

Institut für Visualisierung und Interaktive Systeme

Universität Stuttgart
Universitätsstraße 38
D-70569 Stuttgart

Bachelorarbeit Nr. 78

Gesturegame: Ermittlung der Basiseigenschaften von Gesten auf Touchscreens mittels einer Anwendung für Smartphones

Vivian Eggert

Studiengang:	Softwaretechnik
Prüfer/in:	Prof. Dr. Albrecht Schmidt
Betreuer/in:	Dr. Niels Henze, Alireza Sahami, M. Sc.
Beginn am:	3. Juni 2013
Beendet am:	3. Dezember 2013
CR-Nummer:	H.5.2

Kurzfassung

Gestensteuerungen von Touchdisplays sind aus unserem alltäglichen Leben nicht mehr weg zu denken. Deshalb ist es umso wichtiger zu wissen, welche Art von Gesten für eine bestimmte Aufgabe geeignet sind. Wird eine Geste an einer schwer erreichbaren Stelle platziert, so wird es den Nutzer auf Dauer frustrieren, diese Geste ausführen zu müssen. Der Nutzer wird sich nach alternativen Benutzungsmöglichkeiten umschaun. Falls keine vorhanden sind, wird der Nutzer das System nicht mehr nutzen wollen. Aber auch die Art der Geste ist entscheidend, wie zum Beispiel die Form, Länge und Richtung könnten eine entscheidende Rolle bei der Durchführungseffizienz einer Geste spielen. Dieser Thematik nimmt sich diese Arbeit an. Sie stellt eine Anwendung für touchfähige Mobiltelefone bereit und bietet diese zum Download im Google Play Store an. Um einen größeren Personenkreis anzusprechen, wurde die Anwendung als Spiel entwickelt. Desweiteren wurde eine Laborstudie mit derselben Anwendung, unter gleichen Bedingungen, durchgeführt. Diese Anwendung zeigt dem Nutzer verschiedene geradlinige Gesten an, die dieser dann ausführen muss. Anhand von Geschwindigkeit und Fehlerwerten kann ermittelt werden, wie effizient eine Geste ausgeführt wurde. Die gesammelten Daten werden zentral in einer Datenbank gespeichert. Die Auswertung hat ergeben, dass die Richtung der Geste kaum einen Einfluss auf die Effizienz der Geste hat. Auch die Orientierung der Startgeste hat keinen Einfluss auf die Ausführungseffizienz. Bei einer Verdoppelung der Gestenlänge kann keine Verdoppelung der Ausführungszeit erkannt werden, aber dennoch ein Anstieg bei Verlängerungen der Gesten.

Abstract

Touchscreens are the most used input devices on mobile phones and tablets. Because of this it is very important to know which gestures are suited for a task. When the gesture is placed in an area of the touchscreen that is hard to reach, the usability of the application decreases. Not only the placement of the gesture is important, also the form of the gesture is an important factor for the usability. Gestures may differ in form, length or direction and increase or decrease its usability. This is the subject of this work. An Application for mobile devices with touchscreens was built and deployed via the Google Play Store. The Application was designed as a game to increase the number of potential users. After the Google Play Store study was completed a second user study was performed under constant conditions. The Application shows several gestures the player has to execute in order to progress in the game. A gesture contains a variable amount of lines. A gesture has at least one line and a maximum of four lines. The execution of the task will give information about the execution time and the error value. The effectiveness and efficiency of a gesture can be determined by these values. The collected data are stored in a database. The study has shown that the direction of a gesture has no effect on the effectiveness and efficiency of a gesture. The orientation of the startgesture does not affect the performance either. The study

show that a gesture that is twice as long does not require twice the amount of time, only a small increase.

Inhaltsverzeichnis

1	Einleitung	7
1.1	Ziel	7
1.2	Gliederung	7
2	Grundlagen	9
2.1	Verwandte Arbeiten	9
2.2	Technische Grundlagen	10
3	Konzept	13
3.1	Gesten	13
3.2	Android-Spiel	13
3.3	Datenerhebung	14
3.4	Spieler	14
3.5	Belohnungen	14
4	Implementierung der Systemkomponenten	17
4.1	Androidanwendung	17
4.2	Webserver	26
4.3	Datenbank	26
5	Studie	29
5.1	Testphase	29
5.2	Google Play Store Benutzerstudie	30
5.3	Laborstudie	31
6	Auswertung	33
6.1	Google Play Store Benutzerstudie	33
6.2	Labor Benutzerstudie	41
6.3	Diskussion und Vergleich	47
7	Zusammenfassung und Ausblick	49
	Literaturverzeichnis	51

Abbildungsverzeichnis

4.1	Struktur des Gesamtsystems	17
4.2	Layout der wichtigsten Activities der Anwendung	22
4.3	verschiedene Layouts der Spielactivities	23
4.4	Entity-Relationship-Model der Datenbank	27
5.1	Google Play Store Screenshots	30
6.1	Anzahl der absolvierte Stages	34
6.2	Anzahl der durchgeführten Spiele	35
6.3	Verschiedene Richtungen der Gesten	36
6.4	Startsegmentorientierung der Gesten	37
6.5	Segmentanzahl der Gesten	38
6.6	Nutzungsstunden eines Touchdisplays	39
6.7	Verbesserung während der Ausführung	40
6.8	Druck/Größe von Gesten mit einem Segment	41
6.9	Richtungen in der Laborstudie	42
6.10	Startsegmentorientierung in der Laborstudie	43
6.11	Segmentanzahl in der Laborstudie	44
6.12	Segmentanzahl der besten drei Spieler in der Laborstudie	45
6.13	Nutzungsstunden eines Touchdisplays in der Laborstudie	46
6.14	Druck/Größe in der Laborstudie	47

Verzeichnis der Listings

4.1	Algorithmus, der die Abweichung von der Eingabe zur vorgegebenen Geste berechnet	19
4.2	Fehlerwertberechnung in der Geste	20
4.3	Algorithmus, der die Punkteverteilung, die aus der Toucheingabe resultiert, berechnet	20

1 Einleitung

Mobile Endgeräte sind aus unserem heutigen Leben nicht mehr wegzudenken. Seit dem das Iphone und andere vergleichbare Geräte mit Android erfolgreich eingeführt wurden. Wird bei den meisten Geräten deshalb auf Verwendung von Hardwaretasten komplett verzichtet. So kann das Gerät durch die Benutzung eines Touchdisplays bedient werden. Aus diesem Grund werden häufig gestenbasierte Eingaben zur Steuerung von Anwendungen verwendet. Durch die starke Verbreitung von Smartphones und der großen Auswahl an Anwendungen, häufen sich mit der Zeit die Anwendungen auf dem Gerät. Diese Anwendungen haben teilweise sehr unterschiedliche Bedienkonzepte. Handelt es sich um Informationsbeschaffungsanwendungen, so wird sich die Navigation auf wenige einfache Bewegungen oder Tastatureingaben beschränken. Die Bedienung von komplexeren Systeme soll dennoch angenehm und motivierend sein, deshalb stellt sich die Frage, welche Gesten für welche Aufgaben geeignet sind. Schlecht gewählte Gesten führen zu Frustration bei der Benutzung und können somit unter Umständen dem Erfolg einer Anwendung im Wege stehen. Dadurch könnte der Nutzen einer Anwendung drastisch gesenkt werden.

Wie auch frühere Arbeiten, untersucht diese Arbeit Gesten auf Smartphones. Mit Hilfe von Benutzerstudien wird die Durchführung vorgegebener Gesten betrachtet und ausgewertet.

1.1 Ziel

Das Ziel dieser Arbeit ist es, die Effizienz und Effektivität einer großen Menge an Gesten systematisch zu untersuchen. Natürlich ist auch das Ausführungsgerät an der Effizienz der Geste ausschlaggebend. Dabei soll auch aufgezeigt werden, ob sich Probleme bei der Ausführung bestimmter Gesten ergeben. Für die Realisierung werden zwei Benutzerstudien mit Hilfe einer Anwendung für Smartphones durchgeführt. Bei diesen Benutzerstudien führen die Probanden Gesten auf ihren Smartphones aus. Diese Ausführungen werden für eine spätere Auswertung gespeichert. Um eine große Menge an Probanden zu erreichen, wird die Anwendung über den Google Play Store verteilt. Haben die Studien genügend Daten gesammelt, werden diese aufbereitet, visualisiert und ausgewertet.

1.2 Gliederung

Die Arbeit ist in folgender Weise gegliedert:

Kapitel 1 – Einleitung: Einleitung dieser Arbeit

Kapitel 2 – Grundlagen Beschreibt die technischen Grundlagen die für diese Arbeit relevant sind

Kapitel 3 – Konzept In diesem Kapitel wird die Konzeptionsphase der einzelnen Komponenten beschrieben

Kapitel 4 – Implementierung der Systemkomponenten Dieses Kapitel beschäftigt sich mit der implementierung der für diese Arbeit nötigen Komponenten.

Kapitel 5 – Studie Beschreibt das Vorgehen bei der Durchführung der Benutzerstudien

Kapitel 6 – Auswertung Stellt die gesammelten Daten, die während der Benutzerstudie gesammelt wurden, aufbereitet dar.

Kapitel 7 – Zusammenfassung und Ausblick Fasst die Arbeit zusammen und gibt einen Ausblick für spätere Arbeiten

2 Grundlagen

Dieses Kapitel erläutert die technischen Grundlagen, die nötig sind um diese Arbeit zu verstehen. Desweiteren wird auf verwandte Arbeiten bezug genommen.

2.1 Verwandte Arbeiten

In diesem Abschnitt werden verwandte Arbeiten beschrieben, die sich mit den Thematiken der Toucheingabe auf Smartphones oder der Durchführung von Benutzerstudien befassen. Da die Toucheingabe heutzutage die meist verbreitete Form der Eingabemöglichkeiten bei Smartphones ist, existieren aus diesem Grund viele Arbeiten, die sich mit dieser Thematik beschäftigen. Wie zum Beispiel die Arbeit von Henze et al. [HRB11] In dieser Arbeit wurde ebenfalls ein Spiel mit dem Namen „Hit it!“ für Androidsysteme entwickelt. Mit Hilfe der Anwendung werden Daten gesammelt, die später ausgewertet werden. Das Spielprinzip ähnelt dem von „GestureGame“, welches Teil dieser Arbeit ist. Es besteht aus mehreren Leveln in denen der Nutzer vorgegebene Aufgaben in einem vorgegebenen Zeitfenster erledigen muss. Dem Nutzer werden in jedem Level eine bestimmte Anzahl von Kreisen vorgelegt, die dieser dann in dem Zeitfenster berühren muss. Auch bei dieser Anwendung wurde der Google Play Store genutzt, um die Anwendung an einen großen Personenkreis zu verteilen. „Hit it!“ hat insgesamt 120 Millionen Toucheingaben von 91.731 Installationen gesammelt, was eine beträchtliche Anzahl an Daten ist. „Hit it!“ hat sich auch die Fehlerrate der Toucheingaben angesehen und festgestellt, dass die Fehlerraten an den Rändern des Displays größer als in der Mitte des Displays sind. Desweiteren konnte festgestellt werden, dass es an der rechten unteren Ecke des Displays zu einer Verschiebung der Eingabe nach oben links kam. Dementsprechend gab es an der oberen linken Ecke des Displays eine Verschiebung nach unten rechts.

Im Gegensatz zu dieser Arbeit, sammelt „Hit it!“ Toucheingaben, die ohne Bewegung auf dem Touchdisplay, durchgeführt werden. Dabei wird die Eingabe in Abhängigkeit von Zielgröße und Position betrachtet.

Eine andere Arbeit von Henze et al. [HRB12] hat „Type It!“, ein ähnliches Spiel für Android, entwickelt. „Type It!“ sammelt die Koordinaten der Toucheingaben auf der virtuellen Tastatur von dem Android-Betriebssystem. Auch dieses Spiel wurde im Google Play Store verbreitet und konnte so 47.770.625 Toucheingaben sammeln, die das Eingabeverhalten der Nutzer zeigen.

Die Arbeit von Tran et al. [TTS⁺13] beschäftigt sich auch mit Toucheingaben auf mobilen Endgeräten. Die Studie wurde unter einer beobachtbaren Umgebung durchgeführt und hat Gesten untersucht, die mit zwei fingern ausgeführt und entweder zusammen oder

auseinander gedrückt werden. Dabei wurde festgestellt, dass die Orientierung des Gerätes keine Auswirkung auf die Ausführung hat.

Die Arbeit von Shirazi et al. [SSHD⁺13] bezieht ihre Daten ebenfalls in einer Benutzerstudie, bei der die Probanden die Anwendung aus dem Google Play Store bezogen haben. Somit kann die Arbeit auf Daten von 1330 verschiedenen Geräten zurückgreifen. Die Datenmenge die durch eine im Google Play Store veröffentlichten Anwendung erhoben werden kann, ist enorm und kann bei der richtigen Betrachtung ein aussagekräftiges Ergebnis liefern.

Die bisher genannten Arbeiten haben sich alle auf Gesten, die auf einem Touchdisplay ausgeführt werden, bezogen. Auch andere Bedienkonzepte sind untersucht worden. So befasst sich zum Beispiel die Arbeit von Henze et al. [HLB⁺10], mit der Gestensteuerung durch statische oder dynamische Handzeichen. Dabei dient eine Kamera als Aufnahmegerät der Gesten. Die Tatsache, dass immer mehr Smartphones eine Frontkamera verbaut haben, macht diese Art der Steuerung interessant für Smartphones.

Die Arbeit von Bau und Mackay [BM08] beschäftigt sich mit der Thematik des Gestenlernens durch „feedforward“ und „feedback“, so wird dem Nutzer der weitere Verlauf von möglichen Gesten, sowie die bisher durchgeführte Geste, auf dem Bildschirm angezeigt. Durch diese Technik soll der Nutzer die Gesten schneller erlernen können.

Auch die Arbeit von Wobbrock et al. [WMW09] befasst sich mit Gestensteuerung durch Händen auf Oberflächen. Dabei konnten die Probanden ihre Geste für Aktionen selber definieren, damit sollte ein besseres Verständnis des Benutzerverhalten entstehen.

2.2 Technische Grundlagen

In diesem Abschnitt werden die technischen Grundlagen, die zum Verständnis dieser Arbeit notwendig sind, geschildert und erläutert. Es wird auf Android sowie auf die Grundlagen der Benutzung von Touchdisplays eingegangen. Desweiteren wird ein kleiner Einblick in die Funktionsweise einer Datenbank vermittelt.

2.2.1 Android

Android ist ein Linux-basierendes Betriebssystem, das für touchfähige mobile Endgeräte konzipiert ist, wie zum Beispiel Smartphones und Tablets. Entwickelt wird das System von der Open Handset Alliance ¹, welches Google als Hauptmitglied trägt. Android ist ein freies und quelloffenes Betriebssystem. Im dritten Quartal 2013 hat Android einen Weltmarktanteil von 81% ². Um für Android Programme zu entwickeln, wird das Android-SDK ³ zusätzlich neben dem Java-SDK benötigt. Die erzeugten Programme stehen dann im Android-Package (APK) zur Verfügung und können entweder direkt auf einem Zielgerät installiert, oder in

¹Open Handset Alliance Webseite: <http://www.openhandsetalliance.com/> zuletzt geprüft am 27.11.2013

²IDC Webseite <http://www.idc.com/getdoc.jsp?containerId=prUS24442013> zuletzt geprüft am 27.11.2013

³Android-Webseite <http://developer.android.com/sdk/index.html> zuletzt geprüft am 27.11.2013

den Google Play Store hochgeladen werden. Im Google Play Store befinden sich ca. 700 000 Anwendungen, die dort kostenfrei oder gegen Gebühren heruntergeladen werden können. Androidanwendungen bestehen aus einer oder mehreren sogenannten „Activities“. Eine Activity stellt ein grafisches Layout bereit und verwaltet die Programmlogik auf dieser Activity.

2.2.2 Gesten auf Touchdisplays

Eine Geste auf einem touchfähigen Endgerät bezeichnet die Bewegung des Fingers auf dem Bildschirm. Dabei muss beachtet werden, dass es verschiedene Arten von Touchdisplays gibt. Bei Smartphones und Tablets wurden meistens resistive oder kapazitive Touchdisplays verbaut. Bei resistiven Displays handelt es sich um einen auf Druck reagierenden Bildschirm. Sobald Druck auf einer Stelle erzeugt wird, verbinden sich zwei Schichten im Display und es fließt Strom an dieser Stelle. Ein Vorteil dieser Bauweise ist die Tatsache, dass das Display auch mit Handschuhen oder Gegenständen bedient werden kann. Allerdings sind nützliche Eigenschaften wie Multitouch nur begrenzt bis garnicht möglich. Die bei Smartphones weit verbreitete Technik des kapazitiven Displays funktioniert mit einem elektrischen Feld, das über dem gesamten Display liegt. Durch Berührung entsteht ein geringer Ladungstransport, so kann der genaue Berührungspunkt ermittelt werden. Der Vorteil dieser Technik ist eine angenehmere Bedienung bei Gesten wie Scrollen, da kaum Druck auf den Bildschirm ausgeübt werden muss, um eine Aktivierung des Displays zu erzielen. Im Folgenden werden androidspezifische Verwendungen dieser Displays geschildert.

Eine Geste besitzt X- und Y-Koordinaten, wobei der Nullpunkt an der oberen linken Ecke des Touchdisplays liegt. Von diesem Punkt aus gesehen haben Bewegungen nach rechts eine positive X-Koordinatenänderung und Bewegungen nach unten eine positive Y-Koordinatenänderung zur Folge. Desweiteren bietet die Android API die Möglichkeit den Druck, mit dem die Geste ausgeführt wird, zu bestimmen. Sofern das Gerät dies unterstützt. Auch der Umfang des Druckpunktes kann gemessen werden. Eine Geste besteht aus drei Teilen. Zuerst wird das Koordinatenpaar des Anfangspunktes gespeichert, dann folgen die Koordinatenpaare der tatsächlichen Bewegung und zum Schluss wird noch das letzte Koordinatenpaar als Ende der Geste definiert.

2.2.3 Datenbank

Datenbanken werden genutzt um Daten langfristig zu speichern. In dieser Arbeit wird die Datenbank MySQL ⁴ verwendet. MySQL ist die am meisten verbreitete relationale Datenbank weltweit. Mit Datenbankbefehlen können Daten gespeichert und ausgelesen werden. Das Ergebnis wird immer als Tabelle zurückgeliefert. Mithilfe von Schnittstellen wie der Skriptsprache PHP, können diese Daten dann von einer Androidanwendung ausgelesen oder geschrieben werden. Eine MySQL Datenbank kann unterschiedliche Tabellen enthalten.

⁴MySQL-Webseite: <http://www.mysql.com/> zuletzt geprüft am 27.11.2013

Jede Tabelle sollte mindestens einen Primärschlüssel enthalten, um sicherzustellen, dass jede Reihe einzigartig ist. Auch tabellenübergreifende Zuordnungen sind möglich, da Primärschlüssel an einen Fremdschlüssel einer anderen Tabelle gebunden werden können. In dieser Arbeit wird eine MySQL Datenbank verwendet, um langfristig die Daten aller Gesten zu speichern, die von der entwickelten Androidanwendung gesammelt werden.

3 Konzept

In diesem Kapitel wird erläutert, welche Entscheidungen in der Konzeptionsphase bezüglich der einzelnen Systemkomponenten getroffen wurden.

3.1 Gesten

Das Ziel dieser Arbeit ist es, systematisch den Einfluss von verschiedenen Faktoren auf den Ausführungserfolg von Gesten zu untersuchen. Dabei spielen folgende Faktoren eine Rolle:

- Die Position, an der die Geste durchgeführt werden muss.
- Die Länge der Geste.
- Die Form der Geste.
- Die Zeit, in der die Geste ausgeführt wird.
- Die Richtung, in der die Geste ausgeführt wird.

Bei der Entscheidung, welche Art von Gesten diese Arbeit betrachtet, fiel die Wahl auf Gesten die aus Liniensegmenten zusammengesetzt sind, wobei die Segmente an ihren gemeinsamen Punkten nur in einem rechten Winkel zusammengesetzt werden. Zusätzlich dürfen diese Liniensegmente nur horizontal oder vertikal auftreten. Die einzelnen Liniensegmente haben auf einem Gerät alle dieselbe Länge. Dabei soll die Frage, ob für eine Geste, die aus einem oder mehreren Segmenten besteht, ein linearer Zeitanstieg bei der Durchführung besteht, näher untersucht werden. Auf runde oder schräge Gesten wurde verzichtet, da Gesten mit unterschiedlicher Form nicht aussagekräftig miteinander verglichen werden können.

3.2 Android-Spiel

Die Entscheidung eine Androidanwendung zu bauen, um sich mit dieser Thematik zu befassen, hatte mehrere Gründe. Zum einen ist das Android-Betriebssystem das am weitesten verbreitete Betriebssystem für mobile Endgeräte weltweit. Zusätzlich gibt es eine Vielzahl an kostenlosen Entwicklungswerkzeugen, was die Entwicklung von Androidanwendungen sehr attraktiv macht. Desweiteren besteht die Möglichkeit, eigene Anwendungen einer großen Anzahl an Personen anzubieten. Das kann durch den Google Play Store erzielt werden, oder über das Verteilen einer „APK“ an ausgewählte Nutzer. Um die Anwendung attraktiver

zu gestalten fiel die Entscheidung darauf, die Gesten in einem Spiel einzubetten, um so eine größere Verbreitung der Anwendung hervorzurufen. Der Spaßfaktor sollte für den Nutzer im Vordergrund stehen, damit der Nutzer so viele Gesten wie möglich durchführt. Allerdings sollen im Hintergrund sinnvolle und nützliche Daten gesammelt werden.

3.3 Datenerhebung

Die relevanten Daten, die von der Anwendung erhoben werden, sind folgende: Es sollten die variablen Faktoren wie Position, Länge, Art und Richtung der Geste in Abhängigkeit von Zeit und Genauigkeit betrachtet werden. Desweiteren sollen auch alle geräteabhängigen Metadaten gesammelt werden, damit bei der Evaluierung der gesammelten Daten gegebenenfalls auf die Metadaten zurückgegriffen werden kann. Damit soll sich zeigen, ob sich bei unterschiedlichen Displaygrößen oder unterschiedlichen Geräten gravierende Unterschiede herausstellen. Da die Anwendung möglicherweise von vielen verschiedenen Benutzern gleichzeitig benutzt wird, fiel die Entscheidung der Datenspeicherung auf eine zentrale Datenbank. Das hat den Vorteil, dass die Daten sowohl schnell gelesen als auch geschrieben werden können, und auch spätere Änderungen der Datenstruktur ohne größeren Aufwand implementiert werden können. Zur späteren Selektierung der Daten sind Datenbanken auch vorteilhaft, da mittels Datenbankabfragen eine gefilterte Tabelle mit allen relevanten Daten ausgegeben werden kann.

3.4 Spieler

Eine bestimmte Personengruppe wurde für diese Arbeit nicht in betracht gezogen. Jede Person, die ein touchfähiges Android-Smartphone besitzt, kann ein potenzieller Nutzer der Anwendung sein. Da diese Anwendung im Google Play Store veröffentlicht werden soll, kann sie von jeder Person die Zugriff auf den Google Play Store hat, heruntergeladen und benutzt werden.

3.5 Belohnungen

Damit der Spieler diese Androidanwendung möglichst lange und oft benutzt, soll er während des Spielens motiviert werden. Aus diesem Grund ist das Punktesystem ein Bestandteil des Spieles geworden. Dem Spieler wird nach einer Gestenausführung angezeigt, ob er die Geste gut ausgeführt hat oder nicht. Dies wird visualisiert, indem farbige Kreise auf dem Spielfeld auftauchen und nach oben wandern. In diesen Kreisen befindet sich die Punktzahl, die der Spieler mit dieser Gestenausführung erreicht hat. Die Sättigung der Farbe gibt an, wie die Ausführung der Geste war. Es gibt „grün“ für gute Ausführungen und „rot“ für schlechte Ausführungen und je satter die Farben, umso besser oder schlechter war die Ausführung. Die gesammelten Punkte der einzelnen Ausführungen werden addiert und bilden somit

den Highscore. Die Liste der Highscores aller Spieler kann aus der Anwendung heraus betrachtet werden. Somit kann sich der Spieler mit sich selbst und mit anderen Spielern messen und vergleichen. Diese Tatsache soll die Langzeitmotivation fördern und den Spieler dazu animieren, viele Gesten durchzuführen.

4 Implementierung der Systemkomponenten

Dieses Kapitel beschäftigt sich mit der Thematik der Implementierungsphase und erläutert, welche Komponenten entstanden sind, und wie sie miteinander agieren. Desweiteren werden die Klassenstrukturen der Androidanwendung, das Datenbankschema und die Funktionen des Webservers näher veranschaulicht. Ebenso werden Auszüge des Quellcodes näher erläutert. Der komplette Quellcode der Androidanwendung wird dieser Arbeit als Anhang beigelegt.

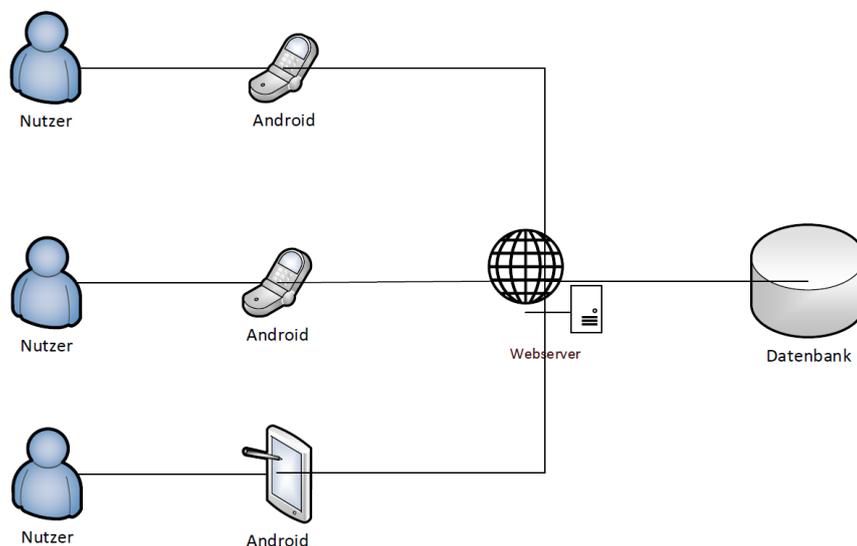


Abbildung 4.1: Diese Abbildung zeigt die Struktur des Gesamtsystems.

In Abbildung 4.1 wird das Gesamtsystem dargestellt. Die Nutzer können die Anwendung auf ihr Android Smartphone oder Tablet installieren. Die Anwendung verbindet sich dann während der Nutzung mit dem Webserver und speichert die Daten in der Datenbank.

4.1 Androidanwendung

Die Androidanwendung wurde mit den Android Developer Tools in Version 22.0.1 entwickelt. Diese Android Developer Tools beinhalten das Android SDK in Version 22.0.1 und Eclipse in

der Version 4.2 (Juno). Die Anwendung wurde während der Implementierungsphase auf einem HTC Wildfire S sowie einem Asus Nexus 7 ausgeführt und getestet.

4.1.1 Nichtfunktionale Anforderungen

Die Androidanwendung soll ab API 7 (Android 2.1) lauffähig sein, da diese API-Version alle benötigten Eigenschaften zur Realisierung der Anwendung besitzt. Desweiteren wurde eine diese API gewählt, da somit ein größerer Benutzerkreis angesprochen werden kann. Da es viele unterschiedliche Android-Geräte gibt, die auch unterschiedlich große Displaygrößen und Auflösungen haben, wurde bei der Implementierung darauf geachtet, dass die Layouts und die Gesten auf allen Gerätegrößen gleich aussehen.

4.1.2 Funktionale Anforderungen

Die Androidanwendung soll Toucheingaben (in Form von Gesten) des Nutzers registrieren und unverändert in eine Datenbank speichern, zusätzlich werden Metadaten des Gerätes, auf dem die Anwendung installiert ist, gespeichert. Die Androidanwendung soll dem Nutzer für gut durchgeführte Gesten positive Punkte vergeben. Für weniger gut oder schlecht durchgeführte Gesten soll der Nutzer weniger oder negative Punkte erhalten. Die Gesten sollen sich aus Liniensegmenten zusammensetzen. Je mehr Gesten der Nutzer durchgeführt hat, desto mehr Segmente können die Gesten besitzen. Dem Nutzer ist ein Zeitfenster in einem definierten Spielabschnitt gegeben, in dem er soviel Gesten wie möglich absolvieren muss. Die Punkte die der Nutzer durch das Ausführen von Gesten erspielt hat, addieren sich in Form von Millisekunden zu diesem Zeitfenster. Das Spiel soll dann beendet werden, wenn dem Nutzer keine Zeit mehr zur Verfügung steht. Neben den im Spiel gesammelten Daten soll die Gesamtpunktzahl in einen Highscore eingetragen werden. Damit die gesammelten Daten in die Datenbank übertragen werden können, ist eine Internetverbindung zwingend notwendig. Die Anwendung benötigt folgende Rechte auf dem Androidsystem:

- „android.permission.INTERNET“ - Wird benötigt, um die Daten in die Datenbank zu schreiben.
- „android.permission.ACCESS_NETWORK_STATE“ - Wird benötigt, um herauszufinden, ob momentan eine Internetverbinug besteht (da das Spiel eine Internetverbindung zwingend benötigt).

Um eine einfache Wartung der Androidanwendung zu gewährleisten, soll außerdem auf eine gut strukturierte und flexible Anwendungsarchitektur geachtet werden.

4.1.3 Spielverhalten

Das Android SDK stellt 3 Methoden zur Gestenabfrage bereit:

- Touch_Down - Stellt die X- und Y-Koordinaten des Anfangsberührungspunktes dar.

- Touch_Move - Stellt die X- und Y-Koordinaten beim Bewegen über dem Bildschirm dar.
- Touch_Up - Stellt die X- und Y-Koordinaten des Endberührungspunktes dar.

Gestenabgleich

Mit Hilfe dieser Abfragemöglichkeiten wird in der Methode „matchPoints“, die Liste der durch Toucheingabe generierten Koordinaten mit denen der vorgegebenen Geste verglichen. Die Liste der Touchkoordinaten muss allerdings vorher noch gekürzt werden, da die Liste der Gestenkoordinaten viel mehr Koordinatenpaare enthält, als die durch Toucheingaben generierte Liste. Das tritt auf, da die Touch_Move Funktion nicht alle Touchevents des Users abfängt. Deshalb werden nur Gestenkoordinaten verglichen, die in vordefinierten Intervallen auftreten. Beinhalteten beide Listen dann die gleiche Anzahl an Koordinaten, werden sie mit folgendem Algorithmus verglichen:

Listing 4.1 Algorithmus, der die Abweichung von der Eingabe zur vorgegebenen Geste berechnet

```
for (int i = 0; i < UserTouchCoordinates.size(); i++)
    // calculates the stepindex of the given GestureCoordinates because the
    // GestureCoordinates have much more coordinates than the UserTouchCoordinates
    indexLine = i * stepsSizeLine;
    // prevent overflow
    if (indexLine < GestureCoordinates.size())
        counter++;
        // calculates the delta between UserTouchCoordinates and GestureCoordinates
        deltaX += Absolute(UserTouchCoordinates.get(i).getX() -
            GestureCoordinates.get(indexLine).getX());
        deltaY += Absolute(UserTouchCoordinates.get(i).getY() -
            GestureCoordinates.get(indexLine).getY());
        // adds points to result when touch is near line, removes points
        // when touch is away from line
        if (deltaY / counter < maxDeltaY AND deltaX / counter < maxDeltaX)
            addition += 1;
        else
            addition -= 1;
    fi

fi
end for
// average delta of gesture
deltaX = deltaX / counter;
deltaY = deltaY / counter;
```

Der in Listing 4.1 dargestellte Algorithmus berechnet zuerst die Vergleichsintervalle mit dem die Gestenkoordinaten und die Touchkoordinaten verglichen werden. Bei dem Vergleich der Gestenpaare werden für die X- und Y- Koordinaten die Deltawerte berechnet. Die Deltawerte werden summiert und später durch die Anzahl der Vergleiche geteilt, so entstehen die

4 Implementierung der Systemkomponenten

durchschnittlichen Deltawert in die jeweilige Achsenrichtung. Für die Punkteberechnung wird die Variable „addition“ bei einem Deltawert der größer als 10 ist, um 1 erniedrigt und bei einem Deltawert der kleiner gleich 10 ist, um 1 erhöht.

Der Fehlerwert einer Gestenausführung wird folgendermaßen berechnet:

Listing 4.2 Fehlerwertberechnung in der Geste

```
// add average of deltaX and deltaY of a gestureInput and divide it by two
penalty = ((deltaX + deltaY) / 2);
```

Die in Listing 4.2 beschriebene Formel zur Fehlerwertberechnung, addiert die durchschnittlichen Deltawerte in X- und Y-Richtung und teilt diese durch zwei um so den Fehlerwert der ausgeführten Geste zu erhalten.

Die aus dem Algorithmus, in Listing 4.1, beschriebenen Variablen „deltaX“ und „deltaY“ beinhalten die durchschnittliche Abweichung der Toucheingabe gegenüber der vorgegebenen Geste in der X- bzw. Y-Richtung. Die resultierenden Punkte aus dieser Touch-Eingabe werden mit folgender Formel berechnet:

Listing 4.3 Algorithmus, der die Punkteverteilung, die aus der Toucheingabe resultiert, berechnet

```
// add average of deltaX and deltaY of a gestureInput and divide it by two
penalty = ((deltaX + deltaY) / 2);

// add addition to prevent short touch events by user
if (UserTouchCoordinates.size() < 5)
    addition = -LengthOfGesture - 50;
fi
// calculates the scorepoints of the touchinput
timeAddition = (LengthOfGesture + (addition) - (penalty * 3)) * multiplier;

// prevent negative score
if (score + timeAddition >= 0)
    score += timeAddition;
else
    score = 0;
fi
```

Der Algorithmus in Listing 4.3 zeigt die Berechnung der konkreten Punktezahl der ausgeführten Geste. Gesten gelten nur als gültig, wenn sie mindestens fünf Toucheingaben des Spielers haben, ansonsten wird die Variable „addition“ auf einen negativen Wert gesetzt. Ansonsten setzt sich die Punkteberechnung folgendermaßen zusammen: auf die Länge der Geste wird „addition“ addiert, dann wird der dreifache Fehlerwert abgezogen. Dieses Zwischenergebnis wird noch mit einem Multiplikator multipliziert. Diese Formel wurde häufig angepasst und verändert, um ein ausgeglichenes und faires Spielverhalten zu erzeugen. Ein negativer gesamt Punktwert ist nicht möglich, null ist der kleinstmögliche gesamte Punktwert. Für die Punkteberechnung werden die folgenden Werte benötigt:

- „maxLengthAddition“ - die Länge der Geste (maximal 200 Pixel)
- „addition“ - Punktevergabe bzw. Punkteabzug in Abhängigkeit der Touchentfernung zur Geste.
- „penalty“ - Die durchschnittliche Deltaabweichung beider Richtungen (wird mit einem konstanten Faktor multipliziert)
- „multiplier“ - Der multiplier ist eine ganzzahlige Zahl im Bereich 1-3, dieser wird bei positiven Punktzahlen erhöht und bei negativen Punktzahlen auf 1 zurückgesetzt.

4.1.4 Datenupload

Die Daten werden an einen Webserver geschickt, der dieser Daten dann in der Datenbank abspeichert. Das geschieht mit der „postData“ Methode, die eine beliebige Anzahl von POST-Variablen an eine URL senden kann. In diesem Fall werden PHP-Skripte aufgerufen, deren einzige Aufgabe es ist, die ankommenden POST-Variablen unmodifiziert in die Datenbank zu schreiben. Das Hochladen geschieht in der Androidanwendung, mithilfe eines „AsyncTask“. Dieser Thread läuft im Hintergrund, um den normalen Betriebsablauf der Anwendung nicht zu blockieren. Gerade bei langsamen Internetverbindungen könnte das Hochladen der Daten einige Sekunden dauern und so den Spielfluss enorm behindern.

4.1.5 Architektur

Die Androidanwendung besteht in der Version 1.1 aus folgenden Komponenten:

- Sechs Activities
 - „Hilfe“ - Hier wird das Spiel in wenigen Sätzen kurz erläutert.
 - „Highscore“ - Hier werden dem Spieler alle Nutzer mit Rang und Punktzahl angezeigt, der aktuelle Nutzer ist dabei gelb hinterlegt.
 - „Lizenzinformationen“ - Hier befinden sich die Lizenzinformationen zu den Audio-Stücken, die in dieser Anwendung verwendet werden.
 - „Spiel-Anwendung“ - Hier wird der Spieleinhalt geladen.
 - „Hauptmenü“ - Hier kann der Nutzer einen Menüpunkt auswählen.
 - „Umfrage“ - Hier kann der Nutzer eine kurze Umfrage zu seiner Person durchführen.
- Neun Klassen zur Datenhaltung
- Acht Klassen zur allgemeinen Verwendung wie zum Beispiel HTTP Verbindungen aufbauen, Threads, etc.

4.1.6 Design

Da die Anwendung als Spiel entwickelt wurde, wurde großer Wert auf die Außenwirkung gelegt. Die Arbeit von Shirazi et al. [SSHG⁺13] behandelt die Thematik der Layoutanalyse von Android-Anwendungen und hat aufschlussreiche Ideen und Anregungen geliefert, die das Layout der Activities angehen.

Dennoch wurde trotz der Tatsache, dass 88.25% der analysierten Anwendungen mehr als eine Sprache unterstützen, Englisch als einzige Sprache in „GestureGame“ verwendet. Dadurch werden rudimentäre Englischkenntnisse zum Lesen der Textausgabe vorausgesetzt. Zur Benutzung werden diese Sprachkenntnisse allerdings nicht benötigt.

Desweiteren wurden die Activities mit einem farbigen Hintergrund ausgestattet. Jede Stage hat einen eigenen farbigen Verlauf als Hintergrund, auch die aufsteigenden Punkte sowie die Zeitleiste sind farblich dargestellt um ein angenehmes Bild für den Nutzer zu erzeugen. Desweiteren wurde die Anwendung sehr schlicht gehalten. So gibt es nur die aktuelle Punkteanzahl, den aktuellen Fortschritt innerhalb der Stage und die aktuelle Stage selbst. Je weiter der Spieler innerhalb einer Stage gekommen ist, desto weiter schreitet der Aufbau des Hintergrundbildes voran.



Abbildung 4.2: Diese Abbildung zeigt das Hauptmenü (links), den Highscore vom 22.10.2013 (mitte) und die freiwillige Umfrage (rechts)

In Abbildung 4.2 werden das Hauptmenü, der Highscore und die Umfrage dargestellt.

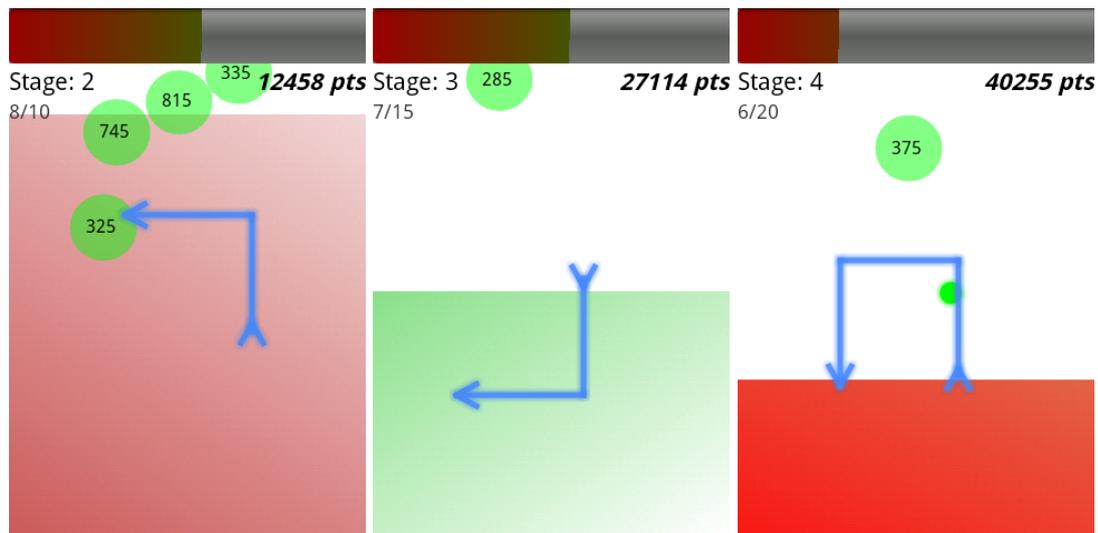


Abbildung 4.3: Diese Abbildung zeigt Ausschnitte aus der Anwendung während ein Spiel durchgeführt wird

In Abbildung 4.3 werden Momentaufnahmen aus dem Spiel, in verschiedenen Stages dargestellt. Am oberen Rand kann die Zeitleiste erkannt werden. In der linken oberen Ecke wird die aktuelle Stage sowie den aktuellen Stand in dieser Stage angezeigt. Die rechte Ecke zeigt die aktuelle Punktzahl an. In den Kreisen können die Punkten von bereits ausgeführten Gesten erkannt werden.

4.1.7 Gesammelte Daten

Die Androidanwendung sammelt bei der Erstausführung einmalig die Gerätedaten und nach jeder erfolgreich oder erfolglos beendeten Stage die bis dort durchgeführten Gesten. Gerätedaten sind folgende:

- GeräteID - Die bei Androidgeräten eindeutige GeräteID (wird in der Datenbank als Primärschlüssel verwendet)
- Gerätemodell - Der Name des Gerätemodell's
- Displaybreite - Die Breite des Displays in Pixeln
- Displayhöhe - Die Höhe des Displays in Pixeln
- DeviceBuild - Industrieller Name des Gerätes
- Produkt - Produktbezeichnung
- Hersteller - Den Namen des Herstellers des Gerätes
- Android API Version - Die Version des Androidbetriebssystems

4 Implementierung der Systemkomponenten

- Zeitzone - Die auf dem Gerät eingestellte Zeitzone
- Diagonale Displaygröße in Inch
- Diagonale Displaygröße in cm

Gespeicherte Daten in einer Stage sind folgende:

- GeräteID - Welches Gerät hat diese Stage gespielt
- Stage - Welche Stage wurde gespielt
- Startzeit - Wann wurde die Stage angefangen
- Endzeit - Wann wurde die Stage beendet
- Punkte - Wieviele Punkte wurden in dieser Stage erspielt
- Fortschritt - Wenn die Stage nicht beendet wurde. Wieviele Gesten wurden trotzdem durchgeführt

Zu einer vollendeten Geste werden folgende Daten gespeichert:

- GeräteID - Welches Gerät hat diese Geste durchgeführt
- Richtung - Aufzählung der Richtungen der Segmente (ausgehend vom Startpunkt)
- GestenStartX - X-Position an der die Geste ihren Ursprung hat
- GestenStartY - Y-Position an der die Geste ihren Ursprung hat
- GestenEndeX - X-Position an der die Geste ihr Ende hat
- GestenEndeY - Y-Position an der die Geste ihr Ende hat
- Gesten Fehler - eine Fehlerindexzahl dieser Geste
- Druck (Durchschnitt) - Den durchschnittlichen Druck, mit der diese Geste ausgeführt wurde
- Größe (Durchschnitt) - Die durchschnittliche Größe der Toucheingabe mit der diese Geste ausgeführt wurde
- Gestennummer - Eine fortlaufende Nummer der Geste in der aktuellen Stage

Zu einer einzelnen Geste werden folgende Daten gespeichert:

- GestenID - Die eindeutige ID einer vollendeten Geste
- Typ - Welcher Typ des Touchevents (Down,Move,UP) ausgeführt wurde.
- X - Momentane X-Position des Touchevents
- Y - Momentane Y-Position der Touchevents
- Druck - Momentaner Druck des Touchevents
- Größe - Momentane Größe des Touchevents

- Zeit - Momentaner Zeitstempel des Touchevents

Die Umfrage, die aus dem Hauptmenü aus erreichbar ist, ist eine freiwillige Möglichkeit anonyme Daten zu seiner Person abzugeben. Die gesammelten Daten sind folgende:

- Alter - Das Alter der Person
- Geschlecht - Das Geschlecht der Person
- Beruf - Der Beruf der Person
- Benutzung von Touchdisplays - Die Benutzungsdauer von Touchdisplays in Stunden pro Tag

4.1.8 Stages

Die Stages sind die Container für die einzelnen Gesten. Wird ein neues Spiel begonnen, so wird immer die erste Stage gestartet. Diese beinhaltet 5 Gesten. Die Gesten besitzen in der ersten Stage genau ein Liniensegment. Dieses Liniensegment kann in 4 verschiedenen Richtungen auftreten: Rechts, links, unten, oben. Diese Richtung sowie die Platzierung auf dem Bildschirm geschieht zufällig. Wird eine Stage erfolgreich abgeschlossen, so bekommt der Nutzer seine erzielten Punkte als Zeit gutgeschrieben. Desweiteren wird die Gestenanzahl in der darauf folgenden Stage um 5 erhöht. Die Auswahl an Gesten erhöht sich auch mit dem Ansteigen der Stage. In jeder neuen Stage bekommen die möglichen Gesten ein zusätzliches Segment. So können Gesten mit bis zu vier Segmenten entstehen.

4.1.9 Format

Die in der Anwendung zwischengespeicherten Daten werden dann nach jeder vollendeten oder nicht vollendeten Stage mittels der Klasse „http“ mit einem http-Post-Verfahren zu dem dafür vorgesehenen PHP Skript auf den Webserver geschickt. Dieses Skript trägt die Daten dann in die für diese Daten vorgesehene Tabelle ein. Werden Daten aus der Datenbank angefordert, wie es bei dem Highscore der Fall ist, generiert das PHP-Skript eine „XML-Zeichenkette“ mit allen benötigten Daten und sendet sie an die Androidanwendung, die diese Zeichenkette dann einliest und weiterverarbeitet. Die Anwendung kann dann die Daten auslesen und in die Highscoreliste eintragen.

4.1.10 Herausforderung

Die größte Schwierigkeit war, die Anwendung für die Spieler ansprechend zu gestalten und dennoch relevante Daten zu sammeln. Auch die Spielgeschwindigkeiten und Spielschwierigkeiten stellen ein Problem dar, da das Spielverhalten der Nutzer stark variiert.

4.2 Webserver

Der Webserver beinhaltet die PHP-Skripte die zum Hochladen der Daten aus der Androidanwendung benötigt werden. Auch beinhaltet der Webserver einen Qr-Code mit der aktuellsten Version der Androidanwendung zum Download. Ansonsten hat der Webserver keine Frontend-Komponenten und hat somit nur die Funktion, die Schnittstelle zur Datenbank herzustellen.

4.3 Datenbank

Die Datenbank wurde durch „PHPMyAdmin“ erstellt und gepflegt. Die Datenbank hat einen eigenen Benutzer „gesturegame“, der Vollzugriff zu dieser Datenbank hat.

4.3.1 Nichtfunktionale Anforderungen

Die Datenbank soll eine MySQL in der Version 5.1.49 sein. Die Daten werden über PHP-Skripte an die Datenbank übertragen, diese Skripte melden sich mit den Benutzer (gesturegame) an der Datenbank an.

4.3.2 Architektur

Die Datenbank beinhaltet 7 Tabellen:

- „debug“ - Beinhaltet diverse Debug-Zeilen, die nicht für die eigentliche Ausführung oder Auswertung benötigt werden.
- „devices“ - Enthält alle Daten eines mobilen Endgerätes, welches die Anwendung gestartet hat. Primarykey ist die GeräteID.
- „highscores“ - Enthält die kompletten Highscoredaten aller DeviceID's die jemals mindestens eine Stage gespielt haben.
- „stages“ - Enthält Informationen zu allen gespielten Stages von jedem Device.
- „survey“ - Enthält die Daten von der Umfrage jedes „devices“.
- „touchdata“ - Enthält die zu einer Geste gehörenden Daten.
- „touchdata_detailed“ - Enthält detailliertere Eingabedaten zu einer Geste.

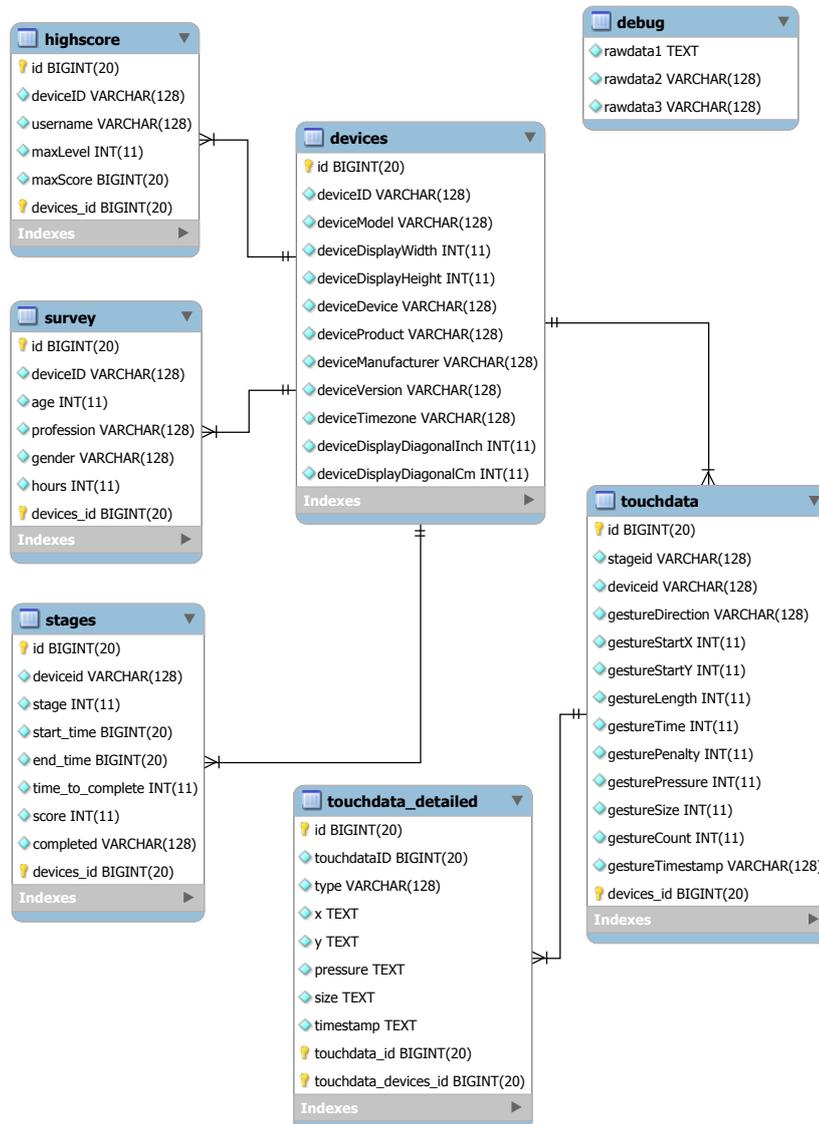


Abbildung 4.4: Diese Abbildung zeigt das Entity-Relations-Model der Datenbank.

In Abbildung 4.4 kann der genaue Aufbau der Datenbank erkannt werden. Die Datenbank besitzt insgesamt 55 Spalten in 7 verschiedenen Tabellen.

5 Studie

In diesem Kapitel werden die Techniken, mit denen die Androidanwendung verbreitet wurde, näher betrachtet. Desweiteren wird der Verlauf der Benutzerstudien, die mittels der Androidanwendung durchgeführt wurden, erläutert.

Zum einen wurde eine Benutzerstudie mit Probanden durchgeführt, die diese Anwendung aus dem Google Play Store bezogen haben. Zum anderen wurde noch eine Laborstudie durchgeführt. Die Probanden die in dieser Studie teilgenommen haben, wurden aus dem Bekanntenkreis ausgewählt.

5.1 Testphase

Während der gesamten Testphase wurde die Anwendung mehrmals auf den Testgeräten ausgeführt, um die Benutzbarkeit der Anwendung in Abhängigkeit des Nutzens der Datenerhebung zu optimieren. Auch im näheren Bekanntenkreis wurde die Anwendung mehreren Probanden zum Testen bereitgestellt und Rückmeldungen bezüglich der Benutzbarkeit umgesetzt. Nachdem die interne Testphase abgeschlossen war, wurde die Datenbank komplett geleert und alle bisherigen Probanden wurden aufgefordert, die Anwendung wiederum mehrmals auszuführen, um ihre Daten erneut zu erfassen.

5.2 Google Play Store Benutzerstudie

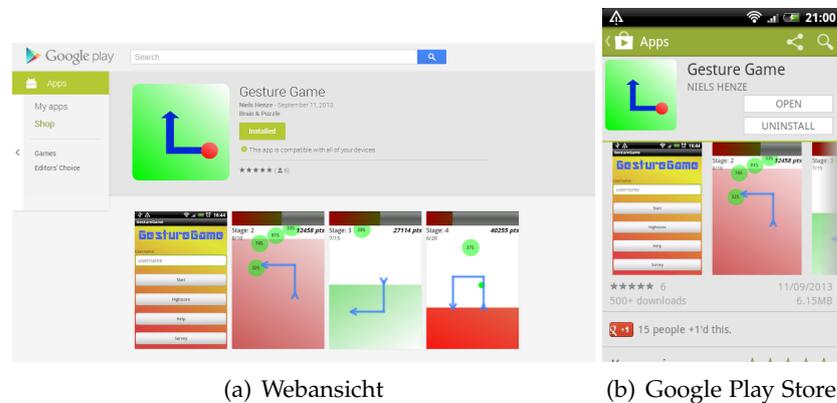


Abbildung 5.1: Die Google Play Store Webseite, von dem die Anwendung heruntergeladen werden kann (a) und die Google Play Store Application von mobilen Endgeräten (b)

Die Anwendung wurde seit der Veröffentlichung am 22.09.2013 bis zum 22.10.2013 575 mal auf Androidgeräten installiert. Wie in Abbildung 5.1 zu sehen ist, fand die Veröffentlichung durch den Google Play Store statt. Von diesen 575 installierten Anwendungen sind am 22.10.2013 noch 150 installierte Anwendungen auf Androidgeräten vorhanden. Die Anwendung wurde insgesamt 288 mal von verschiedenen Geräten aus gestartet. Von diesen 288 Installationen wurde von 286 Geräten mindestens ein Spiel durchgeführt. Von diesen 268 Durchführungen können 180 Datensätze verwendet werden, da bei den restlichen Datensätzen die Datenmenge zu gering ist. Insgesamt wurden 32.669 Gesten durchgeführt und 3.393 Stages gespielt. Verwendet wurden dafür 154 unterschiedliche Gerätemodelle, diese Geräte verfügen über 29 unterschiedliche Auflösungen. Dabei sind 240x320 Pixel die kleinste und 1080x1920 Pixel die größte Auflösung. Mit 86 Geräten war ein Displayauflösung von 480x800 Pixeln die häufigste verwendete Auflösung. Die drei am häufigsten verwendeten Geräte waren: Samsung Galaxy S III (21 mal vorhanden), Samsung Galaxy Y (14 mal vorhanden) und das Samsung Galaxy S II (12 mal vorhanden). Die in der Anwendung integrierte Umfrage wurde von 114 Personen durchgeführt, wobei 21 mit korrekten Daten ausgefüllt wurden. Der jüngste Proband war 17 Jahre und der älteste Proband 43 Jahre alt war. Das Durchschnittsalter beträgt 29,3 Jahre mit einer Standardabweichung von 8,2. An der Umfrage haben 6 Frauen und 15 Männer teilgenommen.

Bei der Betrachtung, aus welchen Zeitzonen die Probanden kommen, die die Anwendung installiert haben, kann folgendes festgestellt werden: Die 3 häufigsten Zeitzonen sind Asien mit 131 Installationen, Afrika mit 66 Installationen und Europa mit 56 Installationen. Die drei am häufigsten verwendeten Androidversionen auf den Geräten waren: Android 4.1 mit 176 Installationen, Android 2.3.3-2.3.7 mit 175 Installationen und Android 4.0.3-4.0.4 mit 130 Installationen.

5.3 Laborstudie

Die Anwendung wurde ein weiteres Mal in einem ausgewählten Benutzerkreis zur Anwendung freigegeben. Dabei haben alle Probanden dasselbe Gerät benutzt. Das Testgerät, auf dem diese Studie durchgeführt wurde, war ein Samsung Galaxy S2 (GT-I9100). Dieses Gerät hat eine Bildschirmauflösung von 480x800 Pixeln und hat Android 4.1 (API 16) installiert. Insgesamt haben 11 Probanden aus dem Bekanntenkreis und näherem Umfeld an der Studie mitgewirkt. Dadurch sind 2.225 Gesten ausgeführt worden. Von den 11 Probanden waren 8 männlich und 3 weiblich. Der älteste Proband war 29 und der jüngste 21. Das Durchschnittsalter beträgt 24,4 Jahre mit einer Standardabweichung von 2,4.

6 Auswertung

In diesem Kapitel werden die Ergebnisse der Daten, die in den Benutzerstudien gesammelt wurden, ausgewertet, dargestellt und näher erläutert. Auf Grund der hohen Streuung der Daten wird eine Visualisierung der Daten durch Boxplots durchgeführt. Zum Vergleich der Ausführungszeit wird der Median, der jeweiligen Auswertung betrachtet. Der berechnete Fehlerwert, siehe Listing 4.2, wird ebenfalls mit den Medianwerten verglichen.

6.1 Google Play Store Benutzerstudie

Die vorhandenen Daten wurden im Zeitraum vom 22.09.2013 bis zum 22.10.2013 gesammelt und in der Datenbank gespeichert. Die Dauer der Studie beläuft sich somit auf vier Wochen. Von den insgesamt 32.669 ausgeführten Gesten können 29.337 Gesten für die Auswertung verwendet werden. Die nicht verwendeten Daten sind sowohl Daten von Nutzern, die weniger als 15 Gesten ausgeführt haben als auch Daten von denjenigen Nutzern, die die erste Stage nicht absolviert haben. Die erste Stage wurde insgesamt 1.726 mal von 286 Spielern ausgeführt, aber nur 544 mal von 107 verschiedenen Spielern beendet.

Die Auswertung des Fragebogens hat ergeben, dass von den 21 Befragten 33% der Studenten sind und 38% einer Arbeit nachgehen. Die Restlichen 29% können, auf Grund von unzureichenden Angaben, nicht eindeutig zugeordnet werden. Die durchschnittliche Nutzungsdauer eines Touchdisplays pro Tag beträgt 2,0 Stunden mit einer Standardabweichung von 2,4.

Im Folgenden wird aufgezeigt, wieviele Spieler die jeweiligen Stages absolviert haben.

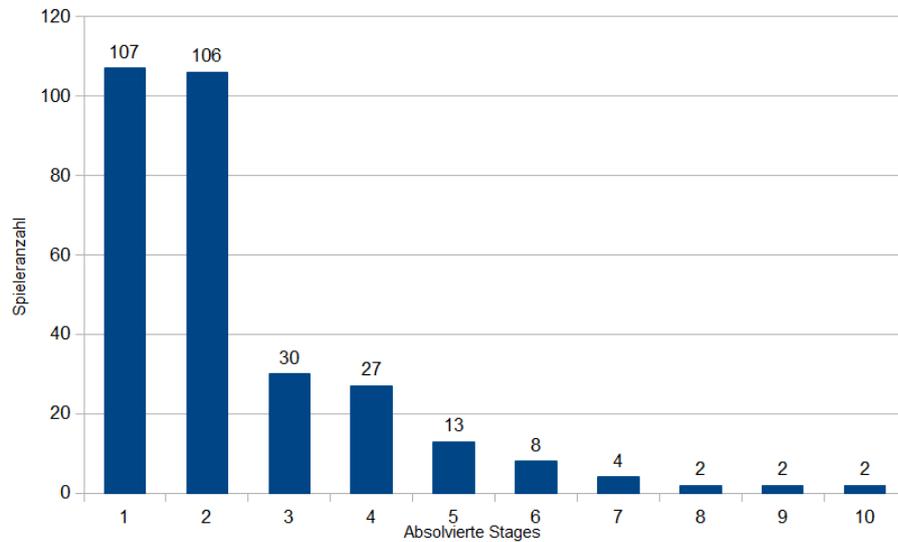


Abbildung 6.1: Diese Abbildung zeigt die Anzahl der Spieler, die die jeweilige Stage erfolgreich absolviert haben

In Abbildung 6.1 wird aufgezeigt, dass 107 Spieler die erste Stage erfolgreich abgeschlossen haben. Die zweite Stage wurde von 106 Spielern erfolgreich abgeschlossen. Ab Stage drei kann ein deutliches Abfallen der Spielerzahl erkannt werden. Nur noch 30 Spieler konnten die Stage drei erfolgreich abschließen. Ein weiterer starker Einbruch von der Spieleranzahl kann in Stufe fünf erkannt werden. Dort konnten nur noch 13 Spieler die Stage erfolgreich abschließen. Ab Stufe acht konnten sich die zwei besten Spieler bis Stufe 10 behaupten. Abbildung 6.1 zeigt zwei große Anstiege in der Spielschwierigkeit in Stage drei und in Stage fünf.

Nachfolgendes Diagramm veranschaulicht, wie oft ein Spiel von einem Spieler gestartet wurde. Dabei werden die kompletten Daten aller Spieler angeschaut, auch von den Spielern, die die erste Stage nicht erfolgreich beenden konnten.

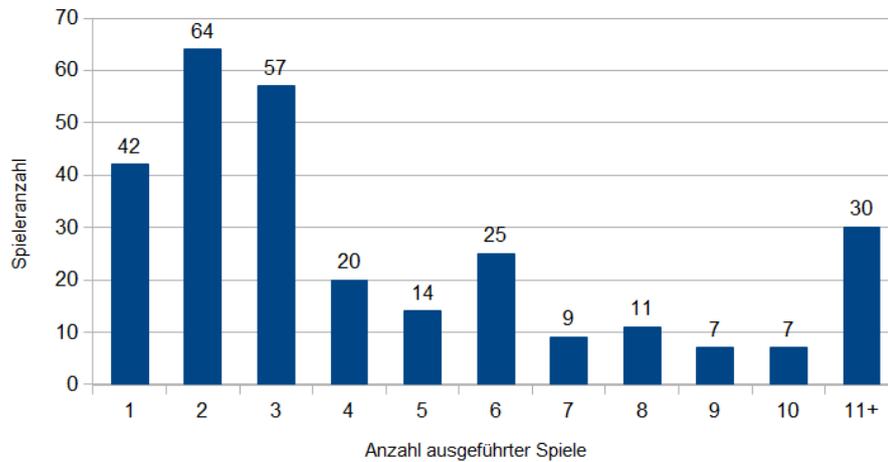


Abbildung 6.2: Diese Abbildung zeigt, wieviele Spieler eine bestimmte Anzahl an Spielen durchgeführt haben

Abbildung 6.2 zeigt, dass 163 Spieler weniger als drei Spiele gespielt haben und die Anwendung danach nicht mehr genutzt haben. Dennoch haben 30 Spieler mehr als 11 Spiele durchgeführt. Wie in Abbildung 6.1 zu sehen ist, haben genau diese 30 Spieler die dritte Stage bestanden.

Im Folgenden werden alle Gestenausführungen von Gesten mit einem Segment betrachtet. Dabei werden nur die Gesten berücksichtigt, die von den erwähnten 107 Spielern ausgeführt wurden. Insgesamt haben diese Spieler 16.720 Gesten mit einem Segment ausgeführt. Die Gesten werden in vier Richtungen eingeteilt und in der Abhängigkeit von ihrer Ausführungszeit und ihrem Fehlerwert betrachtet.

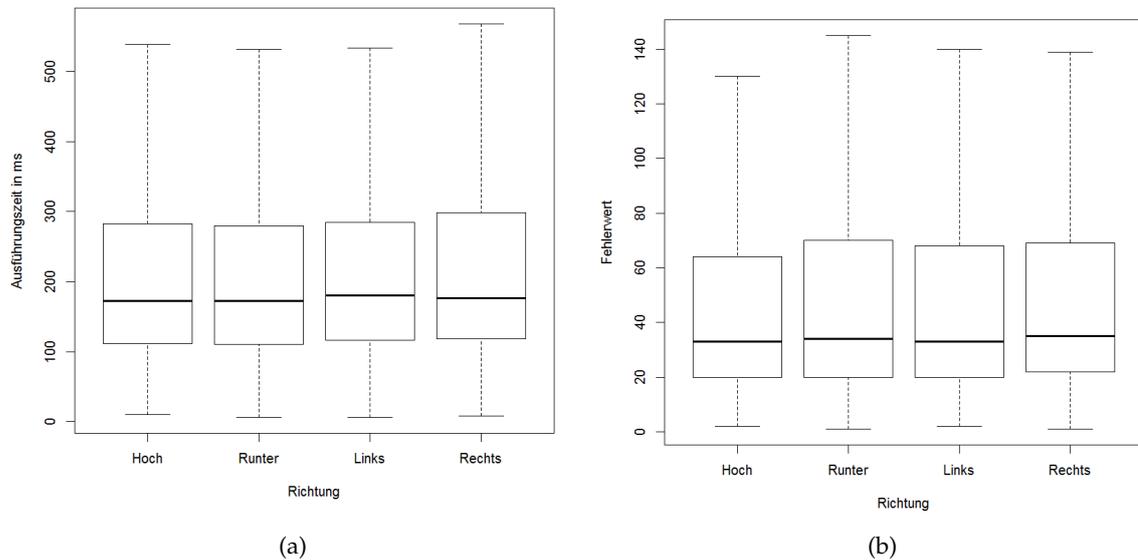


Abbildung 6.3: Diese Abbildungen zeigen die benötigten Ausführungszeiten (a) und die Fehlerwerte (b) für Gesten mit einem Segment in Abhängigkeit der Ausführungsrichtung

Bei der genaueren Betrachtung der Gestenausführungen in Abbildung 6.3, kann keine große Differenz zwischen den einzelnen Richtungen erkannt werden. Gesten in die linke Richtung haben mit einem Median von 180 ms die längste Ausführungszeit. Gesten die in die rechte Richtung zeigen haben einen Median der Ausführungszeit von 176 ms. Gesten, die nach unten zeigen, benötigen 172 ms im Median für ihre Durchführung und Gesten die nach oben ausgeführt werden müssen, haben auch einen Median der Ausführungszeit von 172 ms. Da die Ausführungszeiten in alle vier Richtungen der Gesten weniger als 10 ms auseinander liegen, kann davon ausgegangen werden, dass die Richtung nicht entscheidend für die Ausführungsgeschwindigkeit der Geste ist.

Bei der Betrachtung der Fehlerwerte wird deutlich, dass auch diese Werte dicht beieinander liegen. So haben Gesten, die nach oben oder in die linke Richtung gehen, einen Median des Fehlerwertes von 33. Gesten, die nach rechts ausgeführt werden müssen, haben einen Median des Fehlerwertes von 35 und Gesten die nach unten ausgeführt werden müssen, haben einen Median des Fehlerwertes von 34. Auch bei der Betrachtung des Fehlerwertes wird deutlich, dass die Richtung nicht ausschlaggebend für die Fehlerwerte ist.

Folgende Abbildung zeigt die Ausführungszeit und die Fehlerwerte von Gesten mit zwei Segmenten, wobei hier die Startrichtung der Geste genauer betrachtet wird. Die Startrichtung wird in horizontal und vertikal eingeteilt. Insgesamt wurden 7.756 Daten betrachtet.

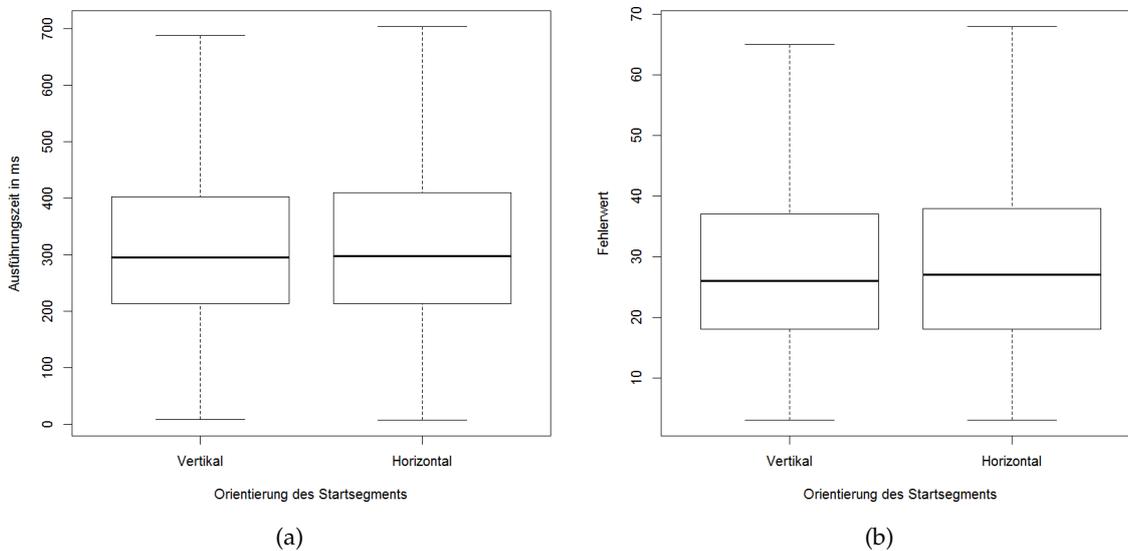


Abbildung 6.4: Diese Abbildungen zeigen die benötigte Ausführungszeit (a) und Fehlerwerte (b) von Gesten mit zwei Segmenten, wobei das Startsegment eine vertikale oder horizontale Ausrichtung hat.

Abbildung 6.4 stellt die Ausführungszeiten und Fehlerwerte von Gesten mit zwei Segmenten dar. Der Median der Ausführungszeit von Gesten, bei denen sich das erste Segment in horizontaler Lage befindet, beträgt 297 ms. Gesten mit einem vertikal liegenden Startsegment besitzen einen Median der Ausführungszeit von 295 ms. Die Ausführungszeiten liegen somit sehr dicht beieinander.

Die Ausrichtung des ersten Segmentes hat keinen großen Einfluss auf die Ausführungszeit. Bei den Fehlerwerten haben Gesten mit einer horizontalen Startgeste einen Median des Fehlerwertes von 27 und vertikale Startgesten besitzen einen Median des Fehlerwertes von 26. Somit liegen auch die Fehlerwerte sehr dicht beieinander. Die Ausrichtung der Startgeste hat auch keinen Einfluss auf die Fehlerwerte der Geste.

Im Folgenden werden alle Gesten die von sämtlichen Nutzern gemacht wurden, die mindestens die zweite Stage erreicht haben. Es werden die Ausführungszeit und die Fehlerwerte in Abhängigkeit von der Segmentanzahl und die damit verbundene Länge der Geste betrachtet.

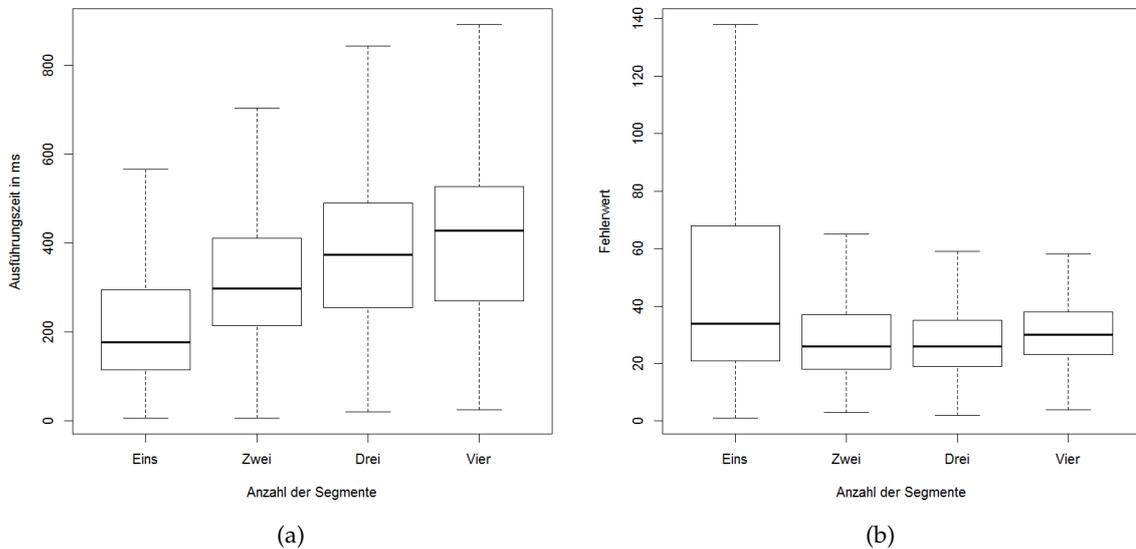


Abbildung 6.5: Diese Abbildungen zeigen die benötigte Ausführungszeit (a) und Fehlerwert (b) von Gesten mit entsprechender Segmentenanzahl an

Abbildung 6.5 veranschaulicht die Ausführungszeit und Fehlerwerte von allen betrachteten Gesten. So beträgt der Median der Ausführungszeit einer Geste mit einem Segment 177 ms. Werden nun die Gesten mit zwei Segmenten betrachtet, beträgt der Median der Ausführungszeit 297 ms. Das bedeutet, dass sich die Ausführungszeit um den Faktor 1,7 vergrößert hat, während sich die Länge der Geste verdoppelt hat.

Bei Gesten mit drei Segmenten beträgt der Median der Ausführungszeit 373 ms. Das entspricht einer Verlängerung der Ausführungszeit um den Faktor 2,1, bei einer Vergrößerung der Länge um das Dreifache.

Gesten mit vier Segmenten benötigen im Median 427 ms für die Ausführung. Das entspricht einer Verlängerung der Ausführungszeit um den Faktor 2,4. Das bedeutet, dass eine Verdoppelung der Ausführungszeit bei einer Verdreifachung der Gestenlänge erreicht wurde.

Die Mediane der Fehlerwerte nehmen nicht mit ansteigender Segmentanzahl zu, sondern liegen im Bereich von 26-34.

Die nächste Abbildung stellt wieder die Fehlerwerte und die Ausführungszeit der Gesten, diesmal in Abhängigkeit von der Benutzungsdauer von Touchdisplays einer Person dar. Dabei werden alle Gestenausführungen von Gesten mit einem Segment betrachtet, die von Spielern ausgeführt wurden, die an der freiwilligen Umfrage teilgenommen haben.

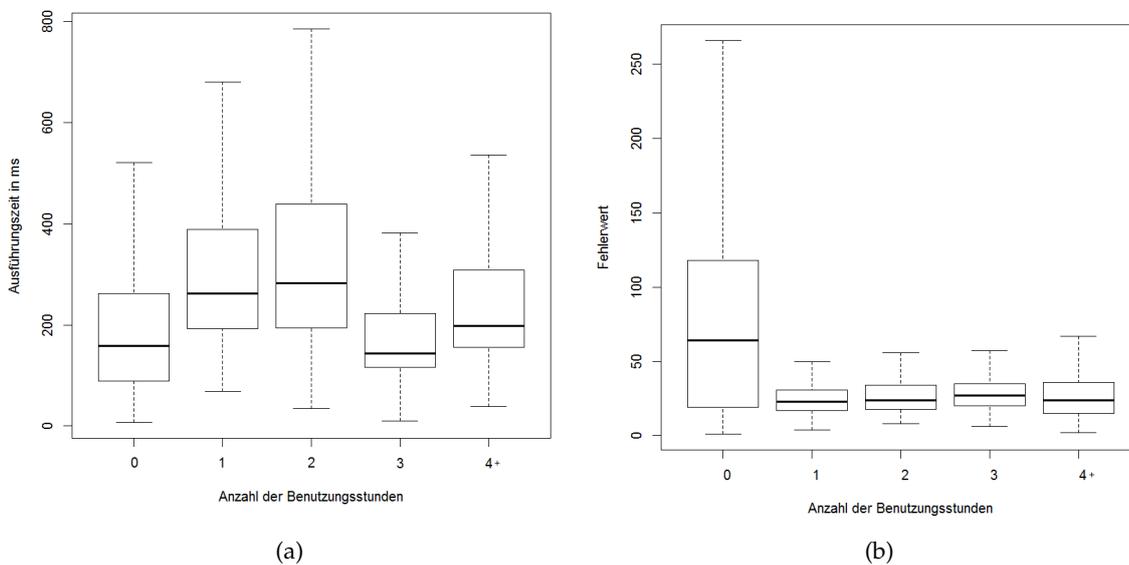


Abbildung 6.6: Diese Abbildungen zeigen die Fehlerwerte und Ausführungszeiten in Abhängigkeit von ihrer Nutzung eines Touchdisplays, gemessen in Stunden Pro Tag.

Abbildung 6.6 zeigt die Ausführungszeiten und Fehlerwerte in Abhängigkeit von den täglichen Nutzungsstunden des Touchdisplays. Dabei muss beachtet werden, dass dem Spieler völlige Freiheit bei seiner Einschätzung der Nutzungsstundenzahl gelassen wurde. Somit beträgt der Median der Ausführungszeit von Gesten mit einem Segment 159 ms bei Spielern, die angegeben haben, dass sie weniger als eine Stunde pro Tag ein Touchdisplay benutzen. Erhöhte Mediane der Ausführungszeiten fallen bei einer und zwei Stunden Benutzungsdauer auf. Mit 282 ms benötigen Spieler mit drei Stunden Nutzungsdauer die längste Zeit für eine Ausführung. Bei den Medianen der Fehlerwerte fällt der erhöhte Wert von 64 bei Spielern mit weniger als einer Stunde Nutzung auf. Die restlichen Mediane der Fehlerwerte liegen zwischen 23 bei einer Stunde Nutzung und 27 bei zwei Stunden Nutzung.

Folgende Abbildung wertet die Fehlerwerte und Ausführungszeiten aller Gesten mit einem Segment aus. Dabei wird von jedem Nutzer die erste und letzte Geste, die er während des gesamten Spielverlaufs ausgeführt hat, betrachtet.

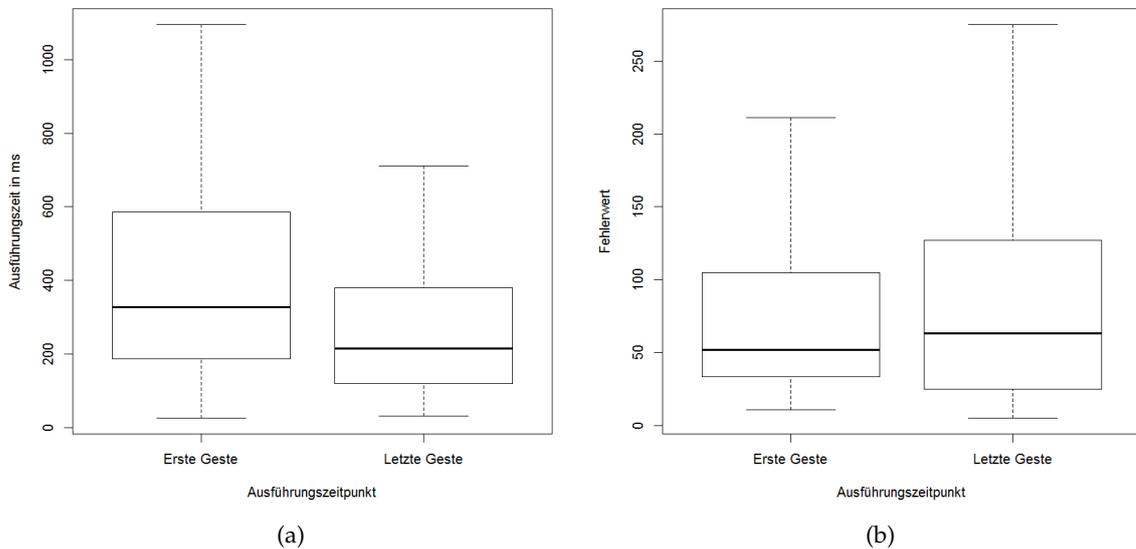


Abbildung 6.7: Diese Abbildungen zeigt die Ausführung der ersten und der letzten Geste mit nur einem Segment

Die Abbildung 6.7 zeigt auf, dass zwischen der ersten und letzten Geste eines Spielers eine Verbesserung in der Ausführungszeit beobachtet werden kann. Der Median der Ausführungszeit der ersten Geste mit einem Segment, liegt bei 328 ms. Die letzte Geste mit einem Segment, die der Spieler durchgeführt hat, wurde mit 215 ms im Median durchgeführt. Somit konnten sich die Spieler im Laufe ihrer Spielzeit um 113 ms bei der Ausführung von Gesten mit einem Segment verbessern. Bei den Fehlerwerten konnte allerdings keine Verbesserung festgestellt werden. Die Mediane der Fehlerwerte der letzten Geste liegen mit einem Wert von 63 über dem Fehlerwert der ersten Geste.

Im Folgenden wird der Fehlerwert in Abhängigkeit von dem ausgeübten Druck und der Größe der Toucheingabe betrachtet. Die Betrachtung beschränkt sich auf Gesten mit einem Liniensegment.

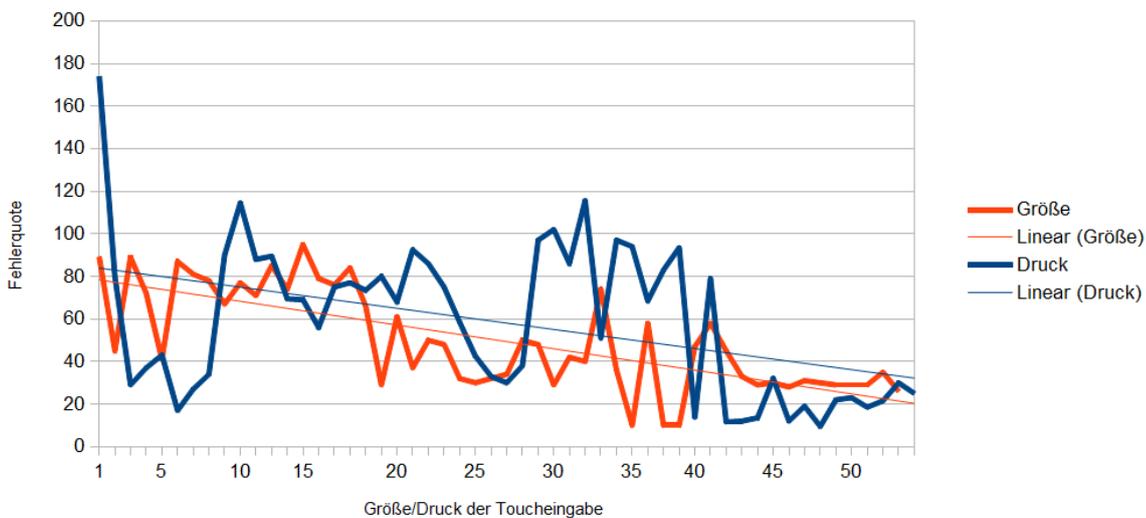


Abbildung 6.8: Diese Abbildungen zeigen den Fehlerwert bei einem bestimmten Druck und Größe einer Toucheingabe, sowie die Trendlinie von beiden Datensätzen, gemessen anhand der Ausführung von Gesten mit einem Segment

Abbildung 6.8 zeigt die durchschnittliche Größe und Druck der Toucheingaben. Die ausgelesene Größe ist von der Android API folgendermaßen definiert: Die Größe ist ein skaliertes Wert von den Druckpunkten auf der gedrückten Fläche des Displays. Dieser Wert kann benutzt werden um „fat touch“- Events zu erkennen. Der Druck liefert, genauso wie die Größe, einen Wert, der die Druckintensität der Toucheingabe wiedergibt. In diesem Schaubild wird verdeutlicht, dass mit zunehmender Größe und Druck der Fehlerwert sinkt. Beide Trendlinien haben eine negative Steigung, somit ist die Verbesserung der Fehlerrate offensichtlich. Druck und Größe der Toucheingabe haben einen relativ ähnlichen Verlauf, da sich mit erhöhtem Druck auf das Display automatisch die aufgelegte Fläche vergrößert.

6.2 Labor Benutzerstudie

Die Laborstudie wurde unter gleichbleibenden Bedingungen durchgeführt, um so eine geräteunabhängige Auswertung zu bekommen.

Alle Probanden haben die Anwendung auf einem Samsung Galaxy S 2 „GT-I9100“ ausgeführt, da die Auflösung dieses Gerätes von 480x800 am häufigsten in der Google Play Store Studie vertreten war.

In dieser Studie wurden 2.225 Gesten ausgeführt. Von diesen Gesten konnten 2.205 in dieser Auswertung betrachtet werden. Die restlichen 20 Datensätze hatten ungültige Werte in der Datenbank. Da alle 11 Probanden die erste Stage bestanden haben, werden die Daten von allen Probanden ausgewertet.

Die Auswertung des Fragebogens hat ergeben, dass von den 11 Befragten 63% der Studenten

sind und 37% einer Arbeit nachgehen. Die durchschnittliche Benutzungsdauer eines Touchdisplays pro Tag beträgt 1,7 Stunden mit einer Standardabweichung von 1,4.

Wie auch bei der Google Play Store Benutzerstudie wird in folgender Abbildung die Gestenausführungszeit und Fehlerrate von Gesten mit einem Segment in Abhängigkeit von ihrer Richtung betrachtet.

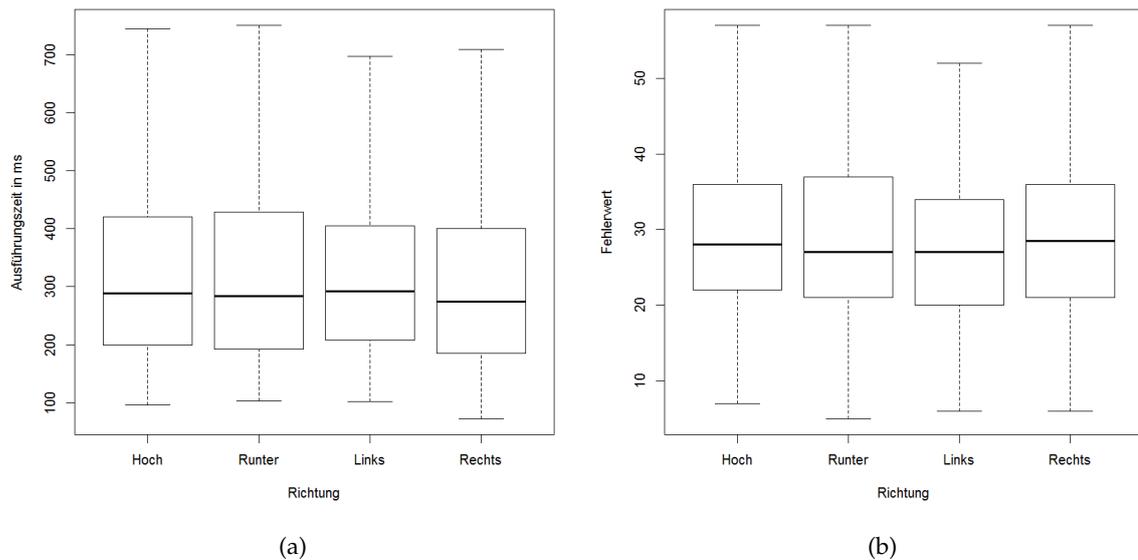


Abbildung 6.9: Diese Abbildungen zeigt die Ausföhrungszeit (a) und Fehlerwerte (b) von Gesten mit einem Segment in Abhängigkeit von ihrer Richtung

Abbildung 6.9 zeigt die Ausföhrungszeiten und Fehlerwerte von Gesten mit einem Segment in Abhängigkeit von der Richtung der Geste. Dabei haben Gesten, die nach unten zeigen, einen Median der Ausföhrungszeit von 283 ms. Gesten, die nach rechts gerichtet sind, benötigen im Median 274 ms für die Ausföhrung. Gesten die nach links zeigen, benötigen mit 292 ms die längste Ausföhrungszeit. Schließlich benötigen Gesten die nach oben zeigen 289 ms im Median zur Ausföhrung.

Die Ausföhrungszeiten aller vier Richtungen liegt dicht beieinander. So kann davon ausgegangen werden, dass auch in dieser Studie die Richtung der Geste nicht entscheidend für die Ausföhrungszeit ist. Bei den Fehlerwerten haben Gesten nach unten und links einen Median des Fehlerwertes von 27. Gesten die nach oben und rechts verlaufen haben einen Median des Fehlerwertes von 28. Auch bei den Fehlerwerten fallen keine großen Unterschiede auf.

Folgende Abbildung zeigt die Ausföhrungszeit und die Fehlerwerte von Gesten mit zwei Segmenten, wobei hier die Startrichtung der Geste genauer betrachtet wird:

Die Startrichtung wird in „horizontal“ und „vertikal“ eingeteilt.

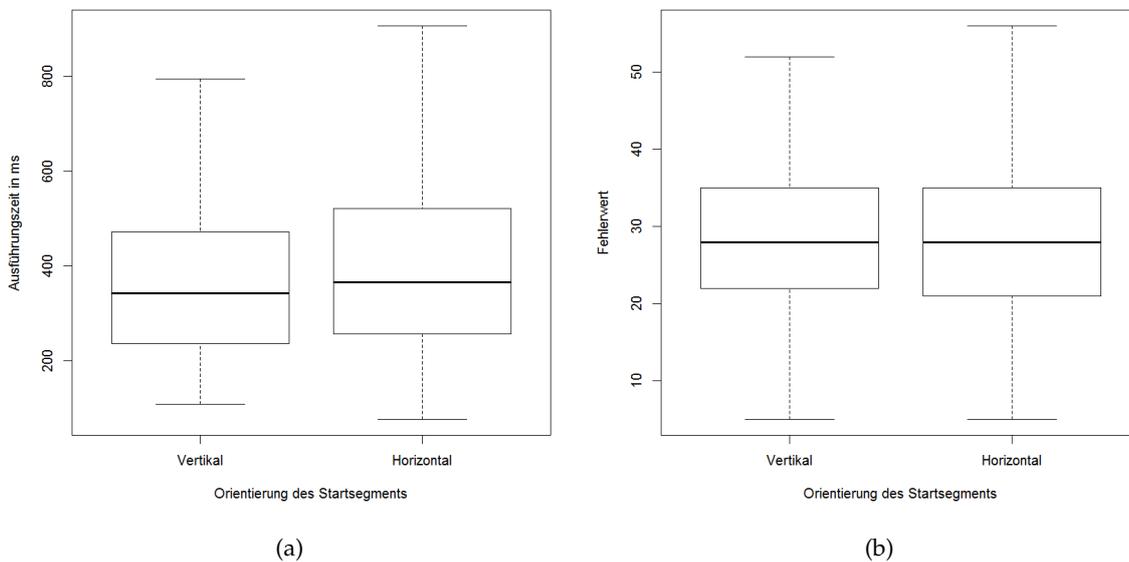


Abbildung 6.10: Diese Abbildungen zeigen die benötigte Ausführungszeit (a) und Fehlerwert (b) von Gesten mit zwei Segmenten, wobei das Startsegment eine vertikale oder horizontale Orientierung hat

Die Abbildung 6.10 stellt die Ausführungszeit und Fehlerwerte von Gesten mit zwei Segmenten dar. Der Median der Ausführungszeit von Gesten, bei denen sich das erste Segment in horizontaler Lage befindet, beträgt 366 ms. Gesten mit einem vertikal liegendem Startsegment besitzen einen Median der Ausführungszeit von 343 ms.

Die Ausführungszeiten liegen somit sehr dicht beieinander. Die Ausrichtung des ersten Segmentes hat keinen großen Einfluss auf die Ausführungszeit. Bei den Fehlerwerten haben Gesten mit einer horizontalen oder vertikalen Startgeste einen Median im Wert von 28. Somit liegen auch die Fehlerwerte sehr dicht beieinander. Die Ausrichtung der Startgeste hat auch keinen Einfluss auf die Fehlerwerte der Geste.

Im Folgenden werden die Gestenausführungszeit und die Fehlerwerte von allen ausgeführten Gesten betrachtet

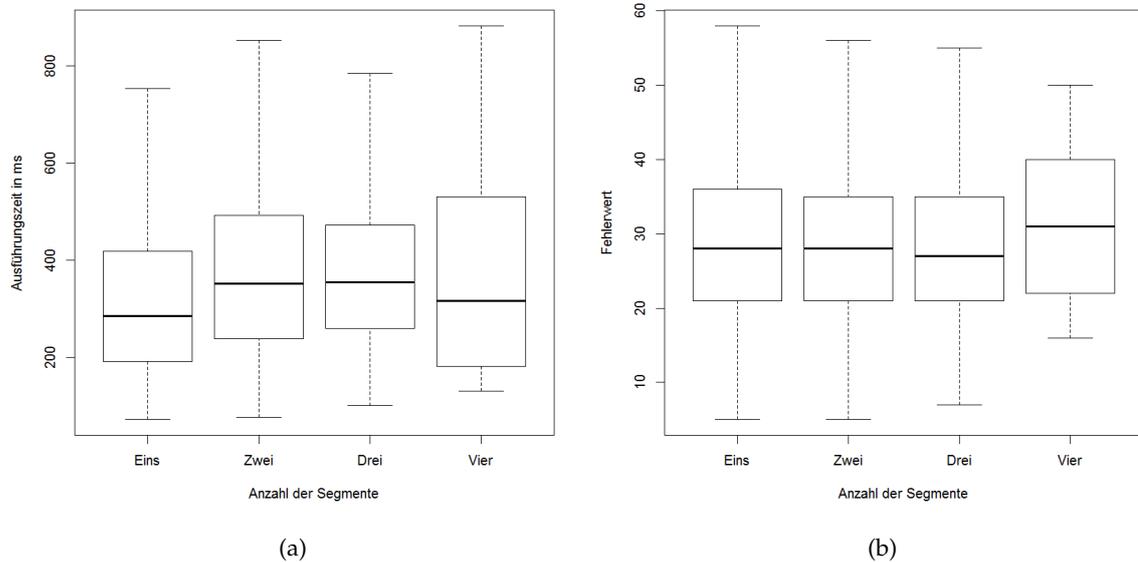


Abbildung 6.11: Diese Abbildungen zeigt die Ausführungszeit (a) und den Fehlerwert (b) von allen durchgeführten Gesten aller Spieler

Wie in Abbildung 6.11 zu sehen ist, wird ein Median der Ausführungszeit von 285 ms für Gesten mit einem Segment benötigt. Gesten mit zwei Segmenten benötigen einen Median der Ausführungszeit von 351 ms. Gesten mit drei Segmenten benötigen 355 ms. Gesten mit vier Segmenten benötigen sogar nur 316 ms im Median für ihre Ausführung.

Diese Werte haben aus folgenden Gründen so große Abweichungen von der Google Play Store Benutzerstudie: Zum einen kann diese Studie auf einen kleineren Datensatz mit 2.205 Daten zurückgreifen und zum Anderen wurden die Gesten mit drei und vier Segmenten nicht von allen Benutzern ausgeführt.

Nur sieben Spieler konnten mehr als fünf Gesten mit drei Segmenten ausführen. Bei Gesten mit vier Segmenten können nur noch Daten von vier Spielern betrachtet werden. Da Spieler, die höhere Stages erreichen, automatisch eine bessere Spielstatistik haben, ist es möglich, dass diese Spieler für die Ausführung einer Geste mit vier Segmenten weniger Zeit benötigen als schlechtere Spieler.

Aus diesem Grund werden in folgender Abbildung die Ausführungszeiten der drei besten Spieler betrachtet. Die drei besten Spieler haben insgesamt 912 Gesten ausgeführt.

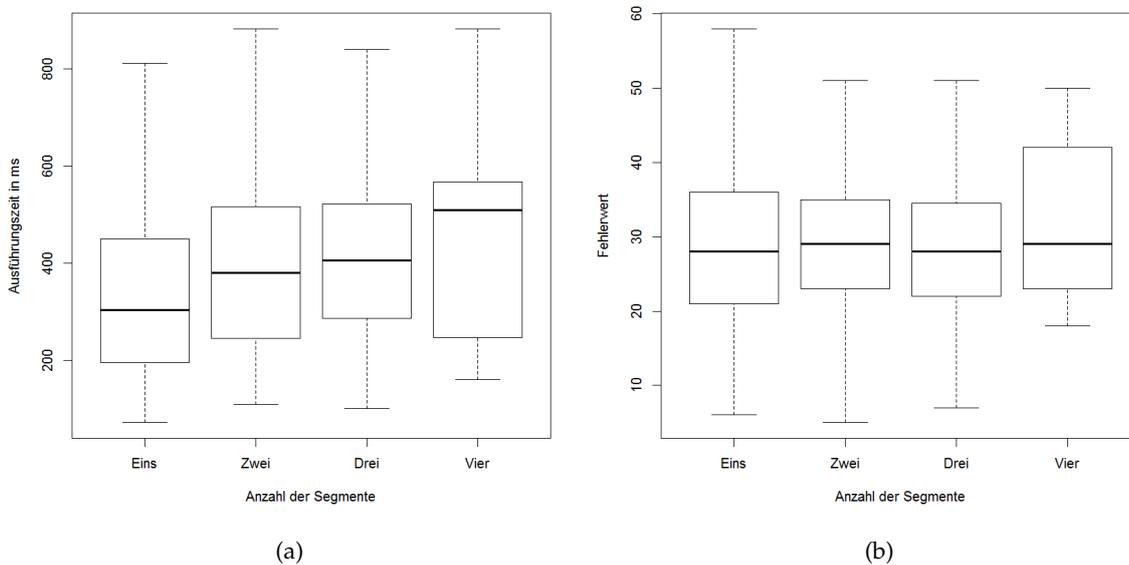


Abbildung 6.12: Diese Abbildungen zeigt die Ausführungszeit (a) und den Fehlerwert (b) von allen Gesten die von den drei besten Spielern ausgeführt wurden

In Abbildung 6.12 wird aufgezeigt, dass sich hier die Ausführungszeiten in Abhängigkeit zu der steigenden Gesteanzahl verlängert.

So haben Gesten mit einem Segment einen Median der Ausführungszeit von 304 ms. Gesten mit zwei Segmenten benötigen 380 ms, das ist eine Steigerung um den Faktor 1,25. Bei Gesten mit drei Segmenten beträgt der Median der Ausführungszeit 405 ms, das entspricht einer Verlängerung um den Faktor 1,33. Bei einer vierfachen Länge liegt die Steigerung der Ausführungszeit bei einem Faktor von 1,67. Dies ist im Gegensatz zu den aus der Google Play Store erhobenen Daten (Faktor 2,4) zwar ein geringerer Wert, aber die Tendenz ist dieselbe. Eine Verdoppelung der Länge einer Geste hat nicht zur Folge, dass sich auch die Ausführungszeit verdoppelt.

Die nächste Abbildung stellt wieder die Fehlerwerte und die Ausführungszeit der Gesten, in Abhängigkeit der Nutzungsdauer von Touchdisplays einer Person dar. Dabei werden alle Gestenausführungen von Gesten mit einem Segment betrachtet, die von allen Spielern ausgeführt wurden.

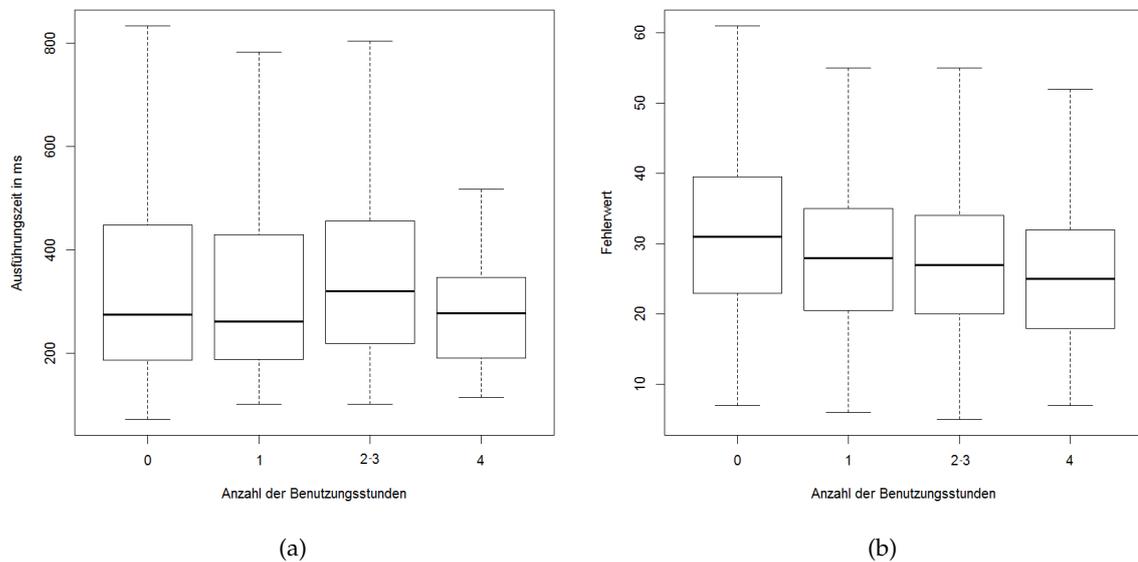


Abbildung 6.13: Diese Abbildungen zeigt die Ausführungszeit (a) und die Fehlerwerte (b) von allen Gesten mit einem Segment

In Abbildung 6.13 wird dargestellt, dass Spieler die weniger als eine Stunde täglich ein Touchdisplay benutzen für eine Geste mit einem Segment 274 ms im Median zur Ausführung benötigen. Bei einer Benutzungsdauer von einer Stunde beträgt der Median der Ausführungsdauer von einer Stunde beträgt der Median der Ausführungsdauer von 260 ms. Bei zwei Stunden Nutzungsdauer wird der größte Median der Ausführungsdauer von 320 ms benötigt. Spieler, die angegeben haben, dass sie drei oder vier Stunden Touchdisplays benutzen benötigen 277 ms im Median.

Die Fehlerwerte sinken, umso länger die angegebene Benutzungsdauer von Touchdisplays ist. So wird ein Median des Fehlerwertes von 31, bei Spielern die weniger als eine Stunde täglich Touchdisplays benutzen, erreicht. Einen Median des Fehlerwertes von nur noch 27 wird bei drei bis vier Stunden Nutzungsdauer erreicht. Den niedrigsten Median des Fehlerwertes besitzen die Spieler, die angegeben haben, dass sie mindestens vier Stunden pro Tag ein Touchdisplay benutzen.

Folgende Abbildung zeigt den Fehlerwert in Abhängigkeit von dem ausgeübten Druck und Größe der eingegebenen Geste. Dabei werden Gesten mit einem Segment betrachtet.

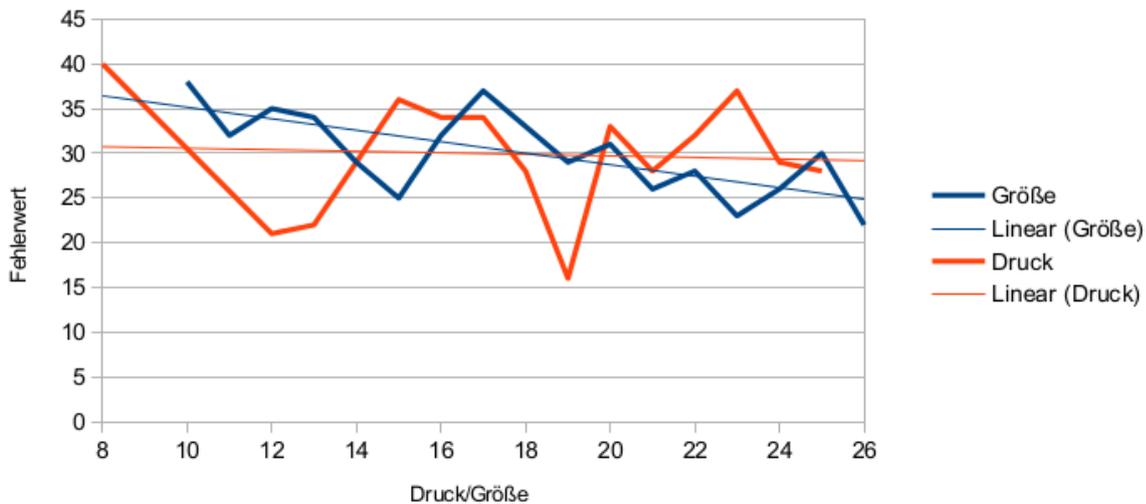


Abbildung 6.14: Diese Abbildung zeigt die Fehlerwerte in Abhängigkeit von Druck und Größe aller Gesten mit einem Segment

Wie in Abbildung 6.14 zu sehen ist, werden die Fehlerwerte mit zunehmendem Druck und Größe der Geste kleiner. Wird die Trendlinie betrachtet, so kann festgestellt werden, dass diese eine negative Steigung besitzen. Der größte Fehlerwert liegt bei 40 mit einem Druck von 8. Bei einem Druck von 19 ist der kleinste Fehlerwert von 16 erreicht worden.

6.3 Diskussion und Vergleich

Werden die Google Play Store Benutzerstudie und die Laborstudie miteinander verglichen, so können viele Gemeinsamkeiten, aber auch einige Unterschiede erkannt werden. In beiden Studien ist zu erkennen, dass die Richtung der Gesten mit einem Segment weder auf die Ausführungszeit, noch auf den Fehlerwert Einfluss hat. Ebenso geht aus beiden Studien hervor, dass die Orientierung der Startgeste keinen Einfluss auf die Ausführungszeit und den Fehlerwert hat. Bei der Google Play Store Studie konnte ein tendenzieller Anstieg der Ausführungszeit beim Erhöhen der Segmentzahl einer Geste beobachtet werden. Im Gegensatz dazu konnte bei der Laborstudie gemessen werden, dass die Durchführung von Geste mit vier Segmenten eine schnellere Ausführungszeit hatte, als eine Geste mit drei Segmenten. Der Grund dafür ist die unzureichende Datenmenge der Gesten mit vier Segmenten. Deswegen wurden die Daten von den drei besten Spielern betrachtet. Die Auswertung von diesen Daten zeigte wieder die Tendenz der steigenden Ausführungszeiten. Die Tatsache, dass sich mit einer Verdoppelung der Gestenlänge, keine Verdoppelung der Ausführungszeit ergibt, könnte folgenden Grund haben: Das Aufsetzen des Fingers am Anfang einer Geste und das Abheben des Fingers am Ende einer Geste, benötigen mehr Ausführungszeit als die eigentliche Geste. Da die Spieler die Gesten unter Zeitdruck

ausgeführt haben, ist anzunehmen, dass die Gesten in einer fließenden Bewegung ohne Abbremsen an den Verbindungsecken erfolgt ist. Dadurch kann auch bei Gesten mit mehreren Segmenten davon ausgegangen werden, dass diese Geste nur einen Start- und End-Punkt besitzt. Dadurch nimmt die eigentliche Geste, im Verhältnis zu den Anfangs- und End-Punkten der Geste, einen kleineren Teil der Ausführungszeit in Anspruch.

Bei dem Fehlerwert kann beobachtet werden, dass sich bei der Google Play Store Studie erhöhte Schwankungen des Fehlerwertes bei Gesten mit einem Segment ergeben. Bei der Laborstudie kann diese Schwankungen bei Gesten mit vier Segmenten beobachtet werden. Warum dies so ist, kann nicht eindeutig aus den Daten bestimmt werden.

Bei den Auswertungen der Benutzungsstunden eines Touchdisplays, bietet die Google Play Store Studie, bei den Ausführungszeiten ein Ergebnis, welches willkürlich erscheint. Möglicherweise können die meisten Spieler ihre Nutzungsdauer von Touchdisplays schlecht einschätzen und verfälschen so das Ergebnis. Bei der Betrachtung der Fehlerwerte, kann allerdings deutlich erkannt werden, dass Spieler, die weniger als eine Benutzungsstunde angegeben haben, eine viel höhere Ausschwenkung des Fehlerwertes besitzen.

Die Auswertung der Ausführungszeiten in der Laborstudie liefert auch ein anscheinend willkürliches Ergebnis. Dennoch kann eine Verbesserung des Fehlerwertes, mit zunehmenden Benutzungsstunden vermerkt werden. Das lässt den Schluß zu, dass sich eine längere Benutzungsdauer von Touchdisplays nicht auf die Ausführungszeit auswirkt, aber auf die Präzision mit der ein Nutzer eine Geste ausführt.

Bei der Betrachtung von dem Druck und der Größe der Toucheingabe kann in beiden Studien festgestellt werden, dass mit einem erhöhten Druck oder Größe die Fehlerwerte kleiner werden. Möglicherweise kommt das daher, dass bei der Ausführung von geraden Gesten ein erhöhter Druck mit dem Finger, das „Ausschwenken“ des Fingers verhindert.

7 Zusammenfassung und Ausblick

Dieses Kapitel fasst die Arbeit zusammen und gibt einen Ausblick über mögliche anschließende Arbeiten

Zusammenfassung

Die Touchbedienung ist die am häufigsten genutzte Form um Smartphones zu steuern. Um die Qualität von Gesten genauer zu betrachten und zu verstehen, wurde eine Anwendung für Android-Geräte entwickelt. Um diese Anwendung für einen großen Benutzerkreis zu machen, wurde diese Anwendung als Spiel entwickelt.

Die Möglichkeit, diese Anwendung in den Google Play Store zu integrieren, hat die Quantität der Daten erhöht. So konnten insgesamt 32.669 Eingabedaten von 288 Personen gesammelt werden. Desweiteren wurde eine Laborstudie durchgeführt, um den Gebrauch der Anwendung unter einer beobachtbaren Umgebung zu bewerten und zu beurteilen. So konnten zusätzlich 2.225 Datensätze betrachtet werden und diese mit den gesammelten Daten aus dem Google Play Store verglichen werden.

Aus den gesammelten Daten kann folgendes Fazit geschlossen werden: Die Richtung einer geradlinigen Geste ist weder für die Ausführungszeit noch für die Fehlerwerte ausschlaggebend. Ob die Startgeste von verwinkelten Gesten eine horizontale oder vertikale Ausrichtung hat, ist für die Ausführungseffizienz ebensowenig ausschlaggebend. Bei Gesten mit mehr Segmenten, kann ein Zuwachs der Ausführungszeit gemessen werden, allerdings bedeutet eine doppelte Länge nicht, dass diese Geste in doppelter Zeit ausgeführt wird. Eine Verdoppelung der Ausführungszeit, tritt erst bei etwa einer Verdreifachung oder Vervierfachung der Gestenlänge auf.

Aus den Werten der freiwilligen Umfrage, kann abgeleitet werden, dass sich mit erhöhter Nutzungsdauer von Touchdisplays am Tag, keine Verbesserung in der Ausführungsgeschwindigkeit ergibt, aber die Fehlerwerte sind geringer umso mehr Nutzungsstunden angegeben wurden.

Auch eine Verbesserung der Ausführungszeit konnte gemessen werden, indem die erste und letzte ausgeführte Geste eines Nutzers betrachtet wurden.

Ausblick

Dieser Abschnitt beschreibt Ideen, was mit dieser Arbeit noch weiter untersucht werden oder weitergeführt werden kann. Diese Arbeit hat sich aus zeitlichen Gründen nur auf gerade

Linien als Gestensegmente konzentriert.

Eine weitere Möglichkeit, interessante Daten zu sammeln wäre, neue Gesten einzuführen, oder die bestehenden komplett auszutauschen. Möglich wären zum Beispiel kreisförmige, schräge oder andere geometrische Formen als Gesten. Es könnte auch versucht werden weitere Informationen über die Eingabeart zu erhalten.

Zum Beispiel könnten manche Stages nur mit einem bestimmten Finger an einer bestimmten Hand durchgeführt werden. Die räumliche Ausrichtung des Smartphones könnte auch eine aufschlussreiche Variable in der Studie sein. Auch die Theorie, dass Gesten an ihrem Start und Endpunkt am zeitintensivsten sind, sollte durch eine erweiterte Datenerhebung untersucht werden. Desweiteren sollte die Anwendung dem Nutzer mehr Motivation vermitteln, da auffällt, dass die Ausführungszahl der Anwendung bei vielen Nutzern niedrig ist. Die Erhöhung der Motivation könnte beispielsweise durch vielfältigere Level oder Auszeichnungen erfolgen. Auch der Schwierigkeitsgrad sollte leicht abgeschwächt werden um so die Anwendung auch für Nutzer, die nicht so viel Erfahrungen mit Anwendungen dieser Art haben, attraktiv zu machen.

Desweiteren sollte die Studie über einen größeren Zeitraum durchgeführt werden, um so mehr Daten sammeln zu können.

Literaturverzeichnis

- [BMo8] O. Bau, W. E. Mackay. OctoPocus: A Dynamic Guide for Learning Gesture-based Command Sets. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology, UIST '08*, S. 37–46. ACM, New York, NY, USA, 2008. doi:10.1145/1449715.1449724. URL <http://doi.acm.org/10.1145/1449715.1449724>. (Zitiert auf Seite 10)
- [HLB⁺10] N. Henze, A. Löcken, S. Boll, T. Hesselmann, M. Pielot. Free-hand Gestures for Music Playback: Deriving Gestures with a User-centred Process. In *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia, MUM '10*, S. 16:1–16:10. ACM, New York, NY, USA, 2010. doi:10.1145/1899475.1899491. URL <http://doi.acm.org/10.1145/1899475.1899491>. (Zitiert auf Seite 10)
- [HRB11] N. Henze, E. Rukzio, S. Boll. 100,000,000 Taps: Analysis and Improvement of Touch Performance in the Large. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services, MobileHCI '11*, S. 133–142. ACM, New York, NY, USA, 2011. doi:10.1145/2037373.2037395. URL <http://doi.acm.org/10.1145/2037373.2037395>. (Zitiert auf Seite 9)
- [HRB12] N. Henze, E. Rukzio, S. Boll. Observational and Experimental Investigation of Typing Behaviour Using Virtual Keyboards for Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12*, S. 2659–2668. ACM, New York, NY, USA, 2012. doi:10.1145/2207676.2208658. URL <http://doi.acm.org/10.1145/2207676.2208658>. (Zitiert auf Seite 9)
- [SSHD⁺13] A. Sahami Shirazi, N. Henze, T. Dingler, K. Kunze, A. Schmidt. Upright or Sideways?: Analysis of Smartphone Postures in the Wild. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services, MobileHCI '13*, S. 362–371. ACM, New York, NY, USA, 2013. doi:10.1145/2493190.2493230. URL <http://doi.acm.org/10.1145/2493190.2493230>. (Zitiert auf Seite 10)
- [SSHG⁺13] A. Sahami Shirazi, N. Henze, R. Goldenberg, B. Schmidt, H. Schmauder, A. Schmidt. Insights into layout patterns of mobile user interfaces by an automatic analysis of android apps. In *Proceedings of the 5th ACM SIGCHI symposium on Engineering interactive computing systems, EICS '13*, S. 275–284. ACM, New York, NY, USA, 2013. doi:10.1145/2480296.2480308. URL <http://doi.acm.org/10.1145/2480296.2480308>. (Zitiert auf Seite 22)

- [TTS⁺13] J. J. Tran, S. Trewin, C. Swart, B. E. John, J. C. Thomas. Exploring Pinch and Spread Gestures on Mobile Devices. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services, MobileHCI '13*, S. 151–160. ACM, New York, NY, USA, 2013. doi:10.1145/2493190.2493221. URL <http://doi.acm.org/10.1145/2493190.2493221>. (Zitiert auf Seite 9)
- [WMW09] J. O. Wobbrock, M. R. Morris, A. D. Wilson. User-defined Gestures for Surface Computing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09*, S. 1083–1092. ACM, New York, NY, USA, 2009. doi:10.1145/1518701.1518866. URL <http://doi.acm.org/10.1145/1518701.1518866>. (Zitiert auf Seite 10)

Alle URLs wurden zuletzt am 27. 11. 2013 geprüft.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Ort, Datum, Unterschrift