Abschnitt 5.2.1). Andernfalls bekommt der Spieler lediglich einen Hinweis, dass noch kein Spiel vorhanden ist.

Der Spieler wählt zunächst einen Spielernamen. Während der Spielzeit kann der Spieler von jedem Client (bei Indoor von jedem Browser, bei Outdoor von jedem Smartphone mit der Applikation) aus über diesen Namen in sein Spiel zurückkehren, falls dies nötig ist (z. B. bei einem Wechsel des Smartphones oder Computers). Solange das Spiel noch nicht gestartet wurde, wird dem angemeldeten Spieler ein Overlay mit einem Hinweis angezeigt.

5.2.3. Mobile

Die Applikation kann über die bereitgestellte APK auf mobilen Endgeräten mit einem Android ab Version 4.2 installiert werden. Vor Spielstart sollte sichergestellt sein, dass die mobilen Knoten über das Spielfeld verteilt stehen, damit sich ihre Batterien nicht zu schnell entleeren.¹⁴

5.3. Spielverlauf

5.3.1. Indoor

Sobald das Spiel gestartet wurde, wird dem Indoor-Spieler die in Abbildung 6 gezeigte View dargestellt. Im linken Bereich befindet sich eine **Karte**, auf der sämtliche Mobile Knoten dargestellt werden. Rechts daneben kann der Spieler **Pakettypen** auswählen und diese einer der zufällig generierten, angezeigten Routen zuweisen. Bevor das Paket nicht erfolgreich zugestellt oder der Sendevorgang abgebrochen wurde, kann kein neues Paket gesendet werden. Ein **Abbruch** des Sendevorgangs ist nach Sendestart über den entsprechenden Button möglich. Außerdem werden dem Spieler im rechten Bildschirmbereich weitere Informationen zum Spiel angezeigt.

¹⁴Die Clients senden Hello-Nachrichten, um ihren Nachbarn ihre Anwesenheit zu signalisieren. Eine Hello-Nachricht kostet den auslösenden Knoten für jeden erreichbaren Nachbarknoten Strom.

5.3.2. Mobile

Nach Spielstart wird im Großteil des Smartphone-Bildschirms eine **Karte** angezeigt, die sich um den Spieler zentriert. Abbildung 10 in Kapitel 4.4.3 zeigt dies Ansicht der App während eines laufenden Spiels. Zwei Kreise um den Mittelpunkt verdeutlichen die **Sende- (blau) und Sammelreichweite (rot)** des Knotens (vgl. Abbildung ??). Auf der Karte werden andere mobile Knoten sowie Gegenstände (Batterien und Booster), die durch Schütteln des Smartphones **eingesammelt werden können**, angezeigt. Der Spieler kann sich innerhalb des definierten Spielfeldes frei bewegen. Durch Tippen auf den Bildschirm sendet der Spieler eine **Ping-Nachricht**, wodurch er Knoten in der Nähe seine Verfügbarkeit signalisiert.

Im oberen Bereich des Screens werden (in dieser Reihenfolge) das Level, der Punktestand, die Anzahl der bekannten Nachbarn, die verbleibende Spielzeit sowie der Batteriestand angezeigt, wie auch in Abbildung 7 in Kapitel 4.4.3.

Sobald dem Spieler – von einem anderen Outdoor- oder einem Indoor-Spieler – ein **Paket** zugewiesen wurde, wird es ihm im unteren Bildschirmbereich angezeigt. Möchte der Spieler ein Paket **senden**, tippt er es an. Die Karte befindet sich nun im **Sende-Modus**, das heißt alle Nachbarn, die potentielle Empfänger für das Paket sind, werden hervorgehoben. Außerdem wird ein **Tippen auf die Karte** nun als Sendeversuch interpretiert, wobei versucht wird, das ausgewählte Paket an den angetippten Knoten zu senden. Bei Erfolg bekommt der Spieler **Punkte** gutgeschrieben und ein Feedback (good/ok/bad) korrespondierend zur erreichten Punktzahl am unteren Bildschirmrand eingeblendet.

5.4. Spielende

Wenn der **Akkustand** für einen mobilen Knoten auf 0% fällt, ist für den betroffenen Spieler das Spiel vorbei.

Nach Ablauf der vorab eingestellten Spielzeit oder wenn weniger als zwei mobile Knoten am Spiel teilnehmen¹⁵ **endet** das Spiel.

Der Spielersteller kann in der Adminansicht den Punktestand der Spieler einsehen und so einen **Gewinner** küren.

6. Fazit/Ausblick

6.1. Fazit

In der vorliegenden Arbeit wurde zunächst die Intention des pervasiven Android-Lernspiels *RouteMe* (siehe Kapitel 1: Einleitung) vorgestellt. Der bisher implementierte Softwarestand inklusive der Level 1 und 2 wurde aus sowohl die Implementierung als auch die Intention betreffend untersucht und ausführlich dargestellt (siehe Kapitel 2: Ausgangssituation). Anhand der gewonnenen Erkenntnisse wurde ein Konzept für die Implementierung des dritten Levels erarbeitet und umfassend erläutert (siehe Kapitel 3: Konzeption). Aufbauend auf dieser Konzeption – mit einigen Änderungen, die aus unerwarteten Problemen und neuen Erkenntnissen entstanden – wurde im Folgenden die Applikation um Level 3 ergänzt (siehe Kapitel 4: Umsetzung). Elementar für die Umsetzung des neuen Levels war es, die Assistenz der Software zu verringern und dem Nutzer mehr Verantwortung und Handlungsspielraum zu bieten. Die Indoor-Spieler müssen nun selbst neu eingeführte Pakettypen für die Priorisierung der versendeten Nachrichten auswählen. Der Outdoor-Spieler bekommt an ihn gesendete Pakete sowie deren Typen angezeigt und muss eigenständig Nachbarn finden, die gültige Routen für das Weitersenden der Pakete besitzen, um diese weiterzugeben und Punkte zu erhalten.

¹⁵ Da das Routen bei nur einem oder gar keinem Knoten unsinnig wäre, wird in diesem Fall das Spiel vorzeitig beendet.

Um die potentielle Weiterentwicklung sowie die Verwendung der App durch den Endnutzer zu erleichtern, findet sich in dieser Dokumentation außerdem eine Installations- und Benutzeranleitung in Kapitel 5: Nutzeranleitung. Im Folgenden wird auf Ergebnisse eines Nutzertests eingegangen sowie im Abschluss ein Ausblick auf bekannte Schwierigkeiten und mögliche Änderungen und Verbesserungen gegeben.

6.2. Nutzertest

Es wurde ein Nutzertest mit insgesamt acht Probanden durchgeführt, von denen fünf als Outdoor- und drei als Indoor-Spieler fungierten. Nach mehreren Testrunden wurde den Testern der User Experience Questionnaire¹⁶ vorgelegt und von ihnen ausgefüllt. Nähere Informationen zum UEQ und dessen Auswertung können der Webseite und den dort beinhalteten wissenschaftliche Arbeiten entnommen werden. Im Folgenden sollen lediglich einige auffallende Ergebnisse der Fragebogenauswertung umrissen werden. Jedoch sollten diese mit Vorsicht betrachtet werden, da die Auswertung bei drei bzw. fünf Testpersonen als nicht besonders aussagekräftig angesehen werden kann. Der Fragebogen sowie die erhobenen Datensätze finden sich in Anhang A: Fragebogen und B: Daten. In Anhang C: Auswertung finden sich Teile der Auswertung aus der Excel-Vorlage des UEQ.

Die Items des Fragebogens sind den Kategorien Attraktivität, Durchschaubarkeit, Effizienz, Steuerbarkeit, Stimulation und Originalität zuzuordnen. In den Abbildungen 20, 16 und 18 sind im Bereich **Skala** transformierte Werte für diese Kategorien berechnet, wobei die Skalen von -3 bis 3 reichen. Werte $> 0,8$ sind hierbei als positiv zu werten, Werte $< -0,8$ als negativ und solche dazwischen als neutral. Auffallend ist hier, dass in beiden Gruppen (In- und Outdoor) sowie in der Gesamtbetrachtung der Testgruppe die Werte für Originalität und Stimulation in der Skala recht hoch sind. Auch fällt auf, dass vor allem bei der Bewertung durch die Outdoor-Spieler der Wert für Steuerbarkeit negativ ist. Hierbei sollte speziell beachtet werden, dass wie in Abbildung 18 ersichtlich ist, die Paare *unberechenbar/berechenbar* und *erwartungskonform/nicht erwartungskonform* deutlich in die negative Richtung ausschlagen. Dies kann einerseits damit erklärt werden, dass zum Zeitpunkt des Tests noch kleinere Fehler mit der Applikation auftraten, andererseits aber auch damit, dass das Spiel durch die randomisierte Generierung von Items sowie das für den Spieler undurchsichtige Agieren (z. B. Pingen) anderer Beteiligter als *unberechenbar* und *nicht erwartungskonform* empfunden werden kann, was nicht unbedingt als negativ sondern vielmehr als zum Spielkonzept gehörend zu werten ist.

6.3. Ausblick

Im Folgenden werden einige bekannte Probleme und Unklarheiten im Spiel allgemein dargestellt und Lösungsansätze umrissen. Diese Probleme sind größtenteils aus Nutzertests hervorgegangen, die mehrfach mit Mitarbeitern des Lehrstuhls durchgeführt wurden und sowohl der technischen als auch der konzeptionellen Validierung der Applikation dienten.

6.3.1. GPS

Die App wurde in der vorangegangenen Arbeit von Hendrik Gefner [Ges13] auf Geräten des Modells „Google Galaxy Nexus (Samsung)“ getestet. In den Benutzertests zu der vorliegenden Arbeit fiel auf, dass auf Geräten des Galaxy Nexus die GPS-Position teilweise deutlich schlechter bestimmt wurde, als mit anderen Geräten wie z. B. dem „Google Nexus 4 (LG)“ oder dem „Motorola Moto G (Motorola)“.

Diese Varianz in der GPS-Bestimmung lässt sich eventuell dadurch erklären, dass die neueren Geräte – die zuletzt genannten, auf denen die Positionierung besser funktionierte – allesamt

¹⁶<http://www.ueq-online.org/>

GLONASS-fähig¹⁷ sind. Vor allem bei Bewölkung liefert GLONASS häufig bessere Ergebnisse als GPS.

Um die App dennoch auf einem Großteil der derzeit genutzten Smartphones spielfähig zu machen, werden im Weiteren verschiedene Ansätze diskutiert, die teilweise von Testnutzern der Applikation angeregt wurden.

Als eine Möglichkeit wurde angedacht, den User mit einem falschen GPS-Standort zu unterstützen, indem man ihm anzeigt, in welcher Richtung für seine Position das Spielfeld liegt. Somit hätte der Nutzer die Möglichkeit, an einem anderen Ort als die Mitspieler dennoch innerhalb des virtuellen Spielfelds zu sein und mitzuspielen. Um den Nachteil auszugleichen, der durch eine schlechte GPS-Positionierung entsteht, könnte man Möglichkeiten andenken wie z.B. mehr Batterien, höhere Punkte oder Ähnliches. Somit würde der Spielspaß nicht verlorengehen.

Des Weiteren wäre es möglich, über Heuristiken, die beispielsweise mögliche Bewegungen hinzuziehen, die Standortgenauigkeit zu erhöhen.

6.3.2. Pervasivität

RouteMe erreicht Pervasivität durch die Nutzung von Smartphones und der Sensoren und Funktionen dieser. Ein Aspekt, der häufig für Probleme sorgte und den Spielfluss unterbrach, war die schwere Lesbarkeit eines Smartphone-Displays in der prallen Sonne (wohingegen bei Wolken die GPS-Ortung ein Problem darstellte). Eine Möglichkeit zur Lösung dieses Problems wäre die Portierung der Software auf eine andere Hardware wie z.B. eine AR-Brille. Auch denkbar wäre eine ausgedehntere Nutzung der vom Smartphone gegebenen Möglichkeiten (z. B. die Einbindung von Audio), um die Pervasivität zu erhöhen.

6.3.3. Karte

Für viele Spieler war der gewählte Kartenausschnitt auf dem Smartphone zu klein und die angezeigten Informationen teilweise nicht intuitiv genug. Die Größe des Kartenausschnitts wurde bewusst gewählt, um den Sende- und Empfangsradius der mobilen Knoten darzustellen. Denkbar wäre eine optische Begrenzung des Spielfeldes auf dem mobilen Endgerät - beispielsweise durch Ausgrauen der Fläche, die außerhalb liegt. Ebenfalls möglich wäre das Anzeigen von mehr Informationen, z. B. einer Legende, der Namen anderer Spieler, Aufgrund der Displaygröße sollte hierbei allerdings auch in Betracht gezogen werden, die Applikation um besagte Informationen erweitert explizit für Tablets bereitzustellen.

6.3.4. Spielfeld

Während des Spiels ist aufgefallen, dass die Outdoor-Spieler sich nur in einem Teilbereich des Spielfeldes, welches voreingestellt eine längliche Form besitzt, bewegten. Dadurch wurden gerade in diesem Bereich alle Items – Batterien und Booster – sehr schnell eingesammelt und den Spielern war es nur noch sehr schwer möglich, nicht vorzeitig durch Batterieentladung auszufallen. Grund für diese Ballung der Spieler ist jedoch die notwendige Nähe zu anderen Knoten, um gültige Einträge in der Routingtabelle zu erhalten und somit Pakete versenden zu können.

Es wäre also sinnvoller, das Spielfeld quadratischer zu gestalten, um eine bessere Ausnutzung des Bereichs zu gewährleisten. Da sich der Ort aus dem im vorhergehenden Projekt durchgeführten Messungen zu GPS-Genauigkeiten am Campus Griebnitzsee ergab und die Umgebung am gewählten Ort die Spielfeldform bestimmt, müsste man hier entweder eine neue Messung durchführen oder auf andere Gelände ausweichen.

¹⁷GLONASS ist ein russisches Navigationssystem, das technisch ähnlich funktioniert wie GPS und ebenfalls über Satelliten realisiert ist. Unterstützt ein Chip die Positionsbestimmung über GLONASS und GPS, so nutzt das Smartphone automatisch beide Möglichkeiten, um einen möglichst exakten Standort zu ermitteln [Wik14]

6.3.5. Administration

Die Adminansicht, welche aus den vorhergehenden Projekten zum großen Teil übernommen wurde, bietet einen Überblick über die gesamte Spielsituation, die Namen und Punkte der Spieler sowie die Position der Knoten (Outdoor-Spieler). Die Übersicht der Knoten wird aufgrund ständiger Positionsänderungen häufig aktualisiert, was zu einem sehr unruhigen Anzeigeführer führt. Um dem entgegenzuwirken, ist eine Sortierung nach Id oder Namen der Knoten sinnvoll.

Zudem werden ausgefallene Spieler entfernt, wodurch die Auswertung nach dem Spiel erschwert wird. Daher sollten alle beteiligten Spieler weiterhin gelistet werden.

6.3.6. Paketwahl (Indoor)

Die Auswahl der Pakete erfolgt derzeit über eine DropDown-Liste und muss vor der Wahl der Route durchgeführt werden, da andernfalls immer das oberste Paket automatisch gewählt wird. Im Test wurde darauf hingewiesen, dass dies zwar ein probates Mittel darstellt, jedoch leicht den Spielfluss verlangsamt, da zunächst die Liste ausgeklappt werden muss. Auch ist anders kein kompletter Überblick über die verbliebenen Pakete möglich.

Als Alternativen wurden Buttons oder Radioboxen genannt. Diese sollen dann in ähnlicher Weise wie die Liste der Routen dargestellt werden. Dies hätte den Vorteil, dass ein kompletter Überblick über alle noch verfügbaren Pakete möglich und lediglich eine Aktion für die Auswahl notwendig ist.

6.3.7. Verwerfen von Paketen (Outdoor)

Um den Outdoor-Spieler weiter zu fordern und seine Eigeninitiative zu erhöhen, wäre es denkbar, ihm weitere Aktionen des Algorithmus zu übertragen. Beispielsweise kann die Entscheidung über das Verwerfen eines Paketes in der Verantwortung des Knotenspielers liegen, indem diesem beim Antippen des Pakets mitgeteilt wird, dass keine Route vorhanden ist. Der Spieler muss nun eruieren, ob es sich um eine kurzfristige Störung handelt oder der Zielknoten ausgefallen ist und das Verwerfen des Pakets sinnvoll wäre.

Literatur

- [DG13] DEHNE, Julian ; GESSNER, Hendrik: RouteMe - Dokumentation des Refactoring. April 2013
- [Elk08] ELKSTEIN, M.: Learn REST: A Tutorial. <http://rest.elkstein.org/>, Februar 2008. – [Online; letzter Zugriff 19.07.2014]
- [Ges13] GESSNER, Hendrik: Erweiterte Konzeption und Implementierung eines Lernspiels über Routing-Protokolle für den Informatik-Unterricht. Oktober 2013
- [GWT] GWT: Google Web Toolkit. <http://www.gwtproject.org/>, . – [Online; letzter Zugriff 19.07.2014]
- [Net03] NETWORK WORKING GROUP: RFC 3561. <https://www.ietf.org/rfc/rfc3561.txt/>, 2003. – [Online; letzter Zugriff 24.09.2014]
- [red] REDHAT AND JBOSSDEVELOPER: Hibernate. Everything data. <http://hibernate.org/>, . – [Online; letzter Zugriff 19.07.2014]
- [Wik14] WIKIMEDIA FOUNDATION: GLONASS. <http://en.wikipedia.org/w/index.php?title=GLONASS&oldid=624078109>, 2014. – [Online; letzter Zugriff 24.09.2014]

A. Fragebogen

Bitte geben Sie Ihre Beurteilung ab.

Um das Produkt zu bewerten, füllen Sie bitte den nachfolgenden Fragebogen aus. Er besteht aus Gegensatzpaaren von Eigenschaften, die das Produkt haben kann. Abstufungen zwischen den Gegensätzen sind durch Kreise dargestellt. Durch Ankreuzen eines dieser Kreise können Sie Ihre Zustimmung zu einem Begriff äußern.

Beispiel:

| | | | | | | | | |
|-----------|-----------------------|----------------------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-------------|
| attraktiv | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | unattraktiv |
|-----------|-----------------------|----------------------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-------------|

Mit dieser Beurteilung sagen Sie aus, dass Sie das Produkt eher attraktiv als unattraktiv einschätzen.

Entscheiden Sie möglichst spontan. Es ist wichtig, dass Sie nicht lange über die Begriffe nachdenken, damit Ihre unmittelbare Einschätzung zum Tragen kommt.

Bitte kreuzen Sie immer eine Antwort an, auch wenn Sie bei der Einschätzung zu einem Begriffspaar unsicher sind oder finden, dass es nicht so gut zum Produkt passt.

Es gibt keine „richtige“ oder „falsche“ Antwort. Ihre persönliche Meinung zählt!

Bitte geben Sie nun Ihre Einschätzung des Produkts ab. Kreuzen Sie bitte nur einen Kreis pro Zeile an.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | | |
|-------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-------------------------|----|
| unerfreulich | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | erfreulich | 1 |
| unverständlich | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | verständlich | 2 |
| kreativ | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | phantasielos | 3 |
| leicht zu lernen | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | schwer zu lernen | 4 |
| wertvoll | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | minderwertig | 5 |
| langweilig | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | spannend | 6 |
| uninteressant | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | interessant | 7 |
| unberechenbar | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | voraussagbar | 8 |
| schnell | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | langsam | 9 |
| originell | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | konventionell | 10 |
| behindernd | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | unterstützend | 11 |
| gut | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | schlecht | 12 |
| kompliziert | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | einfach | 13 |
| abstoßend | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | anziehend | 14 |
| herkömmlich | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | neuartig | 15 |
| unangenehm | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | angenehm | 16 |
| sicher | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | unsicher | 17 |
| aktivierend | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | einschläfernd | 18 |
| erwartungskonform | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | nicht erwartungskonform | 19 |
| ineffizient | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | effizient | 20 |
| übersichtlich | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | verwirrend | 21 |
| unpragmatisch | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | pragmatisch | 22 |
| aufgeräumt | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | überladen | 23 |
| attraktiv | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | unattraktiv | 24 |
| sympathisch | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | unsympathisch | 25 |
| konservativ | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | innovativ | 26 |

B. Daten

| Items | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------|----|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 6 | 6 | 5 | 5 | 1 | 5 | 6 | 2 | 2 | 3 | 4 | 2 | 6 | 6 | 5 | 6 | 6 | 2 | 5 | 3 | 2 | 3 | 2 | 3 | 2 | 6 |
| 6 | 6 | 2 | 2 | 2 | 6 | 6 | 4 | 1 | 3 | 5 | 2 | 6 | 5 | 4 | 6 | 4 | 1 | 3 | 6 | 3 | 4 | 1 | 2 | 2 | 5 |
| 4 | 22 | 2 | 5 | 4 | 4 | 5 | 1 | 2 | 4 | 3 | 4 | 3 | 4 | 6 | 4 | 4 | 2 | 5 | 4 | 3 | 4 | 2 | 4 | 4 | 6 |
| 6 | 6 | 2 | 1 | 3 | 6 | 7 | 2 | 1 | 1 | 4 | 2 | 5 | 6 | 7 | 5 | 5 | 1 | 6 | 4 | 2 | 4 | 2 | 3 | 2 | 7 |
| 3 | 3 | 3 | 6 | 4 | 5 | 6 | 3 | 4 | 2 | 3 | 5 | 2 | 4 | 5 | 3 | 4 | 4 | 6 | 3 | 5 | 3 | 3 | 5 | 5 | 5 |
| 6 | 3 | 1 | 3 | 2 | 5 | 6 | 3 | 1 | 1 | 4 | 2 | 2 | 4 | 7 | 5 | 4 | 2 | 5 | 4 | 5 | 4 | 4 | 3 | 3 | 7 |
| 5 | 6 | 2 | 1 | 3 | 5 | 5 | 2 | 2 | 2 | 5 | 3 | 5 | 6 | 6 | 5 | 4 | 3 | 4 | 5 | 3 | 5 | 1 | 3 | 3 | 5 |
| 7 | 5 | 2 | 2 | 2 | 6 | 6 | 2 | 3 | 2 | 6 | 1 | 6 | 2 | 6 | 6 | 2 | 2 | 3 | 6 | 3 | 6 | 3 | 3 | 3 | 6 |

Abbildung 13: Daten aller Teilnehmer

| Items | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 6 | 3 | 1 | 3 | 2 | 5 | 6 | 3 | 1 | 1 | 4 | 2 | 2 | 4 | 7 | 5 | 4 | 2 | 5 | 4 | 5 | 4 | 4 | 3 | 3 | 7 |
| 5 | 6 | 2 | 1 | 3 | 5 | 5 | 2 | 2 | 2 | 5 | 3 | 5 | 6 | 6 | 5 | 4 | 3 | 4 | 5 | 3 | 5 | 1 | 3 | 3 | 5 |
| 7 | 5 | 2 | 2 | 2 | 6 | 6 | 2 | 3 | 2 | 6 | 1 | 6 | 2 | 6 | 6 | 2 | 2 | 3 | 6 | 3 | 6 | 3 | 3 | 3 | 6 |

Abbildung 14: Daten der Indoor-Teilnehmer

| Items | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------|----|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 6 | 6 | 5 | 5 | 1 | 5 | 6 | 2 | 2 | 3 | 4 | 2 | 6 | 6 | 5 | 6 | 6 | 2 | 5 | 3 | 2 | 3 | 2 | 3 | 2 | 6 |
| 6 | 6 | 2 | 2 | 2 | 6 | 6 | 4 | 1 | 3 | 5 | 2 | 6 | 5 | 4 | 6 | 4 | 1 | 3 | 6 | 3 | 4 | 1 | 2 | 2 | 5 |
| 4 | 22 | 2 | 5 | 4 | 4 | 5 | 1 | 2 | 4 | 3 | 4 | 3 | 4 | 6 | 4 | 4 | 2 | 5 | 4 | 3 | 4 | 2 | 4 | 4 | 6 |
| 6 | 6 | 2 | 1 | 3 | 6 | 7 | 2 | 1 | 1 | 4 | 2 | 5 | 6 | 7 | 5 | 5 | 1 | 6 | 4 | 2 | 4 | 2 | 3 | 2 | 7 |
| 3 | 3 | 3 | 6 | 4 | 5 | 6 | 3 | 4 | 2 | 3 | 5 | 2 | 4 | 5 | 3 | 4 | 4 | 6 | 3 | 5 | 3 | 3 | 5 | 5 | 5 |

Abbildung 15: Daten der Outdoor-Teilnehmer

C. Auswertung

C.1. Indoor

| Item | MW | Varianz | Std. Abw. | Anzahl | Links | Rechts | Skala |
|------|------|---------|-----------|--------|-------------------|-------------------------|-------------------|
| 1 | 2,0 | 1,0 | 1,0 | 3 | unerfreulich | erfreulich | Attraktivität |
| 2 | 0,7 | 2,3 | 1,5 | 3 | unverständlich | verständlich | Durchschaubarkeit |
| 3 | 2,3 | 0,3 | 0,6 | 3 | kreativ | phantasielos | Originalität |
| 4 | 2,0 | 1,0 | 1,0 | 3 | leicht zu lernen | schwer zu lernen | Durchschaubarkeit |
| 5 | 1,7 | 0,3 | 0,6 | 3 | wertvoll | minderwertig | Stimulation |
| 6 | 1,3 | 0,3 | 0,6 | 3 | langweilig | spannend | Stimulation |
| 7 | 1,7 | 0,3 | 0,6 | 3 | uninteressant | interessant | Stimulation |
| 8 | -1,7 | 0,3 | 0,6 | 3 | unberechenbar | voraussagbar | Steuerbarkeit |
| 9 | 2,0 | 1,0 | 1,0 | 3 | schnell | langsam | Effizienz |
| 10 | 2,3 | 0,3 | 0,6 | 3 | originell | konventionell | Originalität |
| 11 | 1,0 | 1,0 | 1,0 | 3 | behindernd | unterstützend | Steuerbarkeit |
| 12 | 2,0 | 1,0 | 1,0 | 3 | gut | schlecht | Attraktivität |
| 13 | 0,3 | 4,3 | 2,1 | 3 | kompliziert | einfach | Durchschaubarkeit |
| 14 | 0,0 | 4,0 | 2,0 | 3 | abstoßend | anziehend | Attraktivität |
| 15 | 2,3 | 0,3 | 0,6 | 3 | herkömmlich | neuartig | Originalität |
| 16 | 1,3 | 0,3 | 0,6 | 3 | unangenehm | angenehm | Attraktivität |
| 17 | 0,7 | 1,3 | 1,2 | 3 | sicher | unsicher | Steuerbarkeit |
| 18 | 1,7 | 0,3 | 0,6 | 3 | aktivierend | einschläfernd | Stimulation |
| 19 | 0,0 | 1,0 | 1,0 | 3 | erwartungskonform | nicht erwartungskonform | Steuerbarkeit |
| 20 | 1,0 | 1,0 | 1,0 | 3 | ineffizient | effizient | Effizienz |
| 21 | 0,3 | 1,3 | 1,2 | 3 | übersichtlich | verwirrend | Durchschaubarkeit |
| 22 | 1,0 | 1,0 | 1,0 | 3 | unpragmatisch | pragmatisch | Effizienz |
| 23 | 1,3 | 2,3 | 1,5 | 3 | aufgeräumt | überladen | Effizienz |
| 24 | 1,0 | 0,0 | 0,0 | 3 | attraktiv | unattraktiv | Attraktivität |
| 25 | 1,0 | 0,0 | 0,0 | 3 | sympathisch | unsympathisch | Attraktivität |
| 26 | 2,0 | 1,0 | 1,0 | 3 | konservativ | innovativ | Originalität |

| Skalen | |
|-------------------|-------|
| Attraktivität | 1,222 |
| Durchschaubarkeit | 0,833 |
| Effizienz | 1,333 |
| Steuerbarkeit | 0,000 |
| Stimulation | 1,583 |
| Originalität | 2,250 |

Bewertung pro Item

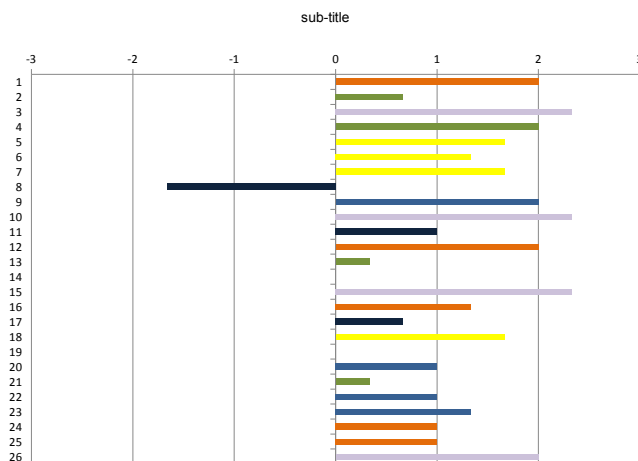


Abbildung 16: Auswertung zu den einzelnen Items und Skala der Kategorien der Indoor-Gruppe

| Attraktivität | | Durchschaubarkeit | | Effizienz | | Steuerbarkeit | | Stimulation | | Originalität | |
|---------------|-------------|-------------------|-------------|-----------|-------------|---------------|-------------|-------------|-------------|--------------|-------------|
| Items | Korrelation | Items | Korrelation | Items | Korrelation | Items | Korrelation | Items | Korrelation | Items | Korrelation |
| 1, 12 | 1,00 | 2, 4 | 0,98 | 9, 20 | -1,00 | 8, 11 | -0,87 | 5, 6 | 0,50 | 3, 10 | 1,00 |
| 1, 14 | -1,00 | 2, 13 | 0,84 | 9, 22 | -1,00 | 8, 17 | -0,50 | 5, 7 | 1,00 | 3, 15 | 1,00 |
| 1, 16 | 0,87 | 2, 21 | 0,94 | 9, 23 | -0,33 | 8, 19 | -0,87 | 5, 18 | 1,00 | 3, 26 | 0,87 |
| 1, 24 | #DIV/0! | 4, 13 | 0,72 | 20, 22 | 1,00 | 11, 17 | 0,87 | 6, 7 | 0,50 | 10, 15 | 1,00 |
| 1, 25 | #DIV/0! | 4, 21 | 0,87 | 20, 23 | 0,33 | 11, 19 | 1,00 | 6, 18 | 0,50 | 10, 26 | 0,87 |
| 12, 14 | -1,00 | 13, 21 | 0,97 | 22, 23 | 0,33 | 17, 19 | 0,87 | 7, 18 | 1,00 | 15, 26 | 0,87 |
| 12, 16 | 0,87 | DK | 0,89 | DK | -0,11 | DK | 0,08 | DK | 0,75 | DK | 0,93 |
| 12, 24 | #DIV/0! | Alpha | 0,97 | Alpha | -0,68 | Alpha | 0,27 | Alpha | 0,92 | Alpha | 0,98 |
| 12, 25 | #DIV/0! | | | | | | | | | | |
| 14, 16 | -0,87 | | | | | | | | | | |
| 14, 24 | #DIV/0! | | | | | | | | | | |
| 14, 25 | #DIV/0! | | | | | | | | | | |
| 16, 24 | #DIV/0! | | | | | | | | | | |
| 16, 25 | #DIV/0! | | | | | | | | | | |
| 24, 25 | #DIV/0! | | | | | | | | | | |
| DK | #DIV/0! | | | | | | | | | | |
| Alpha | #DIV/0! | | | | | | | | | | |

Abbildung 17: Konsistenz der Kategorien für die Indoor-Gruppe

C.2. Outdoor

| Item | MW | Varianz | Std. Abw. | Anzahl | Links | Rechts | Skala |
|------|------|---------|-----------|--------|-------------------|-------------------------|-------------------|
| 1 | 2,0 | 1,0 | 1,0 | 3 | unerfreulich | erfreulich | Attraktivität |
| 2 | 0,7 | 2,3 | 1,5 | 3 | unverständlich | verständlich | Durchschaubarkeit |
| 3 | 2,3 | 0,3 | 0,6 | 3 | kreativ | phantasielos | Originalität |
| 4 | 2,0 | 1,0 | 1,0 | 3 | leicht zu lernen | schwer zu lernen | Durchschaubarkeit |
| 5 | 1,7 | 0,3 | 0,6 | 3 | wertvoll | minderwertig | Stimulation |
| 6 | 1,3 | 0,3 | 0,6 | 3 | langweilig | spannend | Stimulation |
| 7 | 1,7 | 0,3 | 0,6 | 3 | uninteressant | interessant | Stimulation |
| 8 | -1,7 | 0,3 | 0,6 | 3 | unberechenbar | voraussagbar | Steuerbarkeit |
| 9 | 2,0 | 1,0 | 1,0 | 3 | schnell | langsam | Effizienz |
| 10 | 2,3 | 0,3 | 0,6 | 3 | originell | konventionell | Originalität |
| 11 | 1,0 | 1,0 | 1,0 | 3 | behindernd | unterstützend | Steuerbarkeit |
| 12 | 2,0 | 1,0 | 1,0 | 3 | gut | schlecht | Attraktivität |
| 13 | 0,3 | 4,3 | 2,1 | 3 | kompliziert | einfach | Durchschaubarkeit |
| 14 | 0,0 | 4,0 | 2,0 | 3 | abstoßend | anziehend | Attraktivität |
| 15 | 2,3 | 0,3 | 0,6 | 3 | herkömmlich | neuartig | Originalität |
| 16 | 1,3 | 0,3 | 0,6 | 3 | unangenehm | angenehm | Attraktivität |
| 17 | 0,7 | 1,3 | 1,2 | 3 | sicher | unsicher | Steuerbarkeit |
| 18 | 1,7 | 0,3 | 0,6 | 3 | aktivierend | einschläfernd | Stimulation |
| 19 | 0,0 | 1,0 | 1,0 | 3 | erwartungskonform | nicht erwartungskonform | Steuerbarkeit |
| 20 | 1,0 | 1,0 | 1,0 | 3 | ineffizient | effizient | Effizienz |
| 21 | 0,3 | 1,3 | 1,2 | 3 | übersichtlich | verwirrend | Durchschaubarkeit |
| 22 | 1,0 | 1,0 | 1,0 | 3 | unpragmatisch | pragmatisch | Effizienz |
| 23 | 1,3 | 2,3 | 1,5 | 3 | aufgeräumt | überladen | Effizienz |
| 24 | 1,0 | 0,0 | 0,0 | 3 | attraktiv | unattraktiv | Attraktivität |
| 25 | 1,0 | 0,0 | 0,0 | 3 | sympathisch | unsympathisch | Attraktivität |
| 26 | 2,0 | 1,0 | 1,0 | 3 | konservativ | innovativ | Originalität |

| Skalen | |
|-------------------|-------|
| Attraktivität | 1,222 |
| Durchschaubarkeit | 0,833 |
| Effizienz | 1,333 |
| Steuerbarkeit | 0,000 |
| Stimulation | 1,583 |
| Originalität | 2,250 |

Bewertung pro Item

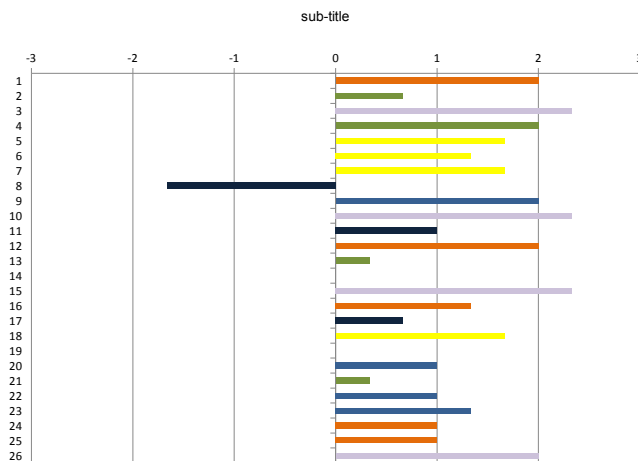


Abbildung 18: Auswertung zu den einzelnen Items und Skala der Kategorien der Outdoor-Gruppe

| Attraktivität | | Durchschaubarkeit | | Effizienz | | Steuerbarkeit | | Stimulation | | Originalität | |
|---------------|-------------|-------------------|-------------|-----------|-------------|---------------|-------------|-------------|-------------|--------------|-------------|
| Items | Korrelation | Items | Korrelation | Items | Korrelation | Items | Korrelation | Items | Korrelation | Items | Korrelation |
| 1, 12 | 1,00 | 2, 4 | 0,98 | 9, 20 | -1,00 | 8, 11 | -0,87 | 5, 6 | 0,50 | 3, 10 | 1,00 |
| 1, 14 | -1,00 | 2, 13 | 0,84 | 9, 22 | -1,00 | 8, 17 | -0,50 | 5, 7 | 1,00 | 3, 15 | 1,00 |
| 1, 16 | 0,87 | 2, 21 | 0,94 | 9, 23 | -0,33 | 8, 19 | -0,87 | 5, 18 | 1,00 | 3, 26 | 0,87 |
| 1, 24 | #DIV/0! | 4, 13 | 0,72 | 20, 22 | 1,00 | 11, 17 | 0,87 | 6, 7 | 0,50 | 10, 15 | 1,00 |
| 1, 25 | #DIV/0! | 4, 21 | 0,87 | 20, 23 | 0,33 | 11, 19 | 1,00 | 6, 18 | 0,50 | 10, 26 | 0,87 |
| 12, 14 | -1,00 | 13, 21 | 0,97 | 22, 23 | 0,33 | 17, 19 | 0,87 | 7, 18 | 1,00 | 15, 26 | 0,87 |
| 12, 16 | 0,87 | DK | 0,89 | DK | -0,11 | DK | 0,08 | DK | 0,75 | DK | 0,93 |
| 12, 24 | #DIV/0! | Alpha | 0,97 | Alpha | -0,68 | Alpha | 0,27 | Alpha | 0,92 | Alpha | 0,98 |
| 12, 25 | #DIV/0! | | | | | | | | | | |
| 14, 16 | -0,87 | | | | | | | | | | |
| 14, 24 | #DIV/0! | | | | | | | | | | |
| 14, 25 | #DIV/0! | | | | | | | | | | |
| 16, 24 | #DIV/0! | | | | | | | | | | |
| 16, 25 | #DIV/0! | | | | | | | | | | |
| 24, 25 | #DIV/0! | | | | | | | | | | |
| DK | #DIV/0! | | | | | | | | | | |
| Alpha | #DIV/0! | | | | | | | | | | |

Abbildung 19: Konsistenz der Kategorien für die Outdoor-Gruppe

C.3. Gesamt

| Item | MW | Varianz | Std. Abw. | Anzahl | Links | Rechts | Skala | |
|------|------|---------|-----------|--------|-------------------|-------------------------|-------------------|--|
| 1 | 2,0 | 1,0 | 1,0 | 3 | unerfreulich | erfreulich | Attraktivität | |
| 2 | 0,7 | 2,3 | 1,5 | 3 | unverständlich | verständlich | Durchschaubarkeit | |
| 3 | 2,3 | 0,3 | 0,6 | 3 | kreativ | phantasielos | Originalität | |
| 4 | 2,0 | 1,0 | 1,0 | 3 | leicht zu lernen | schwer zu lernen | Durchschaubarkeit | |
| 5 | 1,7 | 0,3 | 0,6 | 3 | wertvoll | minderwertig | Stimulation | |
| 6 | 1,3 | 0,3 | 0,6 | 3 | langweilig | spannend | Stimulation | |
| 7 | 1,7 | 0,3 | 0,6 | 3 | uninteressant | interessant | Stimulation | |
| 8 | -1,7 | 0,3 | 0,6 | 3 | unberechenbar | voraussagbar | Steuerbarkeit | |
| 9 | 2,0 | 1,0 | 1,0 | 3 | schnell | langsam | Effizienz | |
| 10 | 2,3 | 0,3 | 0,6 | 3 | originell | konventionell | Originalität | |
| 11 | 1,0 | 1,0 | 1,0 | 3 | behindernd | unterstützend | Steuerbarkeit | |
| 12 | 2,0 | 1,0 | 1,0 | 3 | gut | schlecht | Attraktivität | |
| 13 | 0,3 | 4,3 | 2,1 | 3 | kompliziert | einfach | Durchschaubarkeit | |
| 14 | 0,0 | 4,0 | 2,0 | 3 | abstoßend | anziehend | Attraktivität | |
| 15 | 2,3 | 0,3 | 0,6 | 3 | herkömmlich | neuartig | Originalität | |
| 16 | 1,3 | 0,3 | 0,6 | 3 | unangenehm | angenehm | Attraktivität | |
| 17 | 0,7 | 1,3 | 1,2 | 3 | sicher | unsicher | Steuerbarkeit | |
| 18 | 1,7 | 0,3 | 0,6 | 3 | aktivierend | einschläfernd | Stimulation | |
| 19 | 0,0 | 1,0 | 1,0 | 3 | erwartungskonform | nicht erwartungskonform | Steuerbarkeit | |
| 20 | 1,0 | 1,0 | 1,0 | 3 | ineffizient | effizient | Effizienz | |
| 21 | 0,3 | 1,3 | 1,2 | 3 | übersichtlich | verwirrend | Durchschaubarkeit | |
| 22 | 1,0 | 1,0 | 1,0 | 3 | unpragmatisch | pragmatisch | Effizienz | |
| 23 | 1,3 | 2,3 | 1,5 | 3 | aufgeräumt | überladen | Effizienz | |
| 24 | 1,0 | 0,0 | 0,0 | 3 | attraktiv | unattraktiv | Attraktivität | |
| 25 | 1,0 | 0,0 | 0,0 | 3 | sympathisch | unsympathisch | Attraktivität | |
| 26 | 2,0 | 1,0 | 1,0 | 3 | konservativ | innovativ | Originalität | |

| Skalen | |
|-------------------|-------|
| Attraktivität | 1,222 |
| Durchschaubarkeit | 0,833 |
| Effizienz | 1,333 |
| Steuerbarkeit | 0,000 |
| Stimulation | 1,583 |
| Originalität | 2,250 |

Bewertung pro Item

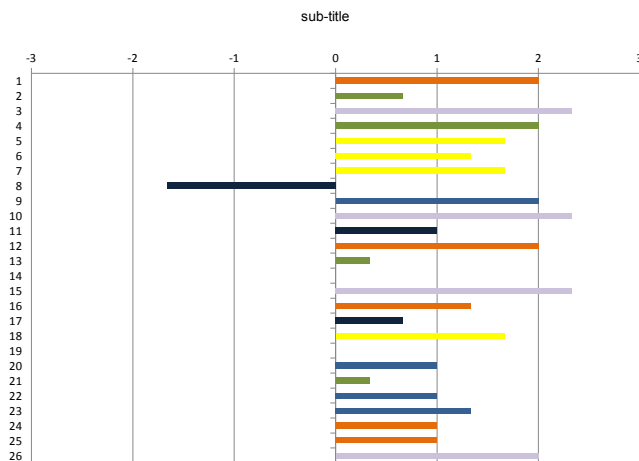


Abbildung 20: Auswertung zu den einzelnen Items und Skala der Kategorien der gesamte Gruppe

| Attraktivität | | Durchschaubarkeit | | Effizienz | | Steuerbarkeit | | Stimulation | | Originalität | |
|---------------|-------------|-------------------|-------------|-----------|-------------|---------------|-------------|-------------|-------------|--------------|-------------|
| Items | Korrelation | Items | Korrelation | Items | Korrelation | Items | Korrelation | Items | Korrelation | Items | Korrelation |
| 1, 12 | 1,00 | 2, 4 | 0,98 | 9, 20 | -1,00 | 8, 11 | -0,87 | 5, 6 | 0,50 | 3, 10 | 1,00 |
| 1, 14 | -1,00 | 2, 13 | 0,84 | 9, 22 | -1,00 | 8, 17 | -0,50 | 5, 7 | 1,00 | 3, 15 | 1,00 |
| 1, 16 | 0,87 | 2, 21 | 0,94 | 9, 23 | -0,33 | 8, 19 | -0,87 | 5, 18 | 1,00 | 3, 26 | 0,87 |
| 1, 24 | #DIV/0! | 4, 13 | 0,72 | 20, 22 | 1,00 | 11, 17 | 0,87 | 6, 7 | 0,50 | 10, 15 | 1,00 |
| 1, 25 | #DIV/0! | 4, 21 | 0,87 | 20, 23 | 0,33 | 11, 19 | 1,00 | 6, 18 | 0,50 | 10, 26 | 0,87 |
| 12, 14 | -1,00 | 13, 21 | 0,97 | 22, 23 | 0,33 | 17, 19 | 0,87 | 7, 18 | 1,00 | 15, 26 | 0,87 |
| 12, 16 | 0,87 | DK | 0,89 | DK | -0,11 | DK | 0,08 | DK | 0,75 | DK | 0,93 |
| 12, 24 | #DIV/0! | Alpha | 0,97 | Alpha | -0,68 | Alpha | 0,27 | Alpha | 0,92 | Alpha | 0,98 |
| 12, 25 | #DIV/0! | | | | | | | | | | |
| 14, 16 | -0,87 | | | | | | | | | | |
| 14, 24 | #DIV/0! | | | | | | | | | | |
| 14, 25 | #DIV/0! | | | | | | | | | | |
| 16, 24 | #DIV/0! | | | | | | | | | | |
| 16, 25 | #DIV/0! | | | | | | | | | | |
| 24, 25 | #DIV/0! | | | | | | | | | | |
| DK | #DIV/0! | | | | | | | | | | |
| Alpha | #DIV/0! | | | | | | | | | | |

Abbildung 21: Konsistenz der Kategorien für die gesamte Gruppe