



UNIVERSITÄT
KOBLENZ · LANDAU

Fachbereich 4: Informatik

Interaktive Schnitzeljagd mit Hilfe von mobilen Endgeräten

Bachelorarbeit

zur Erlangung des Grades Bachelor of Science (B.Sc.)
im Studiengang Computervisualistik

vorgelegt von
Cynthia Engel

Erstgutachter: Prof. Dr.-Ing. Stefan Müller
(Institut für Computervisualistik, AG Computergraphik)

Zweitgutachter: Anna Katherina Hebborn, M.Sc.
(Institut für Computervisualistik, AG Computergraphik)

Koblenz, im August 2014

Erklärung

Ich versichere, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Ja Nein

Mit der Einstellung der Arbeit in die Bibliothek bin ich einverstanden.

Der Veröffentlichung dieser Arbeit im Internet stimme ich zu.

.....
(Ort, Datum)

.....
(Unterschrift)

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Zielsetzung	1
2	Verwendete technische Komponenten	3
2.1	Vom Handy zum Smartphone	3
2.2	Android	3
2.2.1	Entwicklungswerkzeuge	4
2.2.2	Architektur	4
2.2.3	Komponenten einer Android Applikation	5
2.2.4	Der Aktivitätszyklus	6
3	Grundidee	9
3.1	Inspirationsquellen	9
3.1.1	Schnitzeljagd	9
3.1.2	Geocaching	9
3.2	DIN EN ISO	10
3.3	Persönliche Zielsetzung	10
4	Entwicklungsphase	11
4.1	Anforderungen an das Programm	11
4.1.1	Komponenten im Spieleinterface	12
4.2	Evaluation in der Entwicklungsphase	13
4.3	Tatsächliche grafische Umsetzung	14
4.4	Funktionsweise des Prototypen	18
4.4.1	Spielablauf	19
5	Umsetzung des Spielkonzepts in der Software	21
5.1	Implementierung	21
5.2	Zustandsdarstellung in Automaten- und Tabellenform	23
6	Evaluation nach der Programmierungsphase	25
6.1	Durchführung	25
6.2	Auswertung	26
6.2.1	Erster Teil	26
6.2.2	Zweiter Teil	26
6.2.3	Dritter Teil	35
6.3	Ausblick	38
7	Fazit	39

Zusammenfassung

Ziel der Bachelorarbeit war es, eine moderne Art des Geländespiels „Schnitzeljagd“ zu entwickeln. Dieses sollte möglichst auf jedem aktuellen Smartphone spielbar sein, welches mit dem Betriebssystem Android arbeitet. Das Gelände ist begrenzt auf den Universitätscampus Koblenz, somit dient das Spiel auch dazu, den Campus besser kennenzulernen.

Den Benutzern der Campusjagd wird eine mobile Applikation geboten, welche sie anhand von Hinweisen und Rätseln über den kompletten Campus führt, um letztendlich an einen Zielort zu gelangen, an welchem sich ein „Schatz“ befindet. Anstatt wie üblich auf dem Gelände Schnitzel bzw. Schnipsel mit Hinweisen zu verstecken, werden bei der Campusjagd QR-Codes aufgehängt, um so den Weg zu markieren. Zur Täuschung sind auch irrelevante Codes zu finden. Die Codes müssen in der richtigen Reihenfolge abgelaufen werden, d. h. von einem Code erhält der Spieler den Hinweis zum jeweiligen Nächsten. Außerdem ist es möglich, dass aus einem QR-Code mehrere Hinweise für nächste Stationen angezeigt werden.

Abstract

The aim of this thesis was to develop a modern type of a scouting game. This should, if possible, be playable on any current smartphone having an android operating system (OS). The area to search and hunt in is restricted to the campus of the University of Koblenz to achieve a better knowledge of the area around the campus.

The users of the “Campusjagd” are offered a mobile application which leads them through the entire campus via clues and riddles to finally arrive at the last destination where a “treasure” is hidden. Instead of hiding hints or snippets, as in the original game, QR-Codes are placed over the whole area to lead the path. There are also a few fake codes that got placed to lay a false trail. The codes must be passed in the correct order. This means that each code the player receives contains the hint to his consecutively code. It is also possible for several clues for the next stations to be displayed on one QR-Code.

1 Einleitung

Smartphones sind aus dem modernen Leben nicht mehr wegzudenken. Fast jeder besitzt mindestens eines davon. Man kann nur noch schwer auf diese moderne Er-rungenschaft verzichten. Es ermöglicht viele Dinge des täglichen Lebens einfacher und effektiver zu bewältigen, zum Beispiel: Einkaufen, Rezepte suchen, Straßen finden, Spielen und sogar Telefonieren. Vieles geht via sogenannter Applikationen (*kurz: Apps*), welche überwiegend im Internet, aber auch in den Stores der Handy- und Tablet-Hersteller zu finden sind. Desweiteren sind QR-Codes zu einer Modeerscheinung geworden, man findet ihn oft in der Werbung, beispielsweise in Prospekten oder auf Plakaten und Zeitungsanzeigen. Manchmal findet man solche Codes auch einfach auf der Straße, ohne Zusammenhang oder einen Hinweis, wozu er gehört.

Es ist außerdem allgemein bekannt, dass Geländespiele in Verbindung mit Smartphones immer beliebter werden und sehr erfolgreich sind. Dennoch sind solche Spiele meist auf große Flächen ausgedehnt und legen keinen Wert auf die detaillierte Erkundung eines bestimmten Bereiches. Diese Marktlücke soll mit diesem Produkt geschlossen werden.

Diese beiden Komponenten sollen genutzt werden, um ein kleines Spiel für ein Smartphone zu entwickeln und so einen Schritt in die Welt der Spieleentwicklung zu wagen.

1.1 Motivation

Überall auf der Straße sieht man Menschen mit ihrem Smartphone, gleich welchen Alters. Sie sind für viele ein ständiger Begleiter geworden.

Genauso ist es auf dem Koblenzer Campus: Selten sieht man Studenten, ohne Smartphone in der Hand. Das Campusgelände und die Umgebung erhalten nur noch wenig Aufmerksamkeit. Man sollte davon ausgehen, dass jeder Student den Campus kennt, da sogar zu Semesterbeginn eine Campus-Rallye veranstaltet wird, bei der die Erstsemester in kleine Teams eingeteilt werden, die gegeneinander antreten. Allerdings wird dieses Angebot nur vereinzelt von den neuen Studenten angenommen. Die Bereitschaft zur Erkundung des Campus im Rahmen eines Geländespiels könnte dadurch gesteigert werden, dass hierbei moderne Elemente, wie zum Beispiel Smartphones und QR-Codes, eingebunden werden.

1.2 Zielsetzung

Das Ziel der Arbeit ist die Entwicklung einer App, die dem Besucher den Koblenzer Campus näher bringen, jedes Gebäude und auch weitere schöne Orte zeigen soll. Ein besonderes Augenmerk wurde auf die Erstsemester geworfen, aber auch jeder andere Besucher soll diese App nutzen können. Realisiert wird dieser Ansatz in einer prototypischen Spiele-App. Die Nutzer sollen, ausgestattet mit einem Smartphone, den Campus erkunden. Daher wurden dort QR-Codes verteilt, wel-

che sie anhand von Hinweisen suchen und gegebenenfalls auch kleinere Fragen beantworten bzw. Aufgaben lösen müssen. Die Spielkonzepte von Schnitzeljagd und Geocaching sollen hierfür als Inspirationsquelle dienen.

Die Aufgaben wurden so konzipiert, dass der Nutzer wirklich an dem Ort oder in dessen näheren Umgebung sein muss, um diese lösen zu können, damit ausgeschlossen werden kann, dass alles nur aus seiner Fantasie entstanden ist. Sie sollen dem Nutzer den Campus näher bringen und kleine Details zeigen, die man beim ersten Überqueren nicht direkt wahrnimmt.

Das Spiel kann alleine oder auch in einer Gruppe gespielt werden. Es ist hierbei auch irrelevant, ob jeder ein Smartphone hat oder ob sich mehrere Nutzer eines teilen.

2 Verwendete technische Komponenten

Da ich persönlich ein relativ aktuelles Android-Smartphone besitze, liegt es nahe, eine Applikation für das Betriebssystem Android zu entwickeln. So lassen sich die Ergebnisse direkt im Echtbetrieb testen und nicht nur auf einer virtuellen Maschine.

2.1 Vom Handy zum Smartphone

Vor etwas mehr als 30 Jahren wurde von Motorola das erste Mobiltelefon entwickelt, das *DynaTAX 8000X*, welches mit einer 20 cm langen Antenne ausgestattet und noch dazu ein Kilogramm schwer war. Motorola entwickelte in den folgenden 20 Jahren stetig weitere Mobiltelefone, bis schließlich 1992 das erste GSM-fähige Modell präsentiert werden konnte. Nur ein Jahr später brachte IBM das erste Mobiltelefon mit einem Touchscreen auf den Markt. In den 90ern gab es noch viele weitere relevante Entwicklungen im Bereich der Handys: Es konnten SMS verschickt werden, die ersten Organizer wurden eingebaut, die Displays konnten Farben darstellen, um nur wenige zu nennen. Schließlich wurden die Handys immer preiswerter und solider, sodass sie den Massenmarkt erschlossen. Auch die Hardware entwickelte sich immer weiter. Mittlerweile werden weder Antennen noch Tasten an Mobiltelefonen benötigt. Sie wurden zu sogenannten Smartphones, die wie ein kleiner Computer benutzt werden können. Unter anderem wurden die Displays größer und hochauflösender, die Prozessoren schneller und viele neue Funktionen wurden integriert (Kameras, MP3-Player, GPS,...). [1] [2]

Neben der Hardware hat sich auch die Software für die Smartphones weiterentwickelt. Apple brachte 2007 das erste Smartphone mit eigenem Betriebssystem (das *iPhone*) auf den Markt. Ein Jahr später präsentierte Google sein Betriebssystem Android, welches auf dem HTC Dream vorgestellt wurde. [3] Es existieren noch einige andere Betriebssysteme, beispielsweise hat auch Microsoft eine Windowsversion eigens für Smartphones auf den Markt gebracht. [4] Laut dem Marktforscher Kantar hat Android in Deutschland einen Marktanteil von 77 Prozent und führt somit den Markt an, danach kommen iOS (mit 15,3 Prozent) und Windows Phone (mit 6,6 Prozent). [5][6] Auch unter diesem Gesichtspunkt erschien es sinnvoll, die Campusjagd für das Androidbetriebssystem zu entwickeln.

2.2 Android

Das Unternehmen Android wurde Ende 2003 von Rich Miner, Nick Sears, Andrew Rubin und Chris White in den USA gegründet. [7] Zwei Jahre später wurde es von der Firma Google übernommen. [8] Anfang November 2007 wurde von Google eine Probeversion des *Android Software Development Kit* (kurz: SDK) veröffentlicht. Allerdings stand den Entwicklern das mobile Endgerät erst ein Jahr später zur Verfügung. Zu dieser Zeit war das SDK schon weiter entwickelt und benutzerfreundlicher geworden. Es wurde intensiv an dieser Plattform gearbeitet, Fehler wurden schnell behoben, neue Funktionen eingefügt und mitgelieferte Anwendun-

gen stetig verbessert, sodass das Androidbetriebssystem mittlerweile mit dem des Konkurrenten Apple vergleichbar ist. [16]

Android bietet eine Plattform, in der viele Freiheiten gelassen werden und vieles nach Belieben verändert werden kann (beispielsweise Applikationen, Hintergründe oder Widgets können hinzugefügt werden). Diese Funktion wird einerseits von den Endanwendern genutzt, um ihr Gerät zu personalisieren, andererseits können die Hersteller von Smartphones gewisse Applikationen bereits vorab auf dem Handy installieren, um ihm damit ein markeneigenes Profil zu verleihen. Android ist auf verschiedensten mobilen Endgeräten zu finden (beispielsweise Smartphones oder Tablets). Es existieren von zahlreichen Herstellern viele unterschiedliche Modelle für den vielfältigen Gebrauch. [16]

2.2.1 Entwicklungswerkzeuge

Die Grundlage für die Entwicklung [16] einer javabasierten Android Applikation bildet das SDK. Dieses ist für Windows, Linux und Mac OS X verfügbar. Es besteht aus einem Emulator, einigen Tools sowie Dokumentationen und Beispielen. Eine einfache Java-Laufzeitumgebung (JRE) reicht nicht aus, da dort verschiedene Komponenten (beispielsweise Compiler) fehlen.

Google bietet für die Entwicklungsumgebung Eclipse ein Plug-In, das sogenannte *Android Development Tool* (kurz: ADT). Das Schreiben, Debuggen und Veröffentlichen wird dadurch wesentlich vereinfacht.

2.2.2 Architektur

Die Android Architektur [14] lässt sich schematisch in fünf Bereiche gliedern, welche auf Abbildung 1 dargestellt werden.

- **Android Runtime:** Android liefert die virtuelle Maschine Dalvik mit, welche nahezu jedes Programm, das auf einem Android-System von einem Benutzer ausgewählt wird, startet. Selbst geschriebene Programme betrifft dies auch. Dalvik ist auf Sicherheit und Stabilität ausgelegt, aus diesem Grund läuft für jede Applikation ein eigener Prozess mit jeweils einer eigenen Instanz auf ihr. Außerdem wurde die virtuelle Maschine auf minimalen Speicherverbrauch optimiert.

Die Core Libraries enthalten Java Pakete (beispielsweise *java.math*).

- **Linux Kernel:** Die virtuelle Maschine benötigt ein Betriebssystem. Android nutzt hier ein Open-Source-Betriebssystem, dies war früher Linux 2.6. [15], ab Android 4.x wird nun die 3.x-Serie verwendet. [10] Der Linux Kernel dient als Fundament für andere Schichten (siehe Abbildung 1), er fungiert sozusagen als Abstraktionsschicht und kapselt die übrigen Teile der Plattform voneinander ab.

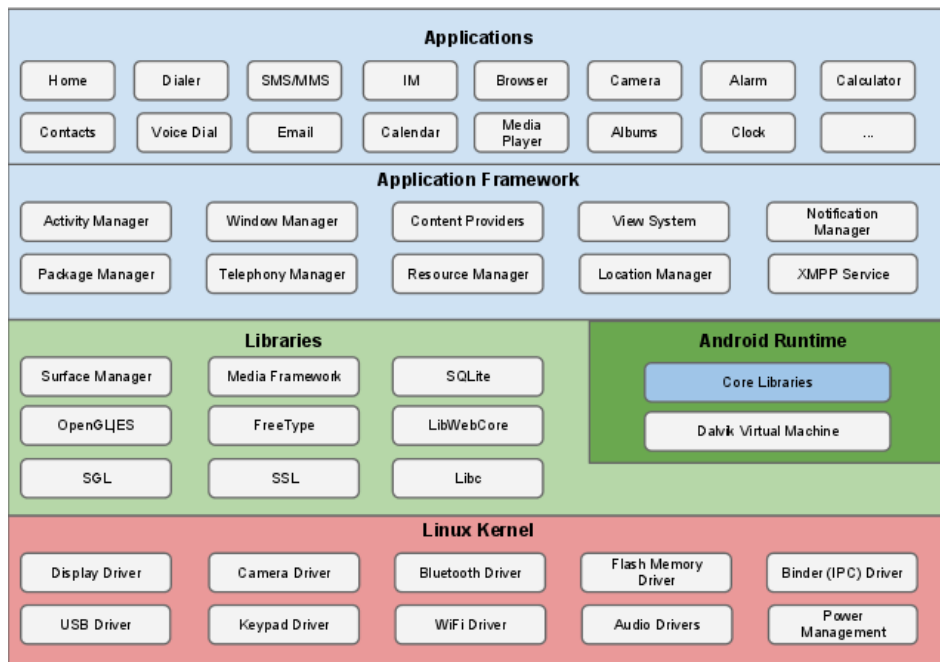


Abbildung 1: Die Android Architekturschichten [9]

- **Libraries:** Es existieren einige C/C++-Bibliotheken (beispielsweise für Datenbanken oder Vektorzeichensätze) für Android, als Entwickler kann direkt über das Anwendungsframework auf diese zugegriffen werden.
- **Application Framework:** Mittels dieses Frameworks ist es möglich Applikationen zu schreiben, welche einen großen Funktionsumfang besitzen und gleichzeitig leicht bedienbar sowie ästhetisch sind. Es dient dazu, von Applikationen Funktionen abzurufen, damit diese für andere Programme nutzbar werden.

Die wichtigsten Bestandteile sind die *Views* (sie können beliebig angepasst werden, da sie die Benutzeroberfläche darstellen), der *Resource Manager* (durch ihn wird der Zugriff auf Grafiken oder ähnliches gewährleistet), der *Notification Manager* (Pop-Up oder Statuszeilenbenachtigungen können durch ihn erzeugt werden), der *Content Provider* (siehe 2.2.3) und der *Activity Manager* (welcher den Lebenszyklus einer Anwendung steuert).

2.2.3 Komponenten einer Android Applikation

Android besteht im Gegensatz zu iOS oder JME aus vielen, lose gekoppelten Einzelteilen (Activities, Services, Content Providern, Intents, Broadcast-Receiver, usw.). [14]

- **Activities** (*deutsch*: Aktivitäten): Eine Applikation kann beliebig viele Acti-

ties besitzen, zwischen ihnen kann hin- und hergewechselt werden. Die Activities beherbergen die Präsentationsschicht, allerdings wird das eigentliche User-Interface (*kurz*: UI) von den Android Komponenten selbst realisiert.

- **Services:** Diese haben kein UI und arbeiten unsichtbar im Hintergrund der Applikation. Sie erwachen je nach AppLogik regelmäßig oder unregelmäßig zum Leben, um beispielsweise den Benutzer über neue E-Mails oder ähnliches zu informieren. Diese Informationen werden durch das Notification-Framework bereitgestellt.
- **Intents** (*deutsch*: Absichten): Dies sind die Informationen (oder auch Nachrichten genannt), welche an Services, direkt an eine Aktivität oder systemweit, wie eine Rundfunk-Nachricht (Broadcast), verteilt werden. Sie stellen die Kommunikation zwischen den einzelnen Komponenten dar.
- **Broadcast-Receiver:** Implizite Nachrichten (Intents) werden von ihnen systemweit konsumiert. So kann Android Mitteilungen verschicken, wenn beispielsweise der Ladezustand des Akkus dem Ende zuneigt. Broadcast-Receiver haben keine UI, können aber in der Statuszeile Benachrichtigungen hinterlassen.
- **Content-Provider:** Mit einem oder mehreren Content-Providern können Daten von anderen Apps genutzt werden. Dazu gehören auch interne Provider wie beispielsweise die Kontaktdaten aus einem Telefonbuch.

2.2.4 Der Aktivitätszyklus

Ein Programm besteht aus einer oder mehreren Aktivitäten [16], diese stellen den sichtbaren Teil des Programms dar. Aktivitäten [14] können sich gegenseitig aufrufen. Oftmals ist die Oberfläche der Applikation äquivalent zu jener der Aktivitätsklasse, welche sie abbildet. Sie repräsentieren die Interaktion mit dem Benutzer.

Gestartet wird eine Aktivität durch die Callback-Methode *onCreate()*, diese kann vom System oder direkt vom Nutzer aufgerufen werden. Sobald dies erfolgt, durchläuft die Aktivität den Aktivitätszyklus, welcher durch mehrere Callback-Methoden gesteuert wird. Das Betriebssystem entscheidet, unter Berücksichtigung des Speicherplatzes und der Priorität der Aktivität, ob eine Aktivität beispielsweise sichtbar ist oder ob ihr Prozess gelöscht werden kann.

Prioritäten einer Aktivität [14] Nachfolgend werden die Prioritäten entsprechend ihrer Rangfolge aufgeführt:

1. **Priorität** (= höchste): Die Aktivität ist im Vordergrund und kann vom Anwender direkt genutzt werden.
2. **Priorität:** Diese Priorität haben Prozesse, welche sich nicht unmittelbar im Vordergrund befinden, aber (wenn auch nur zum Teil) sichtbar sind. Dies kann beispielsweise sein, wenn eine Aktivität eine andere teils verdeckt.

- 3. Priorität:** Dies sind nicht sichtbare Prozesse, diese können entfernt werden, ohne dass der Benutzer dies merkt oder Schaden daran nimmt.
- 4. Priorität (= niedrigste):** Prozesse mit der niedrigsten Priorität werden am ehesten zerstört, da sie keine Aktivität, Service oder Ähnliches ausführen.

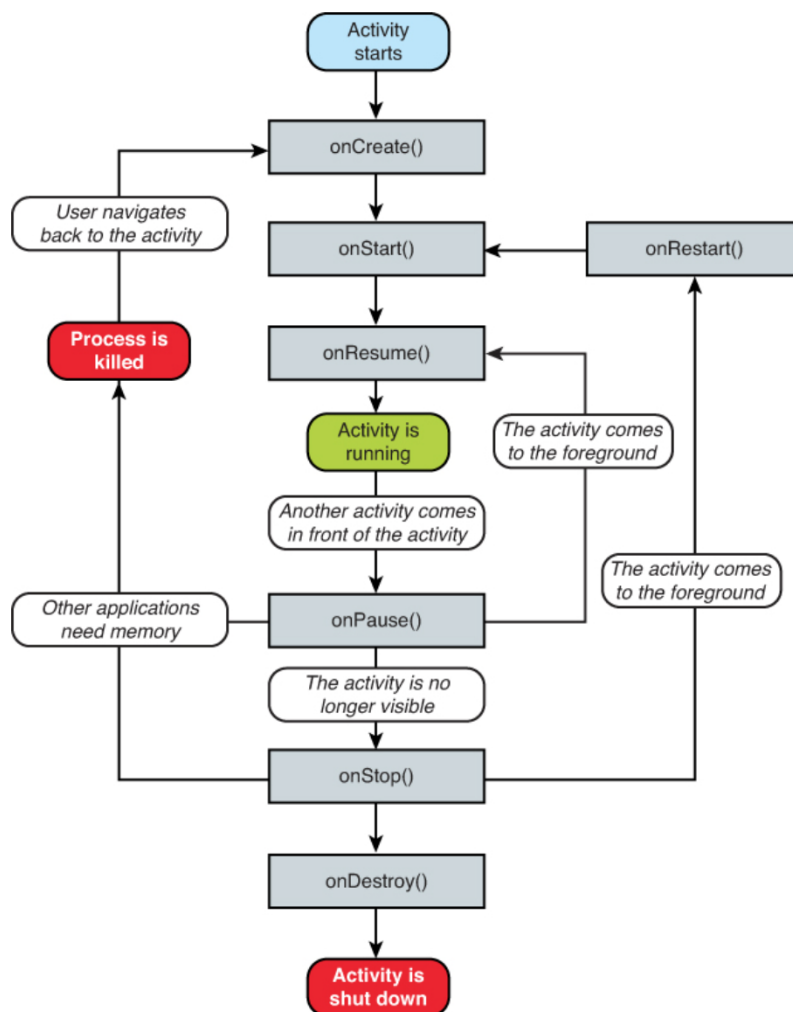


Abbildung 2: Der Android Aktivitätslebenszyklus [11]

Der Lebenszyklus [14] einer Aktivität lässt sich in drei Phasen unterteilen, wie auf Abbildung 2 zu erkennen ist.

1. Die erste Phase findet zwischen `onResume()` und `onPause()` statt. Dort ist das UI sichtbar und die Aktivität vollständig im Vordergrund. Dort kann der Anwender mit dem UI interagieren.
2. Die zweite Phase befindet sich zwischen `onStart()` bzw. `onRestart()` und `onStop()`. Dort ist die Aktivität noch sichtbar, der Nutzer muss allerdings nicht

mit ihr interagieren, da sie beispielsweise von einer anderen Aktivität verdeckt sein kann und sich somit nicht zwangsläufig im Vordergrund befindet.

3. Die dritte Phase befindet sich zwischen *onCreate()* und *onDestroy()*, dies bedeutet, es sind alle Methoden in dieser Phase mit einbezogen. Die Aktivität wird mittels *onCreate()* initialisiert und durch *onDestroy()* komplett zerstört.

Die Aktivität durchläuft folgende Methoden (dabei muss sie nicht gezwungenermaßen alle benutzen)[14]:

- ***onCreate(Bundle savedInstanceState)***: Durch diese Methode wird eine Aktivität gestartet und initialisiert. Außerdem wird das UI geladen und angezeigt. Das Bundle-Objekt wird nur benötigt, wenn ein vorheriger Zustand vorhanden ist.
- ***onStart()***: Diese Methode lässt die sichtbare Phase der Aktivität beginnen.
- ***onRestart()***: Eine Aktivität wird durch diese Methode erneut gestartet, dies kann passieren, falls sie vorher durch *onStop()* gestoppt wurde.
- ***onResume()***: Diese Methode startet, sobald eine Aktivität aufgerufen wird, welche Interaktionen vom Anwender im UI entgegennimmt. Somit ist die Aktivität vollständig im Vordergrund und kann vom Benutzer bearbeitet werden.
- ***onPause()***: Wenn ein Benutzer zu einer Aktivität wechselt, wird diese Methode aufgerufen. Es kann passieren, dass diese Aktivität gelöscht wird, um Speicherplatz zu gewinnen, da dies die letzte Station des Lebenszyklus ist. Dann würden *onStop()* und *onDestroy()* nicht mehr aufgerufen werden. Deshalb sollten Änderungen gespeichert werden, da mit Abschluss von *onPause()* die Aktivität nicht mehr vollständig im Vordergrund ist.
- ***onStop()***: Diese Methode läutet das Ende der sichtbaren Phase ein, da ab sofort eine andere Aktivität für den Benutzer sichtbar ist.
- ***onDestroy()***: Bevor eine Aktivität letztendlich zerstört wird, wird diese Methode zuletzt aufgerufen, dies muss nicht zwingend passieren (siehe *onPause()* Beispiel).

3 Grundidee

3.1 Inspirationsquellen

Im folgenden Kapitel soll ein Konzept für den zu Beginn geschilderten Ansatz aufgestellt werden. Dazu werden zunächst verwandte Spielkonzepte und Arbeiten vorgestellt, in denen der Ansatz teilweise schon umgesetzt wurde.

3.1.1 Schnitzeljagd

Die Schnitzeljagd ist ein altes Geländespiel für Kinder. Es wird ein Areal ausgesucht, das möglichst unübersichtlich ist (zum Beispiel ein Waldstück). Dort werden zwei Gruppen gebildet, einmal die Suchenden und diejenigen, welche gefunden werden sollen. Letztere laufen vor und hinterlassen an gewissen Stellen kleine Schnitzel bzw. Schnipsel als Hinweise für die Sucher, welche erst einige Zeit nach den Versteckern mit der Suche beginnen. Sie müssen möglichst schnell die Hinweise finden und versuchen die Verstecker einzuholen. Wenn sie dies Erreichen, ist das Spiel vorbei.

Als Hinweis können die zu Versteckenden alles verwenden, was ihnen zur Verfügung steht: Von kleinen beschriebenen Zetteln über Pfeile auf Wänden, Bäumen oder sonstigen beschreibbaren Möglichkeiten, bis in Formen zurecht gelegte Äste oder Steine. Es ist außerdem möglich, dass Hinweise gegeben werden, die nicht zum eigentlichen Ziel führen, um die Sucher in eine Sackgasse zu führen.

3.1.2 Geocaching

Geocaching ist ein weltweites Geländespiel für Menschen jeder Altersstufe. Es ist eine moderne Version der Schnitzeljagd bzw. der Schatzsuche. Benötigt wird lediglich ein Gerät, welches GPS Daten ausgeben kann (sei es ein GPS-Gerät oder ein modernes Smartphone) und die Koordinaten des sogenannten Caches (dem „Schatz“). Diese werden von Menschen versteckt, welche deren Koordinaten im Internet für Andere preis geben, damit diese sie suchen können. In jedem Cache befindet sich eine Liste, in welche sich der Sucher mit seinem Onlinenamen eintragen kann, um damit zu beweisen, dass der Cache tatsächlich von dieser Person gefunden wurde.

Caches sollen möglichst an schönen bzw. außergewöhnlichen Plätzen versteckt werden, damit die Menschen, die diese finden, dort auch etwas zu sehen haben. Dies kann besonders als Tourist in einer fremden Stadt nützlich sein, um diese kennen zu lernen. Aber auch als Einheimischer können so Plätze entdeckt werden, an denen früher nur vorbei gelaufen wurde, ohne diesen Aufmerksamkeit zu schenken oder von deren Existenz nichts gewusst wurde.

Es gibt verschiedene Arten von Caches. Normale Caches, welche auch am häufigsten vorkommen, beinhalten direkt die Koordinaten des Schatzes. Andere Caches wiederum locken die Suchenden erst einmal an einen Ort, an welchem ein weiterer Hinweis zum finalen Schatzstandort gefunden wird. Dies kann beliebig

oft ausgeführt werden (es handelt sich hierbei um einen sogenannten Multi-Cache). Andere Cachekoordinaten bekommt der Suchende erst, wenn er ein Rätsel gelöst hat (sogenannte Mystery-Caches).

3.2 DIN EN ISO

Die Richtlinien der Mensch-System-Interaktion werden in der Norm DIN EN ISO 9241 [12] näher erläutert. Diese Norm beschreibt die Anforderungen an die Arbeitsumgebung und die Hardware. Sie umfasst auch sämtliche Gestaltungsempfehlungen von den grundsätzlichen Dialogführung (Part 110), über die Informationsdarstellung (Part 12) und Benutzerführungen (Part 13) bis hin zu den Dialogtechniken (Part 14-17).

Besonders relevant für die vorliegende Arbeit ist Part 12, die Informationspräsentation, denn deren Richtlinien sollen soweit wie möglich in diesem Prototypen eingehalten werden.

Diese Richtlinien sind:

- **Verständlichkeit:** Die Informationen sind in ihrer Bedeutung eindeutig, erkennbar, interpretierbar und leicht verständlich.
- **Unterscheidbarkeit:** Die einzelnen Elemente der Informationen, welche angezeigt werden, können gut voneinander unterschieden werden.
- **Kürze:** Es werden nur relevante Informationen angezeigt, damit der Benutzer nicht mit zu vielen Informationen überlastet wird.
- **Konsistenz:** Die Darstellung der Informationen ist durchgängig einheitlich.
- **Entdeckbarkeit:** Auf die Informationen, welche der Anwender benötigt, wird dessen Aufmerksamkeit gelenkt.
- **Lesbarkeit:** Die Informationen sollen gut und leicht lesbar sein.
- **Klarheit:** Der Inhalt einer Information wird schnell und genau vermittelt.

3.3 Persönliche Zielsetzung

Ergänzend zu der in 1.2 genannten Zielsetzung soll von beiden erwähnten Inspirationsquellen etwas in die Entwicklung der Campusjagd einfließen. Einerseits ist es die Grundidee der Schnitzeljagd, wobei das Ganze auf ein Team – das Sucher-Team – reduziert wird. Andererseits möchte ich mir die Idee eines Multi-Geocaches zu eigen machen, da auch dort die nächsten Hinweise direkt fest an ihren Stationen angebracht worden sind und man diese nur noch suchen muss. Außerdem soll es, angelehnt an Geocaching, eine kleine Schatztruhe (in Form einer alten Filmdose) geben, in die sich die Finder eintragen können, wenn sie dies möchten. Die kleinen Rätsel sind angelehnt an die Idee eines Mystery-Caches.

4 Entwicklungsphase

4.1 Anforderungen an das Programm

Folgendes Anforderungsprofil wurde bei der Konzipierung der Campusjagd zugrunde gelegt:

Die Campusjagd ist ein Geländespiel auf dem Universitätscampus Koblenz. Dort wird an möglichst allen Gebäuden, schönen Plätzen und an Orten, die nicht direkt ersichtlich sind, je eine Station des Spieles platziert. Eine Station bedeutet in diesem Fall, dass an einem Versteck ein QR-Code hängen wird, welchen der Nutzer für weitere Informationen und Hinweise abscannen muss. Diese weisen ihn auf die Position der nächsten Station hin und stellen ihm womöglich kleine Aufgaben, die zu lösen sind. Die Verstecke sollen nach Schwierigkeitsgrad ausgesucht werden, die ersten einfacher, die späteren schwieriger. Dies bedeutet, dass die ersten Stationen einfacher zu finden sein sollen, ergo offensichtlicher versteckt sind. Bei den ersten Stationen soll es nur Hinweise geben, bei den Darauffolgenden sollen auch Fragen zu beantworten sein. Diese Fragen müssen sich auf die Umgebung des QR-Codes beziehen, so dass sie nur gelöst werden können, wenn der Benutzer die Umgebung etwas genauer betrachtet. Es kann auch passieren, dass ein QR-Code zwei oder mehr Hinweise ausgibt. Genauso kann es vorkommen, dass ein Hinweis eine Sackgasse ergibt, um den Benutzer in die Irre zu führen.

Der Benutzer muss mit seinem Smartphone die Stationen ablaufen können, da er dort die Hinweise erhält. Außerdem soll die Applikation die Möglichkeit besitzen, dem Benutzer anhand einer Karte seinen genauen Standort mit Hilfe von GPS-Daten anzuzeigen. Das GPS soll der Benutzer ein- und ausschalten können, je nachdem ob er es benötigt.

Wenn der Benutzer den QR-Code abgescannt und gegebenenfalls die zugehörige Aufgabe gelöst hat, wird er eine Zahl erhalten, welche er sich merken muss. Wenn er alle Stationen aufgesucht hat und alle Zahlen gesammelt hat, wird der Benutzer diese Zahlen zu dem sogenannten Mastercode zusammensetzen können. Mit diesem Code wird er Informationen zum „Schatz“ erhalten. Wenn der Mastercode eingegeben wird, soll der Benutzer noch einmal ein Bild erhalten, an welchem die finale Station zu finden ist. Dort kann sich der Benutzer auf einer Liste in einer versteckten Filmdose eintragen („Schatz“).

Es soll möglich sein, die einzelnen Stationen erneut zu spielen, um vielleicht mit Freunden nochmal einzusteigen. Deswegen soll es eine Art Levelauswahl geben. Ebenso soll der Benutzer das Spiel komplett zurücksetzen können, damit er es von vorne beginnen kann. Außerdem soll dem Benutzer die Wahl gelassen werden, ob er über ein Hauptmenü navigiert oder ob er lieber die Actionbar des Programms benutzt. In der Actionbar sollen alle wichtigen Funktionen (primär das Scannen von Codes) wiederzufinden sein.

4.1.1 Komponenten im Spieleinterface

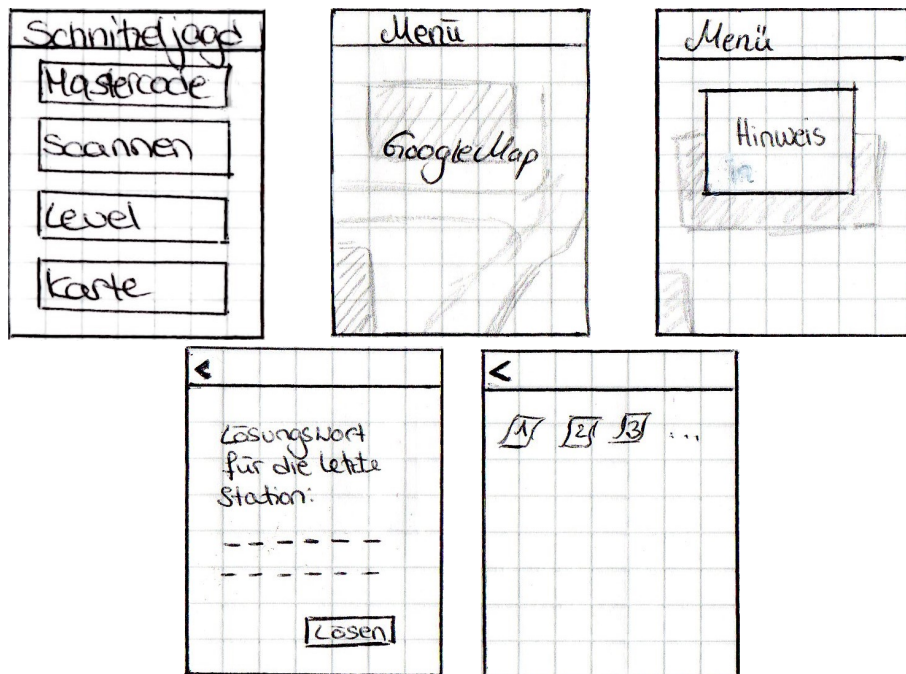


Abbildung 3: Die Komponenten in der Entwicklung

Wird das Programm gestartet, so wird zunächst das Hauptmenü (siehe Abbildung 3 links oben) aufgerufen. Dort hat der Benutzer, die Möglichkeit sich die Karte anzeigen zu lassen, einen QR-Code zu scannen, in die Levelauswahl einzutreten, das Spiel zurückzusetzen oder den Mastercode einzugeben. Unabhängig davon wird, kommt der Benutzer in allen Fenstern durch den Pfeil in der linken oberen Ecke zurück ins Hauptmenü. Dies funktioniert auch durch die Smartphone-eigene Taste.

Auf der Karte soll der Standpunkt des Anwenders per GPS angezeigt werden (siehe Abbildung 3 mitte oben) und von dort aus soll er mit Hilfe der Actionbar auch direkt einen Code absキャン können. Wurde ein Code abgescannt, wird dem Benutzer ein Hinweis angezeigt (siehe Abbildung 3 oben rechts). Die Mastercode-seite soll sehr schlicht gehalten werden, dort soll der Nutzer nur den Code auf dessen Richtigkeit überprüfen können (siehe Abbildung 3 links unten). Wird die Levelauswahl geöffnet (siehe Abbildung 3 rechts unten), kann zwischen den einzelnen Stationen hin und her gewechselt werden, allerdings kann ein Level erst ausgewählt werden, wenn es bestanden ist. Wenn das Spiel zurückgesetzt wird, erscheint eine Notifikation, dass dieser Vorgang alle Spieldaten löscht und ob der Nutzer dies auch wirklich will. Wenn Scannen im Hauptmenü ausgewählt wird, erscheint die Kamera und es kann mit dem Scannen eines QR-Codes begonnen werden.

4.2 Evaluation in der Entwicklungsphase

Nach Fertigstellung des Konzepts wurde eine erste Evaluation durchgeführt. Diese bestand aus einem persönlichen Gespräch mit den Probanden, um herauszufinden, ob das Konzept in der geplanten Form auf diese ansprechend wirkt. Besonders wichtig war hierbei, herauszufinden, ob die eigentliche Grundidee der Arbeit – die Entwicklung eines Geländespiels, das einen Anreiz für mehrere Spieldurchläufe bietet – durch das geplante Konzept angemessen umgesetzt wurde. Auch die Benutzerfreundlichkeit und der zu erwartende Spaßfaktor sollten untersucht werden.

Das persönliche Gespräch wurde als Evaluationsart gewählt, da dort nicht nur Fakten gesammelt werden können, sondern die Probanden auch eigene Ideen oder Verbesserungen eher anbringen können, als bei einem einfachen Fragebogen. In einem Gespräch können Aspekte berücksichtigt werden, die ein Fragebogen gar nicht abdecken würde, da bei der Erstellung des Fragebogens nur ein begrenztes Fragenspektrum erarbeitet werden kann. Das Gespräch gestaltete sich so, dass dem Probanden erst das komplette Konzept erklärt wurde, auch auf die Standartschnitzeljagd wurde, um sicher zu gehen, eingegangen. Danach kamen einige einfache Fragen (zum Beispiel, ob der Proband die Grundidee der Schnitzeljagd mag oder ähnliche Spiele kennt), in Folge dessen wurde direkt auf das Konzept und dessen Funktionalität in Hinblick auf die Aufgabenstellung eingegangen und mit einigen Fragen versucht ein offenes Gespräch zur Ideensammlung zu initiieren.

Im Großen und Ganzen waren sich alle Probanden einig. Die meisten Differenzen gab es eher bei kleineren Aspekten, welchen ihren Ursprung in den persönlichen Vorlieben der Probanden haben. Auch die Bedienung wurde zunächst als intuitiv angesehen, da je nach Gewohnheit entweder die Actionbar oder das Hauptmenü zur Navigation genutzt werden kann. Außerdem wurde bestätigt, dass es wichtig ist, von jedem Punkt aus immer zurück ins Hauptmenü zu kommen. Insgesamt gab es viel positives Feedback und die Probanden ohne Smartphone würden das Spiel gerne einmal testen, oder fanden zumindest die Idee gut.

Die geplante Verwendung der Actionbar wurde als innovativ angesehen, da alle der Meinung waren, dass von der Karte aus auch unbedingt gescannt werden muss, da dies das Hauptfeature neben der Karte selbst ist. Beim GPS gingen die Geschmäcker auseinander, einige fanden es positiv, andere wiederum waren der Meinung, dass der Nutzer mittlerweile beim Blick auf eine Karte erwartet, dass der momentane Aufenthaltsort direkt kenntlich gemacht wird. Das Hauptmenü wurde als sinnvoll angesehen, da es einen Wiedererkennungswert hat, es eine kleine Übersicht über das Spiel gibt und es damit etwas verständlicher und zusammenhängender gestaltet ist. Dies ist gerade für Menschen, die nicht so erfahren mit Smartphones sind, hilfreich, da diese mit der Actionbar vielleicht noch nicht so viel anfangen können.

Die grundlegendste Abweichung der endgültigen Campusjagd gegenüber der Entwicklungsphase ist, dass sich während der Evaluation herausstellte, dass ein solches Geländespiel nur zur einmaligen Verwendung geeignet ist. Die Stationen sind dem Nutzer nach erstmaligem Durchspielen bereits bekannt, sodass kein wei-

terer Anreiz in einer erneuten Nutzung besteht. Nach dem ersten Durchlauf verliert das Spiel seinen Reiz und auch die eigentliche Intention, die hinter dem Spiel steckt. Der Inhalt müsste ständig erweitert werden, um das Spiel interessant zu halten. Dazu gäbe es mehrere Möglichkeiten: Die erste wäre, das Spiel auf eine größere Fläche auszudehnen, das heißt von der eigentlichen Idee ausgehend, den Radius größer um die Universität zu ziehen. Die zweite wäre, direkt mehrere kleine Umgebungen zu wählen, zum Beispiel einige Stationen in der Altstadt oder an der Fachhochschule und dies beliebig auf Koblenz ausdehnen. Die Erweiterung, in Fall eins den Radius zu erweitern und in Fall zwei immer weitere Stationen zu ergänzen, müsste in einem regeltem Abstand passieren, damit das Spiel seinen Reiz nicht verliert.

Eine Variation der ursprünglichen Grundidee wäre es, das Spiel nicht an der Universität starten zu lassen, sondern an einem prägnanten Ort in Koblenz und eine Touristentour daraus zu machen, sodass Touristen einen kleinen Überblick über die schönsten beziehungsweise bekanntesten Plätze in Koblenz bekommen. Allerdings hat das Spiel dadurch ebenfalls keinen, beziehungsweise nur einen bedingten Wiederspielwert, da ein Tourist, der die Stadt einmal komplett gesehen hat, wohl in den seltensten Fällen noch einmal alle Punkte anschauen möchte.

Da, wie oben beschrieben, ein Wiederspielwert jedoch nur mit erheblichem und fortgesetztem Aufwand erzielt werden kann, soll nunmehr ein Geländespiel entwickelt werden, welches den Erstsemestern, ähnlich wie Touristen in einer fremden Stadt, den Universitätscampus näher bringt. Es eignet sich somit als Zusatz für die Campus-Rallye, welche zu Beginn eines Semesters stattfindet. Aber nicht nur Erstsemestern soll es Spaß machen, sondern auch denjenigen, welche den Campus schon kennen und vielleicht selbst die eine oder andere Ecke noch nicht so aufmerksam angesehen haben.

Wenn davon ausgegangen werden kann, dass die Campusjagd nun keinen Wiederspielwert hat und einmalig gespielt wird, werden weder Levelauswahl noch ein Zurücksetzen-Button gebraucht. Alternativ wäre es möglich, diesen Button einzubringen, sodass die Probanden das Spiel nochmals mit Freunden spielen können. Die Anwender könnten es allerdings auch an einem Handy spielen. Damit wäre dieser Button eher optional.

4.3 Tatsächliche grafische Umsetzung

Wenn das Programm gestartet wird, erscheint zunächst das Hauptmenü (siehe Abbildung 4 linkes Bild). Von dort aus wird dem Benutzer die Wahl gelassen, den Mastercode einzugeben, die Karte aufzurufen oder einen QR-Code zu scannen. Die Tatsache, dass er von allen Menüpunkten ausgehend wieder durch die linke obere Ecke zurück zum Hauptmenü kommt, bleibt erhalten.

Wenn Scannen ausgewählt wird, startet die Kamera und fordert den Nutzer auf, einen QR-Code einzuscannen. Falls ein falscher oder kein Code eingescannt wurde, weist das Programm den Benutzer darauf hin. Wenn nun ein richtiger Code eingescannt wurde, erscheint ein Pop-Up, auf dem anfangs nur das Bild für die

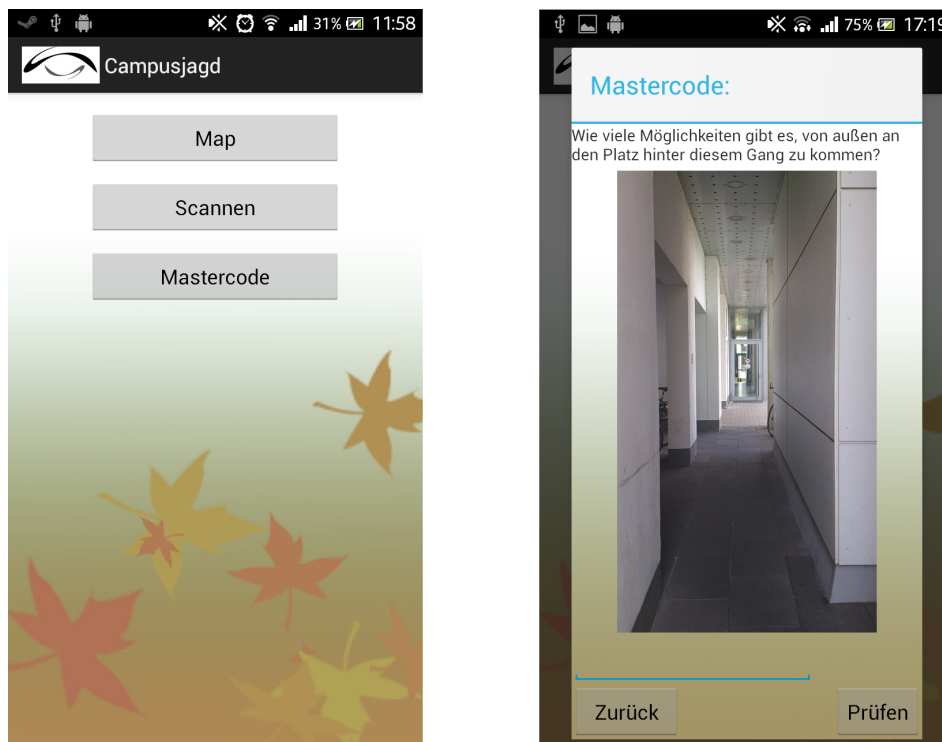


Abbildung 4: Hauptmenü und Scannvorgang im fertigen Programm

nächste Station erscheint, in der oberen Pop-Up-Leiste steht direkt die jeweilige Zahl für den Mastercode. Im späteren Spielverlauf stehen hier noch die Fragen, welche sich auf die zu erkundenden Spielstationen beziehen. Außerdem wird ein Eingabefeld angezeigt, in welches der Spieler seine Antwort eintragen kann (siehe Abbildung 4 rechtes Bild). Desweiteren befinden sich unter diesem Eingabefeld zwei Buttons. Mit dem einen kann der Benutzer seine Antwort prüfen lassen und mit dem anderen gelangt er zurück zum Hauptmenü. Wenn der Nutzer die Frage richtig beantwortet hat, erscheint in der oberen Leiste hinter „Mastercode“ die Zahl dieser Station für den Mastercode. Falls die Antwort falsch eingegeben wurde, wird der Nutzer darauf hingewiesen und muss erneut eine Antwort eingeben.

Beim Mastercode (siehe Abbildung 5 linkes Bild) existieren nur ein Textfeld und ein Button. Dort sollen die bei den Stationen gesammelten Zahlen in der richtigen Reihenfolge eingegeben werden. Falls die eingegebene Zahlenreihenfolge falsch ist, erscheint ein Pop-Up, welches den Benutzer darauf hinweist, dass die eingegebenen Zahlen falsch sind und er es erneut versuchen soll. Wenn sie jedoch richtig sind, erscheint ein anderes Pop-Up (siehe Abbildung 5 rechtes Bild), dieses zeigt dem Nutzer die letzte Station und erklärt, dass dort nun der endgültige „Schatz“ zu finden ist. Werden in das Feld andere Zeichen als Zahlen eingegeben, wie zum Beispiel Buchstaben oder Sonderzeichen, so weist das Programm ihn darauf hin, dass in das Feld nur Zahlen eingegeben werden dürfen.

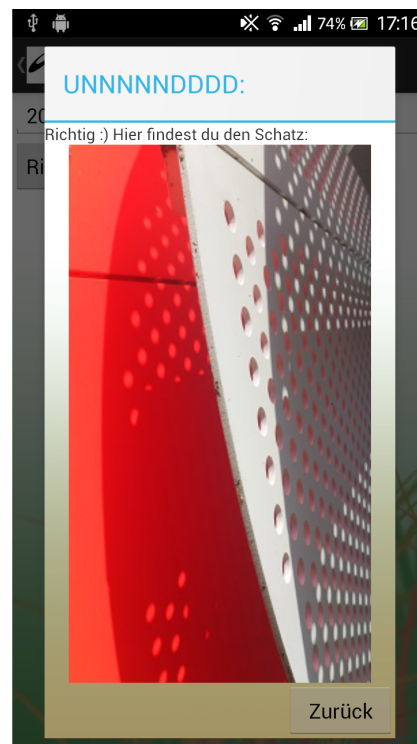
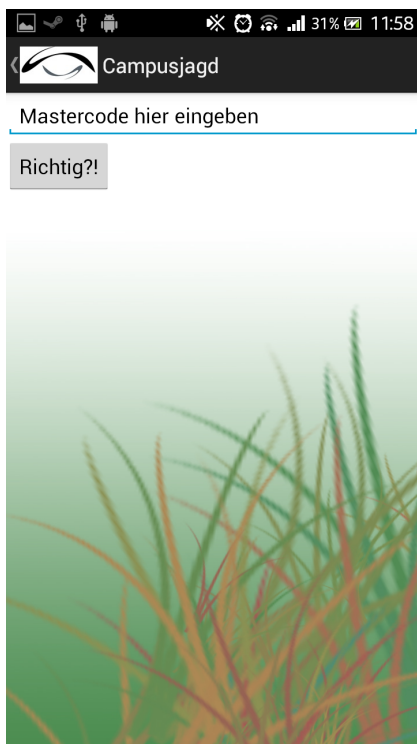


Abbildung 5: Die Mastercodeaktivität und deren Pop-Up

Wird die Karte (siehe Abbildung 6 linkes Bild) gewählt, startet sie auf einem herangezoomten Campus Koblenz. Somit ist der Benutzer direkt am Ort des Geschehens. Rechts unten ist, wie auch bei anderen Karten, die Möglichkeit zum Zoomen gegeben. Oben Rechts kann das GPS an- und ausgeschaltet werden. Dadurch wird die Position in Form eines blauen Punktes der spielenden Person direkt auf der Karte angezeigt. Dieser Punkt vollzieht die Bewegungen des Spielers nach. In der ActionBar befinden sich drei Icons: die Lupe führt zum Scannenbereich, durch das Fragezeichen erscheint die Eingabemöglichkeit für den Mastercode und das Bild schaltet die Karte zwischen Satelliten- und Kartenansicht um (siehe Abbildung 6 rechtes Bild; dort kann auch der blaue GPS-Punkt, mit Radius, erkannt werden).

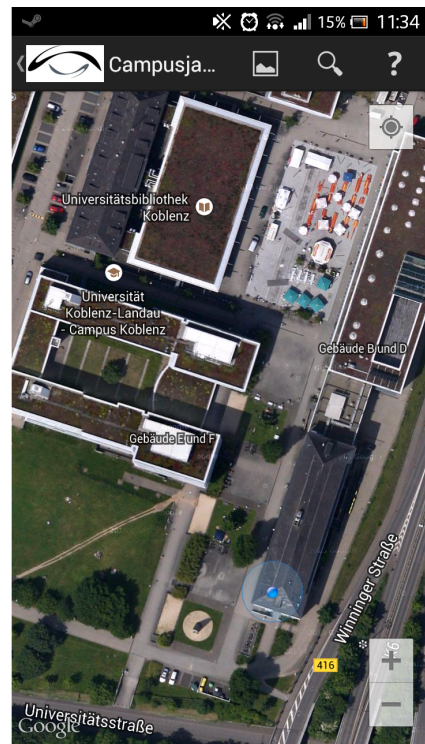


Abbildung 6: Die Karte und deren Funktionen im fertigen Programm

4.4 Funktionsweise des Prototypen

Auf jedem der aufgehängten Stationszettel befindet sich unter dem stationseigenen QR-Code ein einheitlicher QR-Code, welcher den Nutzer auf eine Webseite verweist ¹. Auf dieser wird das Spiel ausführlich erklärt und auch die Installationsdatei ist hier zu finden. Da es sein kann, dass Personen den Code aus Interesse anschauen bzw. abschnappen, ist es wichtig Ihnen vorher zu erklären, wovon das Spiel handelt und nicht einfach direkt den Download zu starten, um zu vermeiden, dass sie sich eine Software auf ihr Smartphone laden, die sie gar nicht nutzen wollen. Falls man einen der Hinweis-QR-Codes mit einem üblichen QR-Code-Scanner abschnappt, wird dieser nur Zahlen zwischen 1 und 99 ausgeben. Werden diese QR-Codes hingegen über den Scanner der Campusjagd-Applikation eingescannt, werden die Zahlen in Stationen umgewandelt und dem Benutzer Hinweise zur jeweiligen Station gegeben.

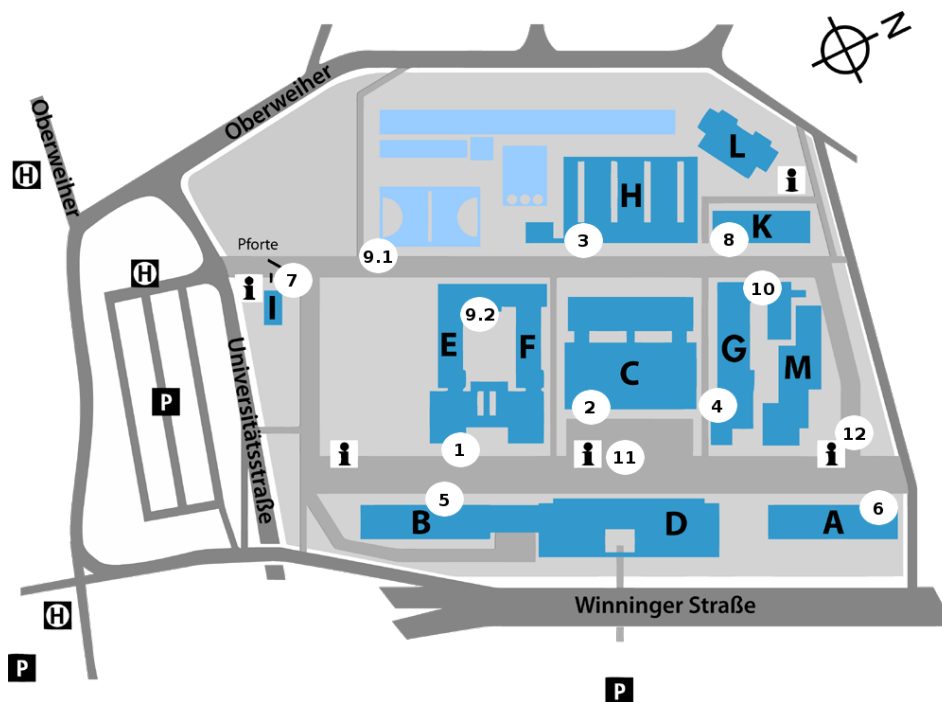


Abbildung 7: Die Verteilung der Stationen auf dem Campus

Es sind insgesamt zwölf QR-Codes auf dem Campus verteilt (siehe Abbildung 7). Wobei Station acht zwei Hinweise auf einmal gibt. Damit ist jedes Gebäude abgedeckt, in dem Veranstaltungen für Studenten stattfinden, sowie das Gewächshaus und das kleine Haus an der Schranke mit der Transponderausgabe. Außerdem gibt es drei Stationen, darunter auch die finale Station mit dem Schatz, welche Plätze an der Universität zeigen, die vielleicht nicht auf den ersten Blick ersichtlich sind.

¹<https://userpages.uni-koblenz.de/~cengel/campusjagd>

An der ersten Station erhält der Spieler den Hinweis auf die weitere Station. An allen darauffolgenden Stationen sind einfach zu lösende Fragen zu beantworten. Es kann auch sein, dass der Benutzer ein Gebäude betreten muss, um sich dort etwas genauer zu informieren, damit er die Frage beantworten kann, wenn er die Antwort nicht kennt. Es gibt eine Ausnahme, die Station 9.1, sie ist eine falsche Station, sie stellt dem Benutzer keine Frage. Die Hinweise sind immer Bilder beziehungsweise Bildausschnitte von einem Gebäude oder einem markanten Gebäudeteil, nach dem der Anwender suchen muss. In der Nähe dieses Hinweisausschnitts befindet sich der neue QR-Code. Die Antwort auf die Frage muss nicht zwingend genau an der selben Stelle des Gebäudes sein. Die QR-Codes selbst sind zu Anfang des Spiels eher offensichtlicher versteckt (siehe Abbildung 8 linkes Bild) gegen Ende hin werden sie schwieriger auffindbar (siehe Abbildung 8 rechtes Bild).



Abbildung 8: Beispiele von versteckten QR-Codes

4.4.1 Spielablauf

Exemplarisch für alle Stationen wird nun Station sieben einmal komplett durchgespielt (die anderen Stationen und deren Fragen befinden sich im Anhang).

Der Benutzer hat die Aufgabe der sechsten Station erfolgreich gelöst und die entsprechende Ziffer für den Mastercode bereits erhalten. Nun sucht er mithilfe des in Aufgabe sechs ausgegebenen Hinweises nach dem versteckten QR-Code der siebten Station (siehe Abbildung 9 linkes Bild). Nach dem Auffinden desselben

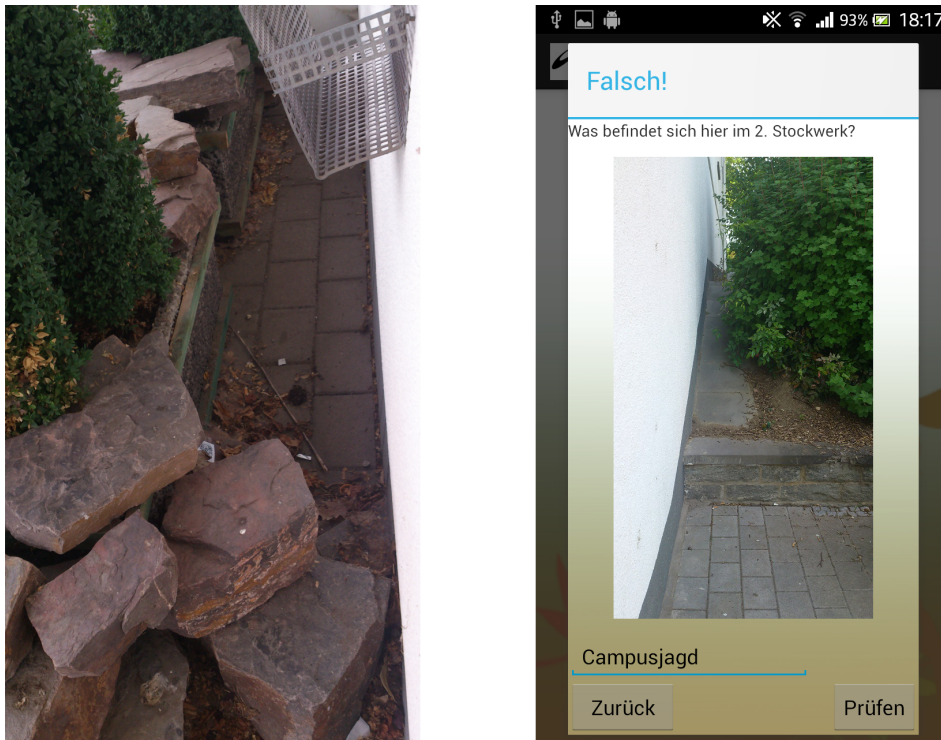


Abbildung 9: Versteck des QR-Codes und Aktivität von Station acht

scannt er ihn mit seinem Smartphone ab und erhält ein Pop-Up mit dem Hinweisbild, welches auf die nächste Station verweist (siehe Abbildung 9 rechtes Bild). Dort sieht man in der oberen Leiste, dass noch keine Mastercodezahl für diese Station vom Benutzer gefunden wurde (vgl. Abbildung 4 rechtes Bild). Er hat somit diese Frage noch nicht richtig beantwortet. Darunter befindet sich direkt die Frage für die achte Station. Nun sucht der Benutzer auf dem Campus nach der im Hinweisbild gezeigten Stelle, bis er diese in der Realität wiederfindet. Als Orientierungshilfe kann er sich auf der Karte anzeigen lassen, wo er sich befindet. Am Ziel angekommen muss der Spieler zunächst die Antwort auf die Frage finden. Sobald er diese gefunden hat, kann er sie in dem Editfeld eingeben und durch den Prüfen-Button auf Richtigkeit überprüfen lassen. War die Antwort richtig, erscheint in der oberen Leiste die Mastercodezahl. Wenn dies nicht der Fall war, erscheint ein „falsch“ (siehe Abbildung 9 rechtes Bild). Danach kann der Benutzer den nächsten QR-Code suchen, dessen Aufenthaltsort in dem Bildausschnitt gezeigt wird. Der Zyklus beginnt erneut, bis er alle Zahlen des Mastercodes gesammelt hat.

5 Umsetzung des Spielkonzepts in der Software

In diesem Kapitel wird kurz auf die Implementierung des Prototypen eingegangen. Die wichtigsten Klassen und deren Zusammenhänge werden hier beschrieben. Die jeweiligen Layout- und XML-Dateien der verschiedenen Klassen werden jedoch nicht im Einzelnen dargestellt.

Außerdem werden die Zustände des Prototypen mit Hilfe eines Zustandsautomaten und einer -tabelle verdeutlicht.

5.1 Implementierung

Wird das Spiel gestartet, so erscheint das Hauptmenü, welches mit Hilfe der Klasse *UIActivity* dargestellt wird. Damit ist diese Aktivität die *MainActivity* der Campusjagd. Sie besitzt primär drei Buttons, welche anhand von Intents auf die jeweiligen Klassen bzw. Aktivitäten der Buttons zugreifen. Anhand des Codes in Listing 1 ist dies beispielhaft für den Mastercode dargestellt. Dies passiert sobald ein Benutzer den jeweiligen Button drückt. Außerdem beendet die *UIActivity*- Klasse das Programm, wenn dies durch das Betätigen der Smartphonetaste vom Nutzer gewünscht wird.

```
1 // ruft die Mastercode Activity auf
2 public void insertMasterCode (View view) {
3     Intent intent =
4         new Intent (this , InsertMasterCode . class );
5     startActivity ( intent );
6 }
```

Listing 1: Aufruf der Mastercode-Aktivität vom Hauptmenü aus

Neben den Klassen für die jeweiligen Buttons (*ScanCode*, *InsertMasterCode* und *MapActivity*) gibt es einerseits noch eine Klasse, welche das Pop-Up des Mastercodes erstellt, und andererseits die Klasse *GameLogic*. Letztere steuert das komplette weitere Verfahren, welches abläuft, nachdem der Benutzer einen QR-Code abgescannt hat. Dieser wird zunächst in einen Integer umgewandelt, welcher daraufhin in die jeweilige Station umgesetzt wird. Jenes geschieht mit Hilfe einer switch-case-Anweisung (vgl. Listing 2). Ein QR-Code besitzt je eine Zahl zwischen 1 und 99, wenn keine dieser Zahlen eingegeben wird, erscheint der Defaulttext („Es wurde kein richtiger Code eingescannt.“). Außerdem gleicht die *GameLogic* die eingegebenen Antworten mit dem richtigen Ergebnis ab, um die richtige Mastercodezahl in der oberen Leiste einfügen zu können. Dies geschieht mittels der Methode *onClick*, welche durch das Klicken des Prüfen-Buttons aufgerufen wird.

```

1 // Station 3
2 case 13:
3     dialog.setTitle("Mastercode: ");
4     text.setText("Welche Fakultät ist in diesem Gebäude
5         vertreten? Nenne eine!");
6     view.setImageResource(R.drawable.a5);
7 break;

```

Listing 2: Auszug aus der Stationsbehandlung (Station 3)

Die Klasse *InsertMasterCode* erstellt mit ihrer *onCreate*-Methode die Mastercodeseite. Prüft ein Benutzer seinen eingegebenen Mastercode durch Buttonklick, wird die Methode *showDialog* aufgerufen. Damit seine Antwort mit der korrekten Lösung abgeglichen werden kann muss die Eingabe zunächst über ein paar Umwege in einen Integer umgewandelt werden. Diese Zahl wird an eine andere Klasse (*MasterDialog*) weitergegeben und dort weiterverarbeitet. Die Rückmeldung, ob der eingegebene Mastercode richtig oder falsch war, wird in einem Pop-Up ausgegeben. Dieses zeigt den entsprechenden Text und ein Bild an, entweder das der letzten Station oder das Default-Bild (vgl. Listing 3).

```

1 \\Überprüfung der eingegebenen Zahl
2 if (textInt == 2019577319) {
3     text.setText("Richtig! Hier findest du den Schatz:");
4     view.setImageResource(R.drawable.a15);
5 } else {
6     text.setText("Da ist irgendetwas falsch ,
7         versuche es nochmal!");
8 }

```

Listing 3: Auszug aus *MasterDialog*: Die Mastercode-Überprüfung

Die Klasse *ScanCode* ruft die handyeigene Kamera auf und scannt einen QR-Code ab. Die so erhaltene rohe Information leitet sie an die *GameLogic* weiter.

In der *onCreate*-Methode der Klasse *MapActivity* wird ein *GoogleMap*-Element erzeugt (vgl. Listing 4). Auf diesem können die GPS-Daten des Spielers dargestellt werden. Außerdem wird hier die Funktion zum Zoomen bereitgestellt. Die Initialisierung der Actionbar findet auch hier statt (vgl. Listing 4), genauso wie die ihrer Buttons zum Ansichtswechsel, Scannen und Eingeben des Mastercodes. Für diese Funktionen gibt es jeweils eigene Methoden, wobei jene für das Scannen und den Mastercode mittels Intents auf die dafür vorgesehenen Klassen verweisen, welche auch vom Hauptmenü erreichbar sind. Außerdem werden die Kamerakoordinaten der *GoogleMap* direkt auf die Universität Koblenz gesetzt, was mit Hilfe der Eingabe von Längen- und Breitengraden geschieht.

```

1 //ActionBar initialisieren
2 getActionBar().setDisplayHomeAsUpEnabled(true);
3
4 //GoogleMap erstellen
5 FragmentManager fmanager = getSupportFragmentManager();
6 Fragment fragment =
7     fmanager.findFragmentById(R.id.map);
8 SupportMapFragment supmapfragment =
9     (SupportMapFragment) fragment;
10 mMap = supmapfragment.getMap();

```

Listing 4: Ein Auszug aus der *onCreate*-Methode der *MapActivity*

5.2 Zustandsdarstellung in Automaten- und Tabellenform

Die Applikation kann verschiedene Zustände annehmen. Die möglichen Zustände und deren Übergänge werden in Abbildung 10 tabellarisch dargestellt. Die erste Spalte beschreibt die möglichen Zustände. Die oberste Zeile führt die Bedingungen auf, mit welchen ein Zustand in einen anderen überführt wird. Dies geschieht in der Applikation durch Benutzereingaben oder -klicks. In der Tabelle selbst befinden sich die Zustände, in welche das Programm wechselt, wenn ein Zustand mit einer gewissen Bedingung angewendet wird. Diese Tabelle wird mit dem Automaten (vgl. Abbildung 11) verdeutlicht. Z_0 stellt hierbei den Anfangsstatus dar, dieser wird aufgerufen, wenn ein Anwender das Programm startet. Bei Beendigung des Programms gelang es in den Endzustand Z_1 . Diese und alle anderen Zustände spiegeln den Programmablauf wider.

	Close	Back	Map	Scan	Code	Code: ✓	Code: ✗	Hint
Z_0	Z_1	×	Z_2	Z_4	Z_3	×	×	×
Z_1	×	×	×	×	×	×	×	×
Z_2	×	Z_0	×	Z_8	Z_7	×	×	×
Z_3	×	Z_0	×	×	×	Z_5	Z_3	×
Z_4	×	Z_0	×	×	×	×	×	Z_6
Z_5	×	Z_3	×	×	×	×	×	×
Z_6	×	Z_0	×	×	×	×	×	×
Z_7	×	Z_2	×	×	×	Z_9	Z_7	×
Z_8	×	Z_2	×	×	×	×	×	Z_6
Z_9	×	Z_7	×	×	×	×	×	×

Abbildung 10: Zustandstabelle

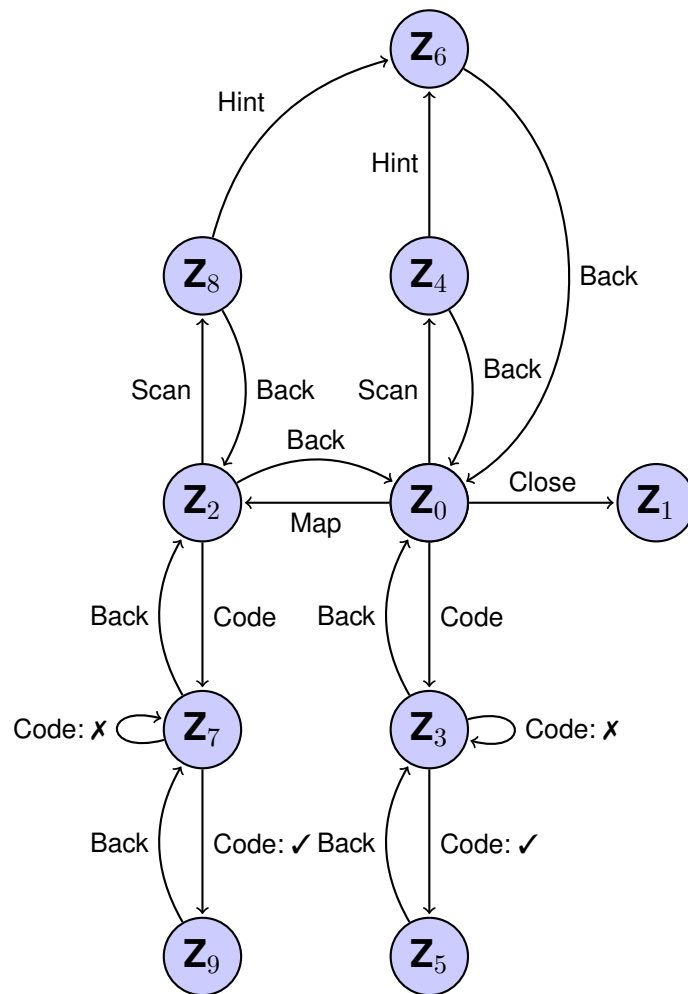


Abbildung 11: Zustandsautomat mit $Z_0 =$ Anfangszustand und $Z_1 =$ Endzustand)

6 Evaluation nach der Programmierungsphase

6.1 Durchführung

Die Evaluation wurde auf dem Campus Koblenz durchgeführt. Die Campusjagd begann am E-Gebäude, welches auch die erste Station des Spiels ist. Den Probanden wurde das Spielprinzip und die Funktionsweise der App erklärt. Die Teilnehmer bekamen entweder ein Smartphone geliehen, auf welchem die Campusjagd bereits installiert war, oder sie nutzen ihr eigenes Smartphone, auf welches die App mithilfe der oben genannten Webseite heruntergeladen wurde. Die Probanden wurden einzeln nacheinander aufgefordert mit dem Spiel zu beginnen. Nach erfolgreichem Abschluss des Spiels wurde ihnen ein Fragebogen ausgehändigt, den sie ausfüllen sollten.

Der Fragebogen (vgl. Anhang) lässt sich in drei Bereiche aufteilen:

- **Erster Teil:** Hier sind zwei persönliche Fragen zu finden. Es wird erfragt, ob der Proband ein Student der Universität Koblenz ist und, falls dies der Fall ist, in welchem Semester er hier studiert. Dies sollte erfragt werden, da davon auszugehen ist, dass Studenten, welche sich schon länger am Campus befinden, diesen auch besser kennen und somit den Schwierigkeitsgrad des Spiels gegebenenfalls nicht so hoch bewerten. Außerdem wird gefragt, ob der Proband ein eigenes Smartphone besitzt. Dies ist ein weiterer Aspekt, der bei der späteren Auswertung berücksichtigt werden muss, da ein Nicht-Smartphone-Nutzer eventuell schlechter mit der Bedienung der Campusjagd klar kommen könnte.
- **Zweiter Teil:** Dieser Teil widmet sich inhaltlichen Fragen zu dem Spiel. Es sollen Informationen gesammelt werden, wie gut der Benutzer mit der Bedienung der Campusjagd zurecht kam, ob sie ihm gefallen hat und welche Aspekte und Komponenten er genutzt hat. Dies wird mittels einer Likert-Skalar ermittelt, bei welcher die Probanden positiv oder negativ formulierten Aussagen zustimmen oder widersprechen können. Dieser Teil enthält sechzehn Aussagen. Die Skalierung ist ungerade, sie besteht aus fünf Abstufungen, welche von *Stimme nicht zu* bis *Stimme zu* verlaufen. Es wurde eine ungerade Anzahl gewählt, da möglichst vermieden werden soll, dass ein Proband, welcher einer Aussage gegenüber tatsächlich neutral eingestellt ist, sich gezwungen fühlt, unter diesem Druck eine Entscheidung (in eine bestimmte Richtung) zu treffen. Damit könnte das Ergebnis verfälscht oder sogar in die Richtung des gewünschten Ergebnisses gebracht werden.
- **Dritter Teil:** Hier geht es um das Endergebnis des Spiels und ob der Proband denkt, dass er neue Orte kennengelernt hat oder ob er das Spiel für seinen grundlegenden Zweck für hilfreich erachtet. In diesem Teil existieren nur Fragen, welche mit ja und nein beantwortet werden können. Allerdings wird

hier dem Probanden bei den meisten Fragen noch eine Möglichkeit gelassen, seine Antwort in Form eines Freitextes zu begründen oder ein Beispiel zu nennen.

6.2 Auswertung

Im Folgenden wird der Fragebogen ausgewertet.

6.2.1 Erster Teil

An der Evaluation nahmen neun Probanden teil, welche für die Campusjagd keine besonderen Voraussetzungen erfüllen mussten. Es waren sowohl männliche als auch weibliche Personen, ihr Alter lag jeweils zwischen 21 und 25 Jahren.

Von diesen neun Probanden studieren sechs am Campus Koblenz (siehe Abbildung 12). Ihre Semesteranzahl befindet sich zwischen zwei und acht. Den meisten Teilnehmern war somit der Universitätscampus schon vorher bekannt, sodass das Spiel im Mittel nicht als sehr schwer bewertet werden dürfte. Vergleiche hierzu auch die Auswertung der entsprechenden Fragen im zweiten Teil.

Ein Proband gab an, dass er kein Smartphone besitze, aber durchaus mit der Bedienung vertraut sei. Die anderen besitzen nach eigenen Angaben ein Smartphone (siehe Abbildung 13).

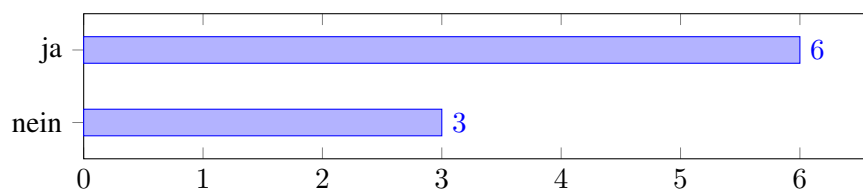


Abbildung 12: Studieren Sie in Koblenz?

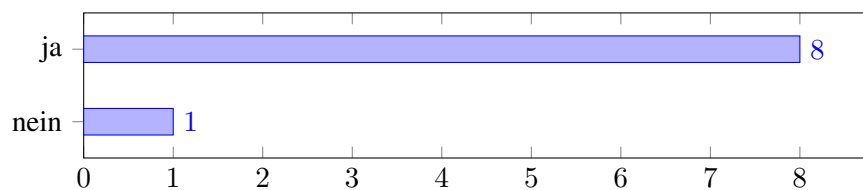


Abbildung 13: Haben Sie ein eigenes Smartphone?

6.2.2 Zweiter Teil

Bis auf einen Probanden, der neutral gestimmt war, hat allen anderen das Konzept des Spiels gut bis sehr gut gefallen (vgl. Abbildung 14). Dies ist eine gute Ausgangsposition für die komplette Evaluation, da keiner der Probanden das Spiel

ablehnt hat. Konstruktive Kritik ist unter diesen Voraussetzungen wesentlich wahrscheinlicher, als wenn das Konzept des Spiels grundsätzlich negativ gesehen wird.

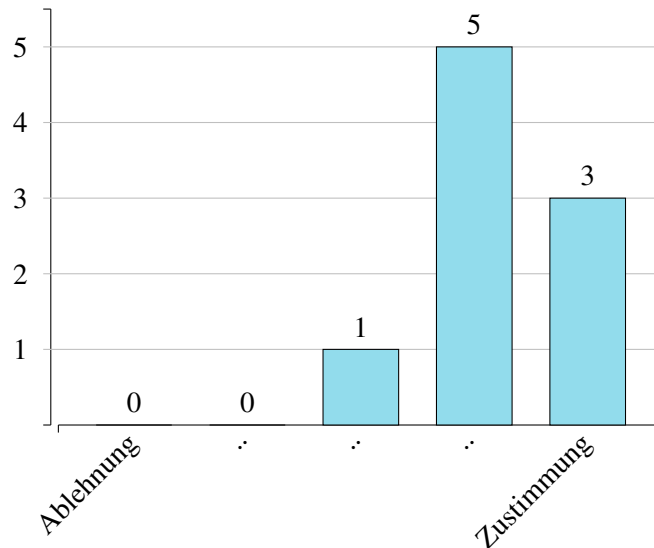


Abbildung 14: Das Konzept hat mir gefallen.

Die Umsetzung wurde von zwei Drittel der Probanden als gut erachtet, das restlichen Drittel war dazu neutral eingestellt (vgl. Abbildung 15), was mit der Beantwortung der vorangegangenen Frage korreliert, da bei mangelhafter Umsetzung wohl auch das zugrundeliegende Konzept negativer bewertet worden wäre.

Dementsprechend hat die Campusjagd sieben Probanden unterhalten (vgl. Abbildung 16). Die Ergebnisse in diesem Bereich sind somit konsistent.

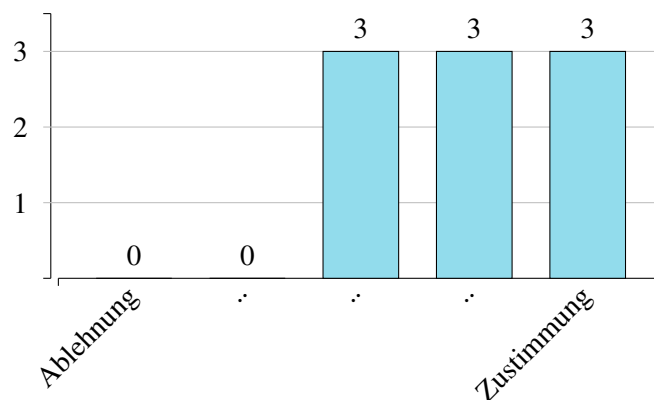


Abbildung 15: Die Umsetzung war gut.

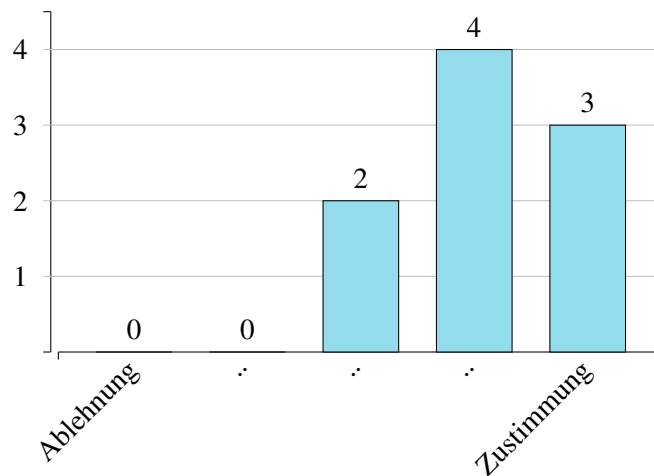


Abbildung 16: Das Spiel hat mich unterhalten.

Zum größten Teil (zwei Drittel) kamen die Probanden gut mit der Bedienung der App zurecht. Ein Proband war dieser Aussage gegenüber neutral eingestellt und zwei stimmten ihr nicht unbedingt zu (vgl. Abbildung 17). Ein ähnliches Ergebnis erzielte die Aussage, dass die Bedienung intuitiv gewesen sei. Allerdings stimmten hier nur vier Probanden zu, zwei enthielten sich und drei fanden, dass diese Aussage eher nicht zutrifft (vgl. Abbildung 18).

Hier müssten noch einmal genauere Nachforschungen betrieben werden, welche Aspekte der Benutzung die Probanden gestört oder gar irritiert haben. Die meisten schienen zwar mit der Bedienung klar zu kommen, dennoch wäre es nach den DIN EN ISO 9241-12 wünschenswert, dass sich möglichst alle Probanden schnell in die Software einfinden und diese als intuitiv und leicht nutzbar empfinden. Außerdem sollten alle Informationen klar vermittelt werden und die Nutzer nicht nach den gewünschten Informationen suchen müssen. Allerdings scheint die Bedienung, so wie sie in dem Prototypen der Campusjagd eingesetzt wurde, obwohl sie nicht bei allen Probanden als intuitiv und gut bedienbar eingestuft wurde, nicht störend für den Spielfluss gewesen zu sein, da der Großteil der Probanden Spaß am Spiel hatte (vgl. Abbildung 16).

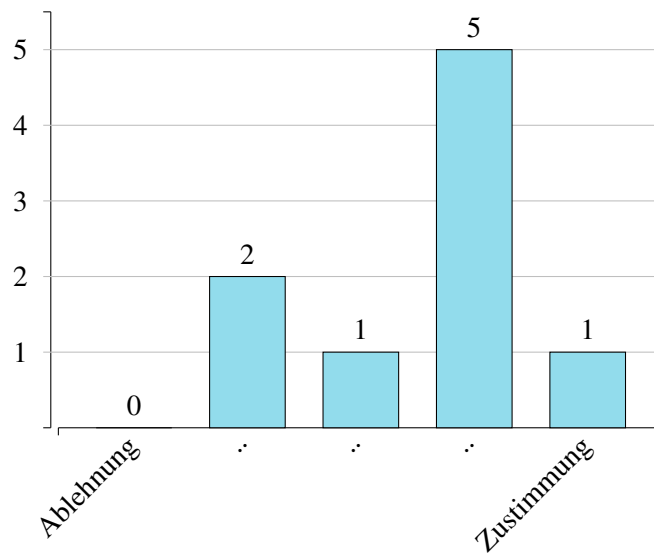


Abbildung 17: Ich kam gut mit der Bedienung klar.

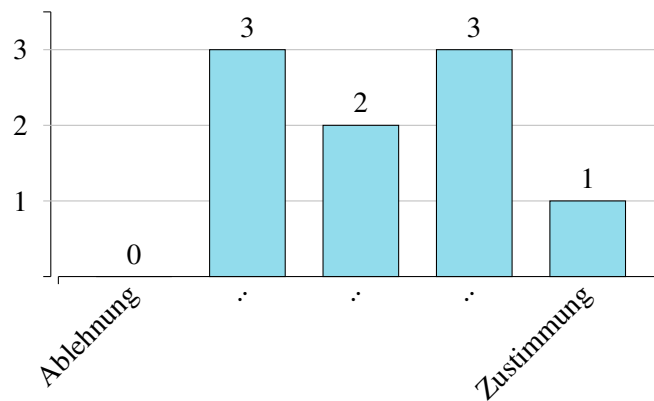


Abbildung 18: Die Bedienung war intuitiv.

Das GPS wurde von vier Probanden selten bis gar nicht genutzt, drei nutzten es gelegentlich und zwei etwas häufiger (vgl. Abbildung 19). Ein ähnliches Ergebnis erzielte die Aussage, die Actionbar sei öfters als das Hauptmenü benutzt worden (vgl. Abbildung 20). Hier verneinten vier Probanden die Aussage komplett, die anderen fünf verteilten sich auf die vier verbliebenen Antwortmöglichkeiten.

Das GPS sollte erhalten bleiben, da es auf der Karte keinen Platz für essentiell wichtige Informationen wegnimmt und einige Probanden es nutzten. Der Umstand, dass einige Teilnehmer diese Funktion nicht verwendet haben, könnte darin begründet sein, dass sie die Karte nicht aufgerufen haben, da diese nicht zwangsläufig notwendig ist, um das Spiel durchzuspielen. Dies würde genau passen, da vier Probanden weder GPS noch die Karte genutzt haben. Hierbei muss auch be-

rücksichtigt werden, dass der Campus den meisten Teilnehmern schon vorher bekannt war. Außerdem wäre es möglich, dass die Probanden sich das Hauptmenü nicht richtig angesehen haben und die Funktionen nicht genutzt haben, da sie ihnen unbekannt waren. Dies könnte damit zusammenhängen, dass einige Probanden das Fehlen einer Anleitung bemängelten (vgl. 6.2.3). Sie haben zwar eine Einführung in die Benutzung erhalten, aber diese später gerne nachgelesen. Mit dieser Anleitung könnte eventuell eine stärkere Annahme der Funktionen Karte und GPS erreicht werden.

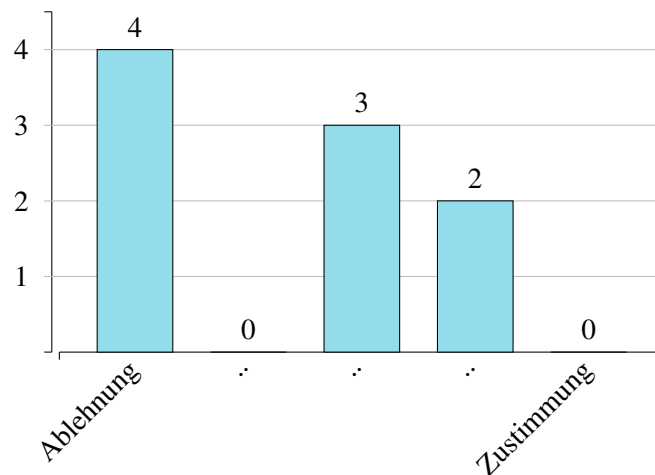


Abbildung 19: Ich habe das GPS oft genutzt.

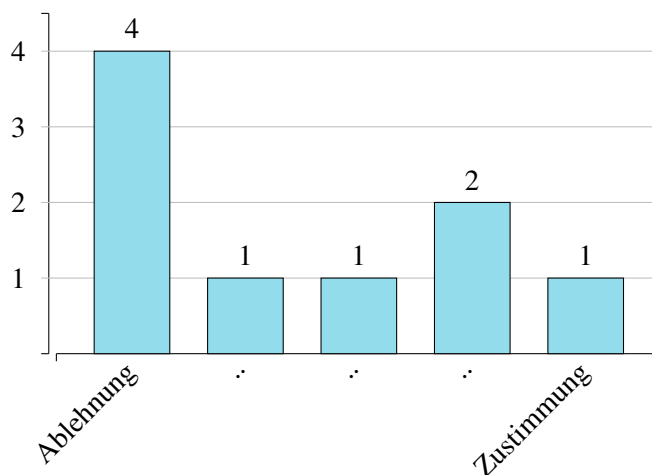


Abbildung 20: Ich habe die Actionbar öfters benutzt als das Menü.

Die Aufgaben haben keinen der Probanden beim Spielfluss gestört (vgl. Abbildung 22). Außerdem wurden sie von allen im Schwierigkeitsgrad als angemessen

betrachtet (vgl. Abbildung 21). Wie bereits eingangs erwähnt, ist das subjektive Schwierigkeitsempfinden der Probanden unter dem Gesichtspunkt zu betrachten, dass die meisten den Campus kennen. Es lässt sich somit aus den Ergebnissen folgern, dass die Aufgaben gut zu dem Spiel und dessen Konzept passen und vermutlich auch für Erstsemester geeignet und lösbar sind.

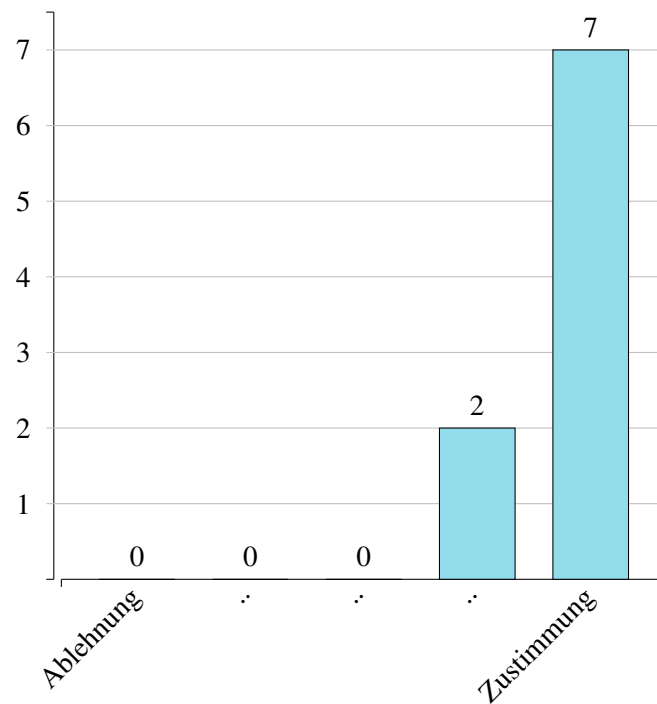


Abbildung 21: Die Aufgaben fand ich im Schwierigkeitsgrad angemessen.

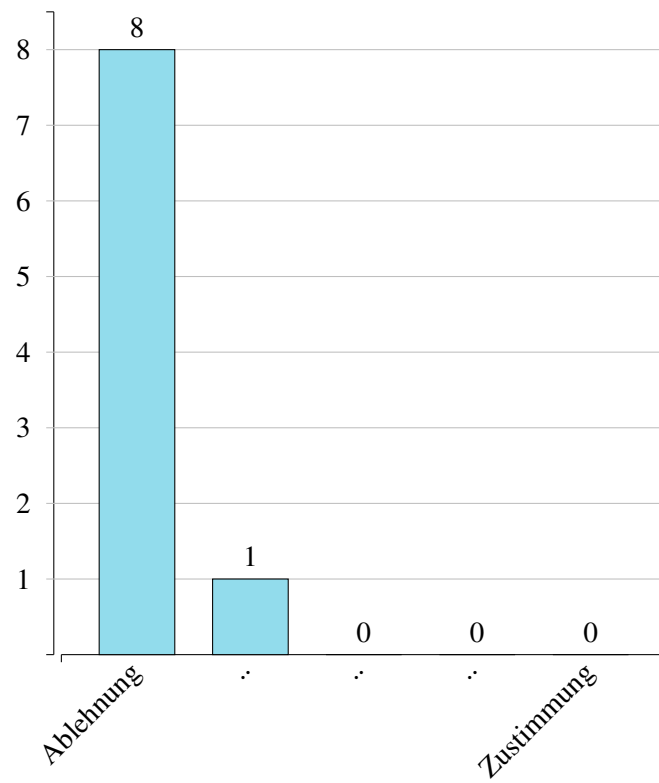


Abbildung 22: Die Aufgaben haben den Spielfluss gestört.

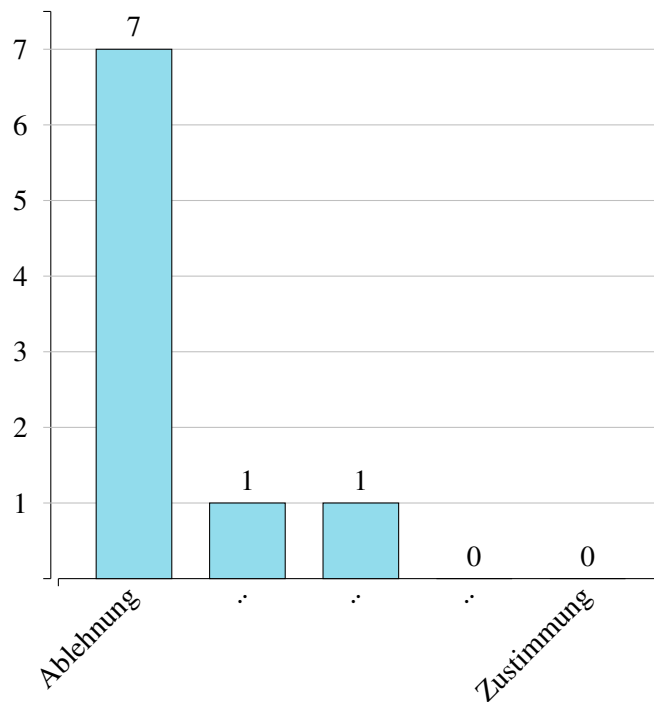


Abbildung 23: Die einzelnen Stationen waren viel zu weit auseinander.

Die ausgesuchten Stationen haben zwei Drittel der Probanden sehr gut gefallen. Das restliche Drittel empfand die Auswahl jedoch zugleich nicht als negativ (vgl. Abbildung 24). Außerdem wurde auch die Wahl der Stationsreihenfolge positiv beurteilt. Sieben der Probanden gaben an, dass die Stationen nicht zu weit auseinander liegen (vgl. Abbildung 23). Auch die anderen beiden Probanden stimmten dem zu. Dies bedeutet, keiner der Spieler hatte das Gefühl, unnötig viel über den Campus gelaufen zu sein.

Keiner der Probanden stimmte zu, dass das Spiel zu viele Stationen hat, also war die Entscheidung richtig, jedes Gebäude einmal zu besuchen, welche für die Studierenden wichtig sein könnten und um gleichzeitig einige schöne Orte zu finden.

Der Aufwand (vgl. Abbildung 25) wurde von keinem Proband als zu hoch erachtet. Drei Spieler lehnten diese Aussage ab, die anderen sechs stimmten ihr sogar überhaupt nicht zu.

Somit lässt sich festhalten: Die Stationen, deren Platzwahl und Entfernung untereinander, sowie der Aufwand des gesamten Spieles ist in sich stimmig und gut gewählt.

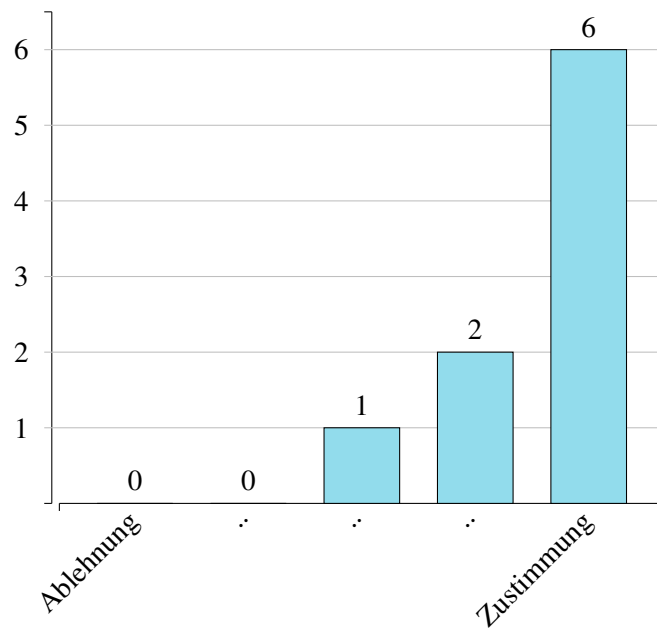


Abbildung 24: Die Plätze für die Stationen waren gut gewählt.

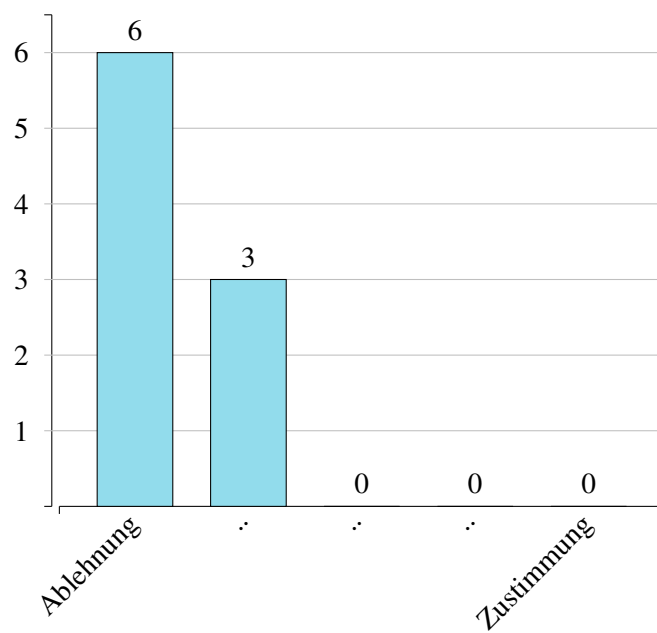


Abbildung 25: Der zeitliche Aufwand für das Spiel war zu hoch.

Die Codes waren (mit einer Ausnahme, welcher sich unter einem Blatt verbarg (vgl. Anhang)) alle mindestens zu einem Viertel sichtbar und die Probanden sollten

sie durch genaueres Hinsehen auf jeden Fall finden. Die QR-Codes wurden von den meisten Probanden als zumindest gut auffindbar eingestuft (siehe Abbildung 26). Hierbei ist es auch wichtig, dass nicht alle Probanden voll der Aussage zugestimmt haben, da die QR-Codes zwar findbar versteckt, aber auch nicht zu offensichtlich platziert sein sollten. Speziell zum Ende des Spiels, um den Spielgrad zu erhöhen. Somit bietet das Spiel eine gewisse Herausforderung und ist nicht langweilig, andererseits beeinträchtigt der Schwierigkeitsgrad auch nicht den Spielspaß.

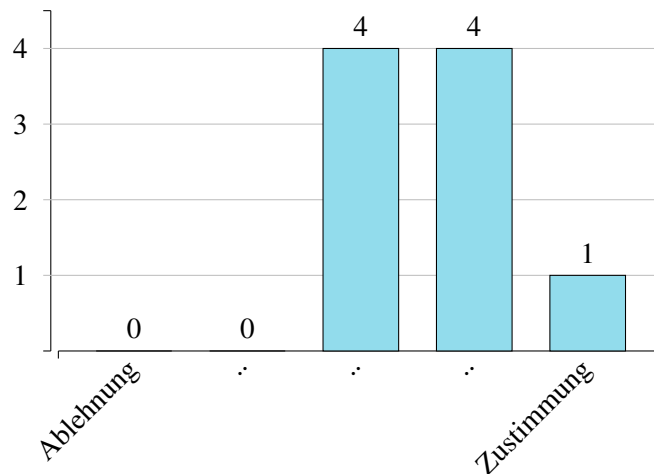


Abbildung 26: Die QR-Codes waren gut zu finden.

6.2.3 Dritter Teil

Die Campusjagd hat allen Spielern vom Prinzip her gefallen und sie würden sie weiterempfehlen. Dies kann auch damit belegt werden, dass der Gesamteindruck der App durchweg positiv war, da alle Probanden bei Abbildung 14 bis 16 mindestens neutral angekreuzt haben. Außerdem haben alle Probanden das Spiel durchgespielt. Nur ein Proband kreuzte an, eine vergleichbare App zu kennen (siehe Abbildung 27). Er nannte eine Geocaching-App für Android.

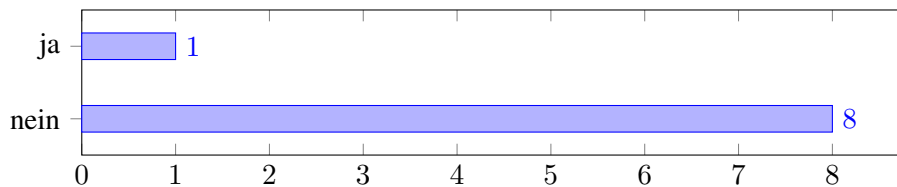


Abbildung 27: Kennen Sie vergleichbare Apps?

Sechs der neun Probanden gaben an, dass sie die Campusjagd nicht nochmal spielen würden (siehe Abbildung 28). Die Positionen der QR-Codes sind nun bekannt, das Spiel habe somit seinen Reiz verloren. Hier wird das Ergebnis der ersten

Konzeptevaluation bestätigt, das Spiel besitzt für die Mehrzahl keinen Wiederspielwert.

Die Tatsache, dass die meisten Probanden den Campus schon vor dem Spiel gut kannten (Abbildung 29), wurde bereits im ersten Teil eingehend erläutert.

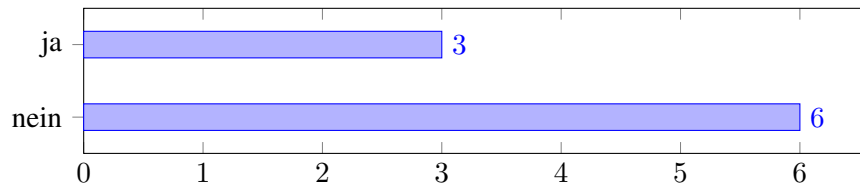


Abbildung 28: Würden Sie die App erneut nutzen?

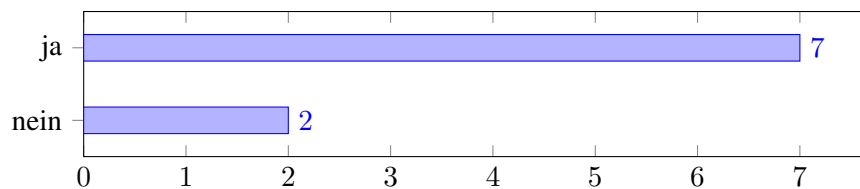


Abbildung 29: Denken Sie, dass Sie den Campus vor dem Spiel gut kannten?

Obwohl mit Abbildung 29 sieben Probanden angaben den Campus gut zu kennen, gaben alle neun Probanden an, sie hätten neue Orte auf dem Campus kennen gelernt. Die Spieler konnten hier auch mehrere Orte nennen. Darunter war das Gewächshaus wohl der am wenigsten bekannte Ort, da sechs Probanden notierten, diesen noch nicht gekannt zu haben. Danach kamen mit großem Abstand das Haus an der Schranke mit der Transponderausgabe (von drei Probanden genannt), das Methodenzentrum (von Zweien genannt) und die Blume am Ende des Spiels (von Einem genannt) (vgl. Abbildung 30). Damit kann festgehalten werden, dass selbst Spieler, welche den Campus zu kennen meinen, neue Orte durch die Campusjagd erkunden können. Somit ist sie nicht ausschließlich für Spieler geeignet, welche den Campus noch nie besucht haben, das selbst gesetzte Ziel wurde somit sogar übertroffen.

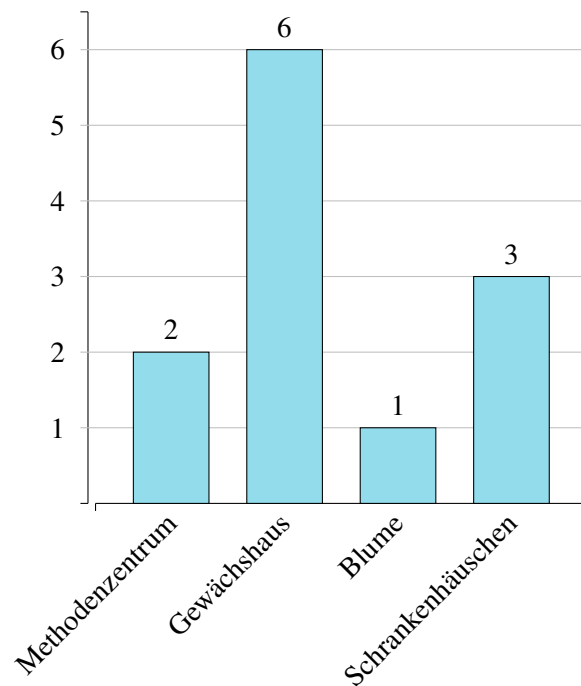


Abbildung 30: Neu kennengelernte Orte.

Fünf Probanden bejahten die Frage, ob etwas essentiell wichtiges am Spiel gefehlt habe (vgl. Abbildung 31). Diese Fünf begründeten ihre Antwort einerseits mit dem Fehlen einer Anleitung, andererseits mit mangelnden Sicherungsmöglichkeiten (vgl. Abbildung 32). Eine Einführung in die Nutzung der App haben die Probanden zwar erhalten, aber es wäre sinnvoll diese während des Spiels im Hauptmenü noch einmal abrufbar zu machen, um eine Funktion, die nicht mehr eindeutig klar ist, nachlesen zu können. Entgegen der Konzeptevaluation wäre es außerdem gut, eine Art Levelauswahl zu ermöglichen, damit der Nutzer die zuletzt gespielten Stationen noch einmal abrufen kann und nicht nochmals aufsuchen muss, falls versehentlich das aktuelle Pop-Up geschlossen wird.

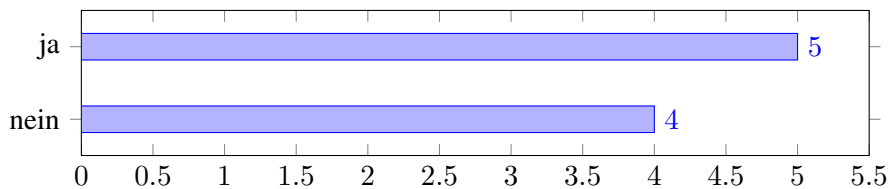


Abbildung 31: Hat Ihnen etwas wichtiges an dem Spiel gefehlt?

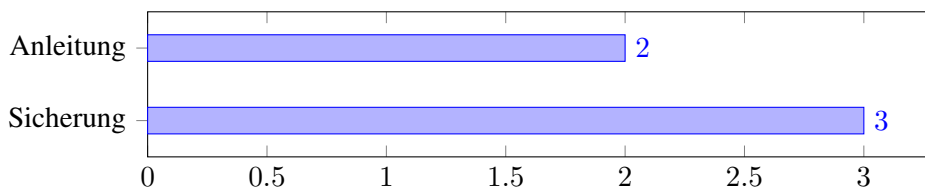


Abbildung 32: Was hat Ihnen gefehlt?

Alle Probanden können sich das Spiel als Gruppenspiel vorstellen. Genauso stimmten alle zu, das Spiel ist für einen ersten Überblick über den Campus gut geeignet. Somit könnte die Campusjagd durchaus für Erstsemester verwendet werden. Außerdem könnten diese die Applikation in Gruppen spielen, damit sie neue Leute kennen lernen und somit erste Kontakte knüpfen können. Es ist ein zusätzlicher Ansporn, wenn mehrere Gruppen das Spiel gleichzeitig in Konkurrenz spielen, etwa in Form eines Wettbewerbes.

6.3 Ausblick

Ein paar kleine Änderungen müssten an dem Prototypen noch vorgenommen werden, damit alle Richtlinien der DIN EN ISO 9241-12 umgesetzt werden. Vor allem ist es wichtig, nochmals auf die Verständlichkeit und Entdeckbarkeit der Informationen einzugehen. Ganz im Speziellen sollte dies mit einer Bedienungsanleitung zu dem Spiel passieren, welche von einigen Probanden gewünscht wurde und dazu dienen kann, die Karte und ihre verschiedenen Funktionen mehr in den Mittelpunkt des Spiels zu rücken. Zum anderen sollte die Bedienung überdacht werden, da diese für einige Nutzer nicht intuitiv genug war. Dazu sollte noch ein Test mit verschiedenen Bedien-Konzepten durchgeführt werden, um zu erfahren, welche Form der Bedienung intuitiv am sinnvollsten ist. Außerdem soll es eine Möglichkeit geben, die bereits erreichten Level nochmals aufzurufen und erneut zu Spielen. Dies kann vor allem hilfreich sein, wenn das Spiel unvollendet abgebrochen und nach einiger Zeit wieder aufgenommen wird.

Es existieren noch zahlreiche Möglichkeiten, wie dieser Prototyp erweitert und ergänzt werden könnte. Zum einen könnten Achievements eingeführt werden, dies heißt, dass ein Benutzer zu gewissen Stationen kleine Belohnungen bekommt: Beispielsweise nach der ersten Station die Karte (um auf diese nochmals Aufmerksamkeit zu lenken), daraufhin das GPS und zuletzt eine Art Radar, welches dem Spieler anzeigt, ob er in die richtige Richtung läuft, etwa anhand einer kleinen Ampel auf der Karte (rot entspricht einer großen und grün einer geringen Distanz zum Zielort). Durch letztere Idee wird der Nutzer wiederum angespornt, die Karte zu nutzen und mit ihr zu interagieren. Eine andere Erweiterung wäre, dass auf der Karte kleine Punkte angezeigt werden, die für die Stationen stehen, welche der Benutzer schon gefunden hat.

Die Software müsste auf unterschiedliche Displaygrößen angepasst werden, da sie nur auf einem Sony Xperia S und einem Samsung S3 Mini getestet wurde.

7 Fazit

Wie zu Beginn herausgefunden, ist es nicht ohne erheblichen Aufwand, z.B. nur mit regelmäßigen Updates oder geänderten Abläufen, möglich, ein Geländespiel mit den Elementen von Geocaching und Schnitzeljagd zu entwickeln, welches mehrfach nutzbar ist. Daher wurde vom ursprünglichen Konzept abgewichen und ein Spiel entwickelt, mit dessen Hilfe der Universitätscampus in Koblenz erstmals oder genauer kennen gelernt werden kann. Dennoch wurde das persönliche Ziel erreicht, eine Android-Applikation mit den vereinten Konzepten von Schnitzeljagd und Geocaching zu erstellen.

Die Anforderungen an die Applikation selbst wurden alle erfüllt. Der Benutzer des Spiels wird über den Campus geführt und kann dort alle essentiell wichtigen Gebäude und Geländestücke kennenlernen. Dazu muss er QR-Codes abscannen und kleine Aufgaben lösen.

Die Evaluation hat ergeben, dass die meisten Anforderungen an die Richtlinien der DIN EN ISO 9241-12 eingehalten wurden, die Vermittlung der Informationen und die Verständlichkeit jedoch noch ein gewisses Verbesserungspotenzial enthält. Dennoch hatten alle Probanden Spaß mit dem Spiel und könnten es sich als Gruppenspiel und als Spiel für Erstsemester sehr gut vorstellen. Das grundlegende Ziel der Arbeit wurde somit erreicht.

Anhang

Stationen und deren Fragen

Die Fragen beziehen sich jeweils auf das gezeigte Bild, also die nachfolgende Station.

Bei manchen Fragen können mehrere Antworten richtig sein, diese stehen durch ein Komma getrennt hintereinander bei „Antwort“.

Station 1 (vgl. Abbildung 33):

Dies ist die Startstation, hier wird dem Nutzer direkt das Bild für die nächste Station präsentiert.

Station 2 (vgl. Abbildung 34):

Frage: An welchem Gebäude befindet sich diese Schaukel? Nur den Gebäudebuchstaben!

Antwort: H

Station 3 (vgl. Abbildung 35):

Frage: Welche Fakultät ist in diesem Gebäude vertreten?

Antwort: Mathematik, Biologie, Chemie, Physik

Station 4 (vgl. Abbildung 36):

Frage: Welcher Fachbereich ist in diesem Gebäude anzutreffen? (Bitte nur die Zahl eintragen)

Antwort: 4

Station 5 (vgl. Abbildung 37):

Frage: Rechts ist die Tür, um in das Gebäude zu kommen. Was verbietet der Sticker auf der linken Tür?

Antwort: Hunde, Hund

Station 6 (vgl. Abbildung 38):

Frage: Was wird hier von 11-12 Uhr ausgegeben? Die _____

Antwort: Transponder

Station 7 (vgl. Abbildung 39):

Frage: Was befindet sich hier im 2. Stockwerk?

Antwort: Methodenzentrum

Station 8 (vgl. Abbildung 40 & 43):

Hier gibt es zwei Hinweise, zum erste wird keine Frage gestellt und der Anwender in eine Sackgasse geführt.

Der zweite Hinweis enthält die Frage: Wie viele Möglichkeiten gibt es, von außen an diesen Platz hinter diesem Gang zu kommen?

Antwort: 2

Station 9 (vgl. Abbildung 41):

Frage: Welches Gebäude ist hier genau gegenüber? Nur der Buchstabe!

Antwort: K

Station 10 (vgl. Abbildung 42):

Frage: Letzte Frage: Welche Farbe ist ganz oben?

Antwort: Blau

Links sind die QR-Codes der jeweiligen Station zu sehen und rechts das Pop-Up, welches die Applikation anzeigt, wenn dieser QR-Code abgescannt wird.

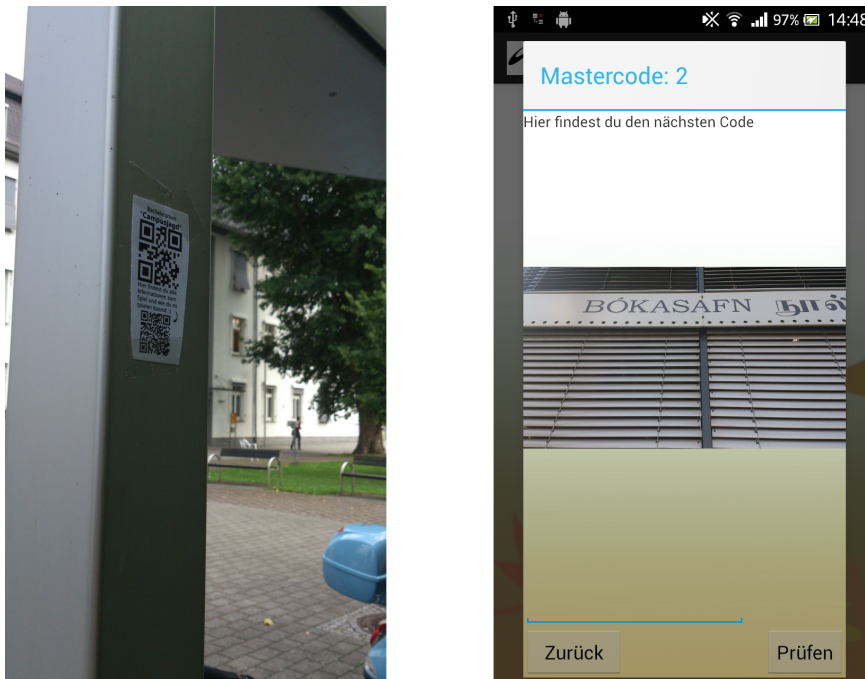


Abbildung 33: Station 1

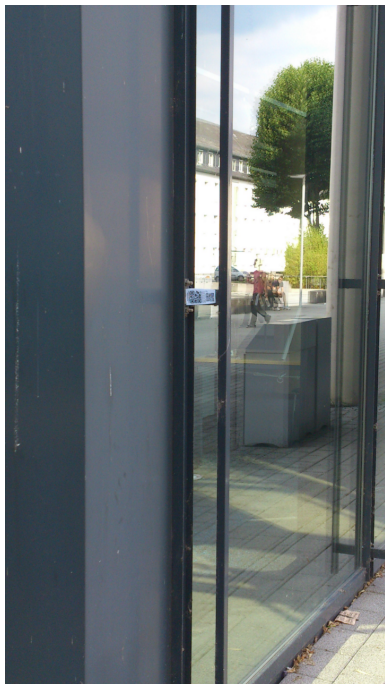


Abbildung 34: Station 2

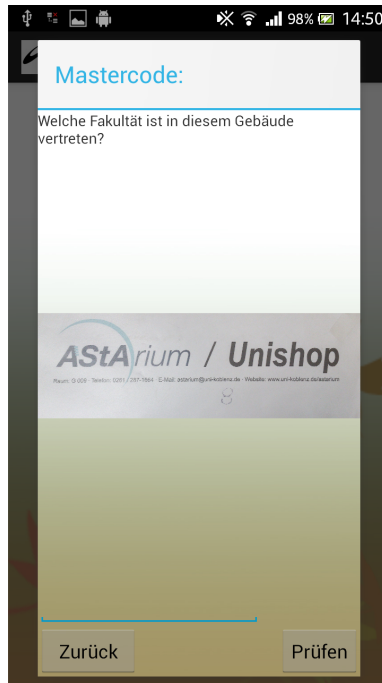


Abbildung 35: Station 3

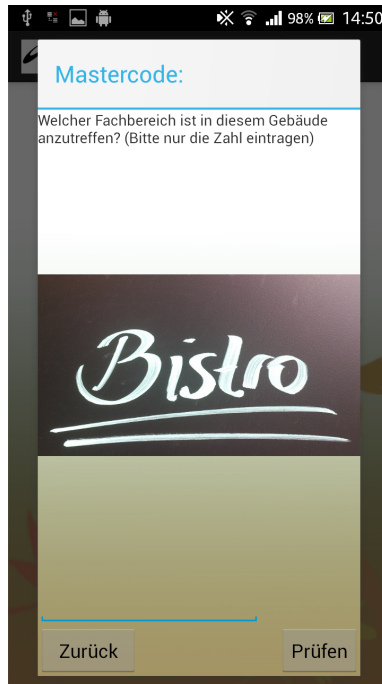


Abbildung 36: Station 4

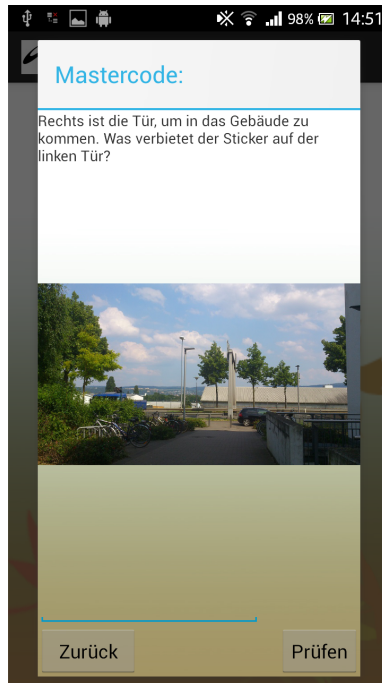


Abbildung 37: Station 5

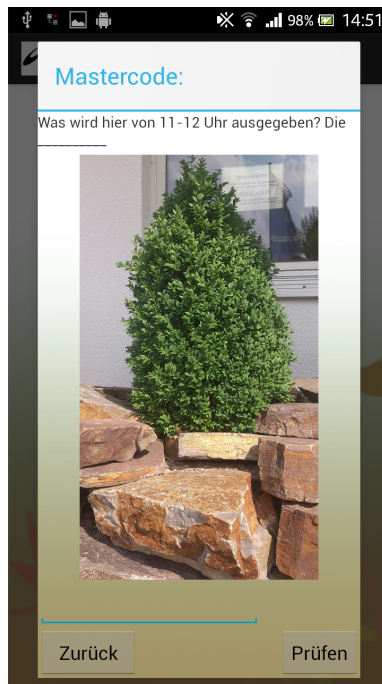


Abbildung 38: Station 6

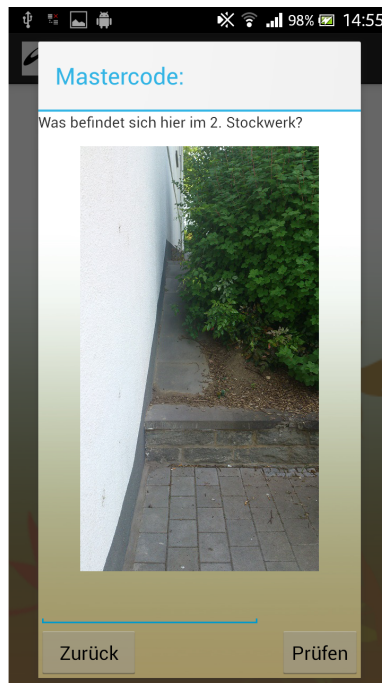


Abbildung 39: Station 7

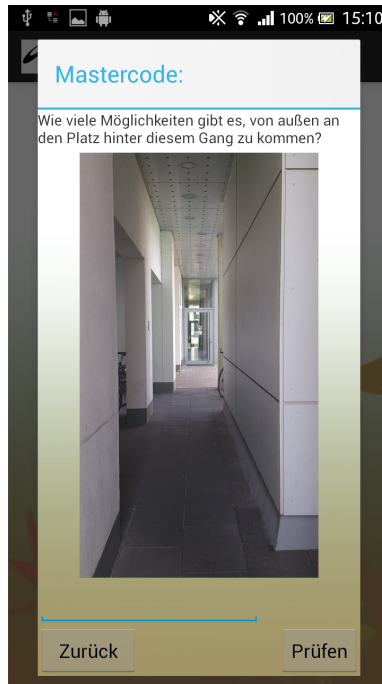


Abbildung 40: Station 8

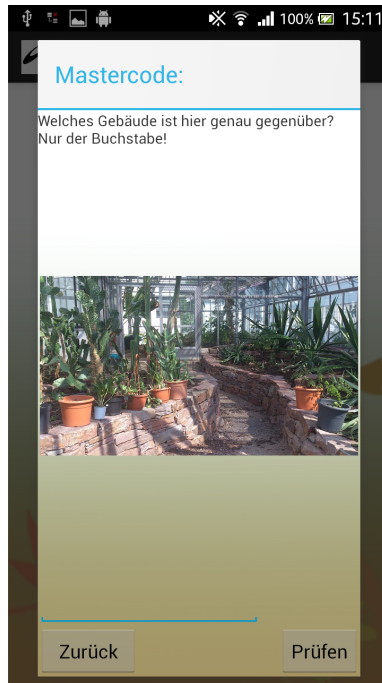


Abbildung 41: Station 9



Abbildung 42: Station 10

Eine Besonderheit stellt die Station auf Abbildung 43, welche eine Sackgasse ist.

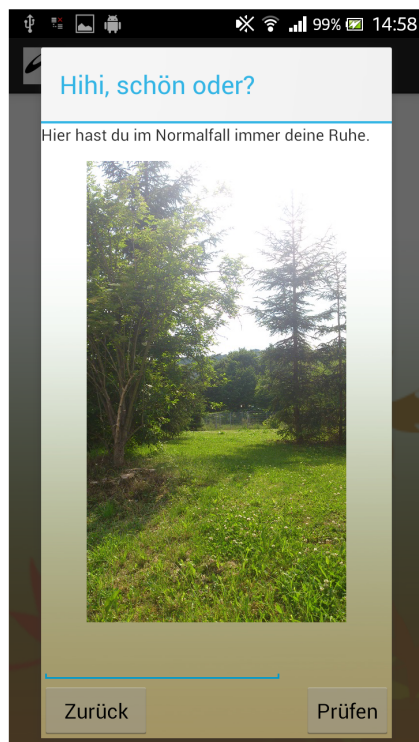
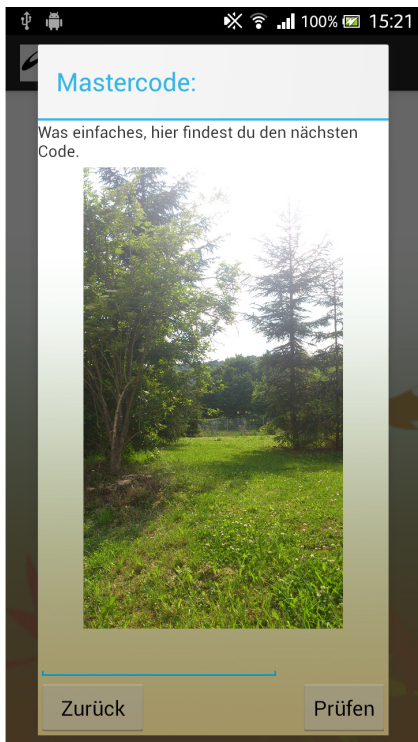


Abbildung 43: Die falsche Station

Fragebogen zur Bachelorarbeit „Campusjagd“

Bitte Zutreffendes ankreuzen!

Persönliches

1. Studieren Sie in Koblenz?

Ja Nein

Wenn ja, in welchem Semester: _____

2. Haben Sie ein eigenes Smartphone?

Ja Nein

Wenn nein, sind Sie mit der Bedienung vertraut? Ja Nein

Inwiefern stimmen Sie folgenden Aussagen zu?

Stimme nicht zu. ———— *Stimme zu.*

1. Das Konzept hat mir gefallen. ————

2. Die Umsetzung war gut. ————

3. Das Spiel hat mich unterhalten. ————

4. Die einzelnen Stationen waren viel zu weit auseinander. ————

5. Die Bedienung war intuitiv. ————

6. Ich kam gut mit der Bedienung klar. ————

7. Der zeitliche Aufwand für das Spiel war zu hoch. ————

8. Das Spiel hatte zu viele Stationen. ————

9. Ich habe das GPS oft genutzt. ————

10. Ich habe die Actionbar öfters benutzt als das Menü. ————

11. Die Aufgaben fand ich im Schwierigkeitsgrad angemessen. ————

12. Die Aufgaben haben den Spielfluss gestört. ————

13. Die Plätze für die Stationen waren gut gewählt. ————

14. Die QR-Codes waren gut zu finden. ————

Fragen zum Spiel

1. Hat Ihnen das Spielprinzip gefallen?

- Ja Nein

2. Haben Sie das Spiel durchgespielt?

- Ja Nein

Wenn nein, warum nicht? _____

3. Kennen Sie vergleichbare Apps?

- Ja Nein

Wenn ja, welche? _____

4. Würden Sie die App weiterempfehlen?

- Ja Nein

5. Würden Sie die App erneut nutzen?

- Ja Nein

Wenn nein, warum nicht? _____

6. Denken Sie, dass Sie den Campus vor dem Spiel gut kannten?

- Ja Nein

7. Haben Sie durch das Spiel neue Orte kennengelernt?

- Ja Nein

Wenn ja, welche? _____

8. Hat Ihnen etwas wichtiges an dem Spiel gefehlt?

- Ja Nein

Wenn ja, was? _____

9. Können Sie sich das Spiel als Gruppenspiel vorstellen?

- Ja Nein

10. Das Spiel soll Erstsemestern einen ersten Überblick über den Campus geben. Denken Sie, dass das Spiel für diesen Zweck hilfreich ist?

- Ja Nein

Literatur

- [1] <http://www.startmobile.net/die-meilensteine-in-der-geschichte-der-handys-und-mobilfunktelefone/>;
19.07.2014, 14:11.
- [2] <http://www.ruhrnachrichten.de/nachrichten/topnews/Entwicklung-des-Handys-Vom-Knochen-zum-Smartphone;art333,2116141>;
19.07.2014, 14:01.
- [3] <http://www.pcwelt.de/ratgeber/Handy-Historie-Wie-alles-begann-Die-Geschichte-des-Smartphones-5882848.html>;
19.07.2014, 14:13.
- [4] http://www.chip.de/artikel/Handy-Betriebssysteme-Android-iOS-Symbian-Windows-Phone-7-bada_36053670.html;
29.07.2014, 21:11.
- [5] <http://www.kantarworldpanel.com/global/News/Apple-regains-momentum-as-Windows-stutters>;
19.07.2014, 13:35.
- [6] <http://www.macerkopf.de/2014/04/28/ios-android-und-co-neue-weltweite-marktanteile/>;
19.07.2014, 13:25.
- [7] <http://forum.xda-developers.com/showthread.php?t=1423091>;
19.07.2014, 13:45.
- [8] <http://www.golem.de/news/andy-rubin-android-war-urspruenglich-fuer-digitalkameras-gedacht-1304-98779.html>;
19.07.2014, 13:21.
- [9] <https://source.android.com/devices/tech/security/images/image00.png>;
27.07.2014, 15:46.
- [10] <http://www.android-user.de/Magazin/Android-User/2013/06/Unterschiede-und-Gemeinsamkeiten-zwischen-Android-und-Linux>;
29.07.2014, 21:31.

- [11] <http://4.bp.blogspot.com/-SaNLCPsdI3g/UhHMYXMT-HI/AAAAAAAAAFF0/CYGcvV05LpA/s1600/Screen%2BShot%2B2013-08-19%2Bat%2B%25E4%25B8%258B%25E5%258D%25883.41.36.png>;
27.07.2014, 17:46.
- [12] <http://www.ukpt.de/pages/dateien/software-ergonomie.pdf>;
11.08.2014, 15:44.
- [13] <http://www.uni-koblenz-landau.de/uni/anfahrt/campus-koblenz/campusplan.png/image>;
05.08.2014, 16:06, verändert.
- [14] Sven Haiges. *Android Schnelleinstieg*. entwickler.press, 2011.
- [15] Andrew Hoog. *Android Forensik*. Franzis Verlag GmbH, 2012.
- [16] Thomas Künneth. *Android 3*. Galileo Press, 2011.