

Digitales Assistenzsystem für Reha- Anwendungen im ambulanten und stationären Bereich

Masterthesis an der Fakultät für Gestaltung der Hochschule Augsburg

Erstgutachter: Prof. Jens Müller
Zweitgutachter: M.A. Erich Seifert



Hochschule
Augsburg University of
Applied Sciences

Michael Jünger

Augsburg, 13.07.2016

Erstellungserklärung:

Hiermit erkläre ich, dass ich die vorgelegte Arbeit selbstständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen oder Hilfsmittel verwendet sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.

Name,
Michael Jünger

Datum,
13.07.2016

Unterschrift

Vorwort

Bereits seit Jahren wird an Hard- und Software entwickelt, die vom Anwender körperliche Aktivität verlangen, um ein vorgegebenes Ziel zu erreichen. Speziell im Bereich der Rehabilitation (Reha) ist diese Anwendungsmöglichkeit interessant, da der Patient eine Übung dadurch selbstständig ohne Trainer, jedoch mittels Unterstützung der Software durchführen kann. Entsprechende Gerätschaften hierfür sind bereits verfügbar und dürften in nahezu jedem Reha-Zentrum vorzufinden sein. Dabei handelt es sich um professionelle Geräte, die meist nur einem definierten Zweck dienen, wie etwa der Verbesserung des Koordinationsvermögens. Solche Geräte sind jedoch zumeist sehr teuer, weshalb für eine Privatperson deren Anschaffung kaum vorstellbar ist.

Doch auch im privaten Sektor hat sich einiges getan. Spätestens mit dem Release der Anwendung Wii fit™ für die Spielekonsole Wii von Nintendo ist es möglich, sportliche Aktivitäten Zuhause durchzuführen. Der Anwender wird dabei von einem virtuellen Trainer unterstützt, der Feedback über die Körperhaltung und Durchführung der Übung gibt. Die Daten hierfür erhält die Software zum einen über das Balance-Board, auf das sich der Anwender stellt. Mittels Gewichtsverlagerung werden Bewegungen registriert, wodurch die Software beispielsweise erkennen kann, ob der Anwender gerade einen Hula-Hopp-Reifen schwingt. Zum anderen ist im Handcontroller ein Gyroskop verbaut, welches Aufschluss über die Haltung der Hand gibt. Andere Hersteller wie Sony mit der PlayStation oder Microsoft mit der Xbox, haben ähnliche Produkte im Angebot.

Allen Produkten für den Heimgebrauch gemein ist, dass sie hauptsächlich der Unterhaltung dienen. Wenngleich sie auch einen positiven Effekt auf die Physiologie des Anwenders haben können, so wurden diese Produkte nicht explizit zur Unterstützung von Reha-Maßnahmen entwickelt. Diesen Umstand zu ändern, ist ein Ziel dieser Arbeit.

Denn zum einen braucht eine Gesellschaft, die durch medizinische Fortschritte im-

mer älter wird, neue Therapiemöglichkeiten, um den degenerativen Prozessen des Alterns entgegenzuwirken. Zum anderen benötigen auch weiterhin junge Menschen nach einem Unfall eine Reha. Die personellen und finanziellen Ressourcen zur Aufrechterhaltung einer qualitativ hochwertigen Reha sind in einer Gesellschaft jedoch begrenzt.

Die Entwicklung eines computergestützten Reha-Programms, welches qualitativ den Anforderungen einer durch einen Trainer begleiteten Reha gerecht wird, gibt personelle Ressourcen frei, die für Patienten eingesetzt werden können, denen ein selbstständiges Training physiologisch nicht möglich ist. Gleichzeitig sind die Kosten einer rein computergestützten Therapie geringer, was eine Einsparung von finanziellen Ressourcen zur Folge haben kann. Die so freigegebenen finanziellen Ressourcen können der Weiterentwicklung neuer, leistungsfähiger Systeme und den Patienten zur Verfügung stehen, die einen Bedarf an einer intensiven Reha mit Begleitung durch einen Trainer haben.

Weiter ist durch eine solche Anwendung die zunehmende Verlagerung der Reha in stationäre Einrichtungen wie Krankenhäuser, Seniorenheime und dem privaten Bereich möglich, wodurch die Auslastung einer Reha-Einrichtungen gesenkt werden kann. Diese Entlastung kann dann eine Spezialisierung innerhalb der Einrichtung auf Härtefälle zur Folge haben, was wiederum die Qualität der Reha für schwere Fälle steigert.

Um dies zu erreichen muss ein solches System möglichst kostengünstig sein, damit auch Privatanwender die finanziellen Mittel dafür aufbringen können. Das Hauptziel der Arbeit ist demnach, den Prototyp eines Assistenten für die Reha zu entwickeln, der in der Preisklasse vergleichbar mit der Anschaffung einer Spielkonsole ist.

Inhaltsverzeichnis

Inhaltsverzeichnis	III
Abkürzungsverzeichnis	V
Abbildungsverzeichnis	VI
1 Begriffserklärungen	1
1.1 Personal Computer	1
1.2 Xbox	1
1.3 Kinect	1
1.4 JavaScript Object Notation	1
1.5 Extensible Markup Language	2
1.6 Rehabilitation	2
1.7 SQLite	2
1.8 Django	2
1.9 Python	2
1.10 Object Relational Mapping	3
1.11 Syntax-Highlighting	3
2 Reha-/Exergames	4
2.1 Entwicklung	4
2.2 Anwendungen für medizinische Zwecke	5
2.3 Anwendungen für spielerische Zwecke	6
3 Spielkonzept	7
3.1 Workflows	7
3.1.1 Flow bei Erstbenutzung	8
3.1.2 Workflow bei bestehenden Nutzern	11
4 Technisches Konzept	12
4.1 Die Komponenten im Überblick	12
4.1.1 Xbox/Personal Computer (PC)	12

4.1.2	TV/Monitor	13
4.1.3	Kinect	13
4.1.4	PC-Adapter	14
4.1.5	Lautsprecher	15
4.2	Rehabilitation Execution Markup Language	15
4.3	Object Relational Mapping für Unity und SQLite	16
4.4	Reha Game Database Editor	20
5	Visuelles und auditives Designkonzept	22
5.1	Farbgebung	22
5.1.1	Farben in Menüs	22
5.1.2	Farben in der Übung	23
5.2	Auditives Feedback	24
6	Wirtschaftliches Konzept	25
6.1	Privathaushalte	26
6.2	Medizinische Einrichtungen	27
7	Schlussteil	28
	Literaturverzeichnis	31

EPL Entwicklung Produktion Lautenschläger	5
JSON JavaScript Object Notation	VI
ORM Object Relational Mapping	17
PC Personal Computer	III
Reha Rehabilitation	I
XML Extensible Markup Language	VI
API Application Programming Interface	13
RGDBE Reha Game Database Editor	
REML Rehabilitation Execution Markup Language	VI

Abbildungsverzeichnis

2.1	CortexTrainer	5
3.1	Flowchart bei Erstbenutzung	8
3.2	Neuen Benutzer erstellen	8
3.3	Hauptmenü	9
3.4	Übungsauswahl	9
3.5	Detailansicht einer Übung	10
3.6	Ausführung einer Übung	10
3.7	Statistikmenü	11
3.8	Flowchart bei existierenden Nutzern	11
4.1	Technischer Aufbau	12
4.2	Kinect v2	13
4.3	Kinect Gelenkmodell	14
4.4	Kinect Distanzmessung	14
4.5	Rehabilitation Execution Markup Language (REML) im JavaScript Object Notation (JSON)-Format	15
4.6	REML im Extensible Markup Language (XML)-Format	16
4.7	Klassendiagramm der Datenbank-Modelle	17
4.8	Auszug des Datenbankmodell Patient	18
4.9	Many to Many Relation	19
4.10	Annotationen	20
4.11	Neue Übung erstellen	21
4.12	Übersicht der Übungen	21
5.1	Das Einstellungsmenü	23
5.2	Farben der Gelenke in der Übung	23
6.1	Privathaushalte mit PC bis 2015	25

1. Begriffserklärungen

Die im Folgenden genannten Begriffe sind Definitionen häufig genannter relevanter Fachbegriffe der IT oder Reha

1.1 Personal Computer

Ein kleiner stationärer oder tragbarer Computer für den Büro- oder Heimgebrauch

1.2 Xbox

Eine Spielekonsole von Microsoft

1.3 Kinect

Ein Eingabegerät als Zubehör für die Xbox, das mittels Kameras Personen im Raum erfassen und deren Bewegungen erkennen kann.

1.4 JavaScript Object Notation

Ein Textformat um Daten zwischen verschiedenen Geräten und Plattformen auszutauschen (Bassett, 2015, vgl. JSON Is a Data Interchange Format)

1.5 Extensible Markup Language

Ein Dateiformat, welches der Speicherung von Daten dient und diese auf unterschiedlichen Systemen verfügbar macht (Koch, 2010, XML – ein Blick zurück und voraus, 1)

1.6 Rehabilitation

Therapeutische Maßnahmen zur Heilung oder Verbesserung körperlicher Beeinträchtigungen

1.7 SQLite

Eine leichtgewichtige auf SQL (Script Query Language) basierte relationale Datenbank

1.8 Django

Ein Web-Framework basierend auf der Programmiersprache Python

1.9 Python

Eine freie, plattformunabhängige Programmiersprache

1.10 Object Relational Mapping

Übersetzt die Tabellen einer relationalen Datenbank in ein Objekt eines Computerprogrammes (Dentler, 2010, vgl. Introduction)

1.11 Syntax-Highlighting

Darunter ist die grafische Aufbereitung einer definierten Programmiersprache gemeint. Schlüsselwörter, Ausdrücke und Datentypen werden dabei in unterschiedlichen Farb- und Textstilen dargestellt, was die Lesbarkeit von Programmcode deutlich erhöht.

2. Reha-/Exergames

„Exergames sind Computerspiele, die über Körperbewegungen gesteuert werden und so zu mehr Bewegung motivieren“ (Viktor Wendel, Sandro Hardy, Stefan Göbel, Ralf Steinmetz, 2010, Personalisierung Exergames, 1). Es handelt sich folglich um Computerspiele, für deren Steuerung spezielle Eingabegeräte benötigt werden, die Bewegungen des Körpers in die virtuelle Spielwelt übertragen. In Abgrenzung zu klassischen Eingabemethoden wie Tastatur, Maus oder Joystick/Gamepad ist hier jedoch der Einsatz des ganzen Körpers gefragt und nicht nur die Bewegungen der Finger oder Hand.

2.1 Entwicklung

Die Anfänge der sogenannten Exergames können bis auf das Jahr 1983 zurückdatiert werden. In diesem Jahr brachte die Firma Amiga Computer Inc. das Joyboard auf den Markt. Dabei handelt es sich technisch um einen Joystick, auf den sich der Spieler jedoch stellen musste. Es war daher besonders gut geeignet um Sportspiele wie Ski fahren, surfen oder olympische Disziplinen, bei denen viel rennen an Ort und Stelle erforderlich war, zu spielen (Mical, 1987, How We Created the AMIGA Computer, 45).

Seitdem brachten diverse Hersteller Spiele auf den Markt, die vom Spieler mehr forderten, als die bloße Steuerung mittels Finger über einen Controller. Etwa der Power Glove von Mattel aus dem Jahre 1989, der explizit für den NES entwickelt wurde oder Dance Dance Revolution aus dem Jahr 1998 von Konami. Der große Durchbruch allerdings ließ einige Zeit auf sich warten, als 2008 die Anwendung Wii fit™ auf den Markt kam (Wikipedia, 2014, Wii fit™).

Seitdem hat sich der Markt im Consumer-Bereich nicht nennenswert verändert. Zwar gibt es einige neue und interessante Konzepte sowie Eingabemöglichkeiten. Zum Beispiel die Kinect von Microsoft, die den ganzen Körper lediglich mittels verschiedener Kameras trackt (engl. Verfolgt) und die Bewegungen unmittelbar auf das zu steuernde Objekt überträgt. Bei der Leap Motion, die ein ähnliches Konzept wie die Kinect verfolgt, liegt der Fokus auf Gestensteuerung per Hand und Finger. Die dafür vorgesehenen Anwendungen allerdings dienen mehr dem Unterhaltungsfak-

tor, als dem therapeutischen und sind nicht explizit für die Reha geschaffen.

Der Markt für Exergames lässt sich grob in zwei Bereiche unterteilen.

2.2 Anwendungen für medizinische Zwecke

In diesem Marktsegment agieren wenige spezialisierte Unternehmen, die ihre Produkte gezielt für medizinische Einrichtungen entwickeln, wo diese fast ausschließlich zum Einsatz kommen. Ein Beispiel hierfür ist der Physiomat bzw. dessen Nachfolgemodell, CortexTrainer, der Firma Entwicklung Produktion Lautenschläger (EPL). Anwendung findet der Physiomat etwa im Reha-Zentrum ProVita in Augsburg. Laut den dort beschäftigten Trainern sei er für Gleichgewichts-, Konzentrations-, Koordinations- und Krafttraining geeignet, jedoch handele es sich um ein komplexes Gerät, welches sehr schwer zu bedienen sei.

Abbildung 2.1: CortexTrainer



Derartige Geräte bieten ein hohes Maß an Professionalität, da sie nicht zuletzt in Kooperation mit Reha-Zentren und Fachleuten entwickelt werden. Dadurch sind sie perfekt an die Anforderungen der vorgesehenen Übungen angepasst. Zwar soll eine solche Übung den Anwender nicht langweilen, dennoch liegt hier der Fokus klar auf dem therapeutischen Nutzen, was zu Lasten der grafischen Darstellung geht. Gerade bei jüngere Patienten, die Videospiele in realistischer Grafik gewohnt sind, kann sich dies negativ auf deren Motivation ausüben die Anwendung zu nutzen. Zudem ist, wie bereits angedeutet, die Bedienung komplex und erfordert somit geschultes Personal. Dadurch verteuert sich die Anschaffung neben den eigentlichen Kosten für die Hardware zusätzlich und Wartungsarbeiten sowie Reparaturen können nur vom Produzenten durchgeführt werden.

2.3 Anwendungen für spielerische Zwecke

Hierunter sind solche Produkte zu verstehen, die für den Consumer-Bereich, also für den privaten Gebrauch, entwickelt werden. Anders als bei den Anwendungen für medizinische Zwecke steht hier jedoch vorwiegend der Unterhaltungsfaktor im Vordergrund. Solche Spiele bieten somit meist eine ansprechende Grafik und ein durchdachtes Spielkonzept. Die körperliche Betätigung, die zum Erreichen des Spielziels vom Spieler verlangt wird, ist nicht der Zweck des Spiels, sondern vielmehr ein gewollter Nebeneffekt, der sich zudem gut verkauft. Dies soll nicht bedeuten, dass derartige Produkte für therapeutische Zwecke ungeeignet sind, sondern es soll vor Augen werfen, dass es gegebenenfalls Einschränkungen für die Therapie geben kann, da diese Produkte eben nicht explizit dafür entwickelt wurden. Verschiedene Studien konnten bei regelmäßigem Training dennoch positive Auswirkungen auf die Physiologie der Anwender nachweisen. Laut einer Studie der Universität Leipzig konnte das Training mit der Anwendung Wii fit™ die Reaktionsfähigkeit von Senioren steigern (Froeb, 2011, vgl. Gleichgewichtstraining mit Nintendo® Wii fit™, 184).

Folglich liegt hier ein Potential, dass es für therapeutische Zwecke nutzbar zu machen gilt. Die Vorteile liegen dabei klar auf der Hand. Zum einen ist die Hardware bereits vorhanden und wird stetig weiter entwickelt. Sie ist zudem vergleichsweise kostengünstig, wodurch es auch kleinen Einrichtungen und privaten Haushalten möglich wird, die erforderlichen Gerätschaften zu erwerben. Zudem sind die Geräte kompakt, wodurch sie wenig Platz in Anspruch nehmen.

3. Spielkonzept

Beim RehaGame handelt es sich weniger um ein Spiel, als vielmehr um eine Anwendung. Ziel ist es, durch kontinuierliche Übungen den physiologischen Zustand des Anwenders zu verbessern. Dazu stehen verschiedene Übungen zur Auswahl, aus denen der Benutzer die für ihn passende wählen kann. Diese kann er so oft wiederholen, wie er möchte.

Am Ende einer Übung bekommt er dann eine Auswertung über die Ausführung der Übung angezeigt. Über das Benutzerprofil gelangt der Anwender ebenfalls in das Statistik-Menü, wo ihm über diesen Weg allerdings alle gesammelten Daten zu den bisher ausgeführten Übungen angezeigt werden. Dies dient der Kontrolle über die Genauigkeit der Ausführung und ist nicht zuletzt vor allem für den Trainer interessant.

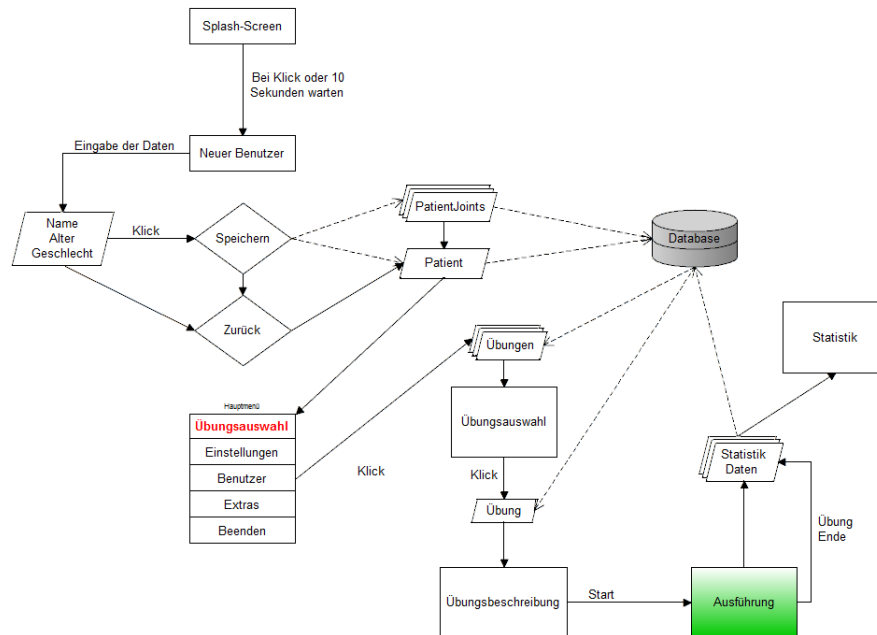
Die Anwendung ist zudem personalisierbar und kann mehr als einen Benutzer verwalten. Dadurch können mehrere Anwender am gleichen Gerät trainieren.

3.1 Workflows

Die Anwendung soll für den Benutzer möglichst einfach zu bedienen sein. Daher folgt die Anwendung zwei verschiedenen Pfaden. Welchem der Pfade gefolgt wird ist abhängig davon, ob es bereits einen Benutzer im System gibt, oder ob bisher keiner angelegt wurde, was bei Erstbenutzung der Fall ist.

3.1.1 Flow bei Erstbenutzung

Abbildung 3.1: Flowchart bei Erstbenutzung



Bei Erstbenutzung der Anwendung oder wenn kein Benutzer mehr vorhanden ist, lädt das Programm direkt nach dem Splash-Screen das Menü zum Anlegen eines neuen Benutzers.

Benutzermenü

Hier erhält der Anwender eine Eingabemaske für den Namen, das Alter und sein Geschlecht. Füllt er diese aus und bestätigt er die Aktion durch speichern, wird er als Benutzer im System angelegt.

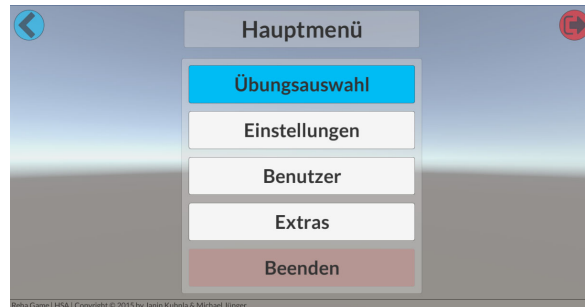
Abbildung 3.2: Neuen Benutzer erstellen



Nach einem Klick auf den Zurück-Button, oben links, gelangt er zurück ins Hauptmenü. Dort hat er die Möglichkeit in das Menü Übungsauswahl zu wechseln, Ein-

stellungen vorzunehmen, sein Profil aufzurufen sich Zusatzinformationen über die Anwendung anzeigen zu lassen oder das Programm zu beenden.

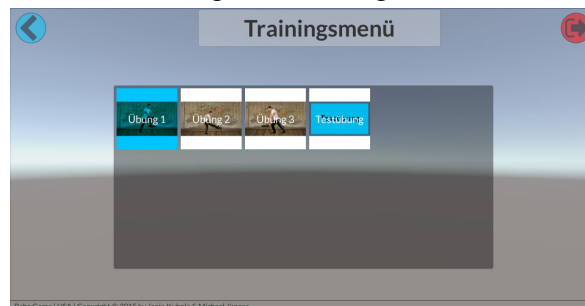
Abbildung 3.3: Hauptmenü



Übungsauswahl

Hat der Anwender seinen Benutzer im System angelegt und möchte nun eine Übung starten, so muss er zunächst in die Übungsauswahl. Der vorliegende Prototyp beinhaltet drei Übungen. Für die korrekte Ausführung wurde mit dem Reha-Zentrum Pro-Vita aus Augsburg zusammen gearbeitet. Der Anwender kann nun eine der Übungen wählen, worauf hin sich die Detailansicht der Übung lädt.

Abbildung 3.4: Übungsauswahl

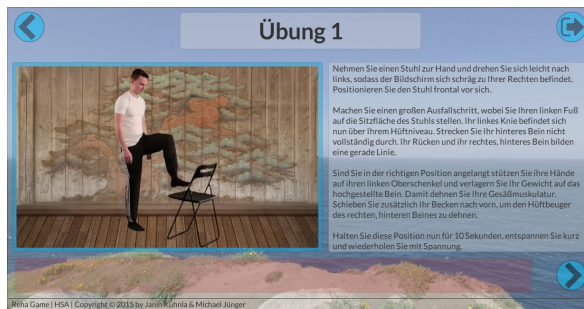


Detailansicht der Übung

In der Detailansicht werden alle Informationen zur Übungsausführung angezeigt. Über ein Video wird die Ausführung bildhaft erklärt. Die Videobox ist zu Beginn ausgewählt, die Wiedergabe startet bei einem Klick auf diese. Rechts vom Video befindet sich eine textuelle Beschreibung der Ausführung. Diese kann sich der Anwender vorlesen lassen, in dem er die Box auswählt und darauf klickt. Unterhalb des Videos und der Textbeschreibung befindet sich eine weitere Textbox für zusätzliche

Informationen, wenn es bei der Übung etwas besonderes zu beachten gilt. Auch diesen Text kann sich der Anwender vorlesen lassen.

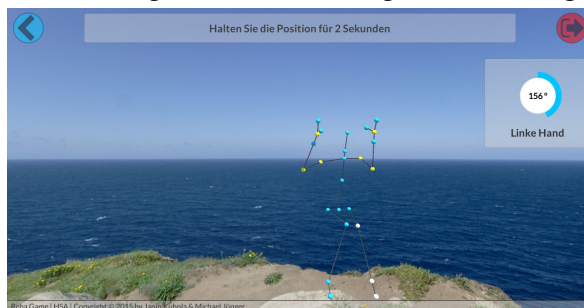
Abbildung 3.5: Detailansicht einer Übung



Die Übungsausführung

Wurde die Übung gestartet, muss der Anwender den Anweisungen folgen. Diese werden am oberen Bildschirmrand mittig angezeigt. Auf der rechten Seite befindet sich weiter eine kleine Infobox. Diese zeigt gegenwärtig den Winkel des aktiven Gelenks an. Die Farbgebung der einzelnen Gelenke wird im Kapitel *Visuelles und auditives Designkonzept* näher erläutert.

Abbildung 3.6: Ausführung einer Übung

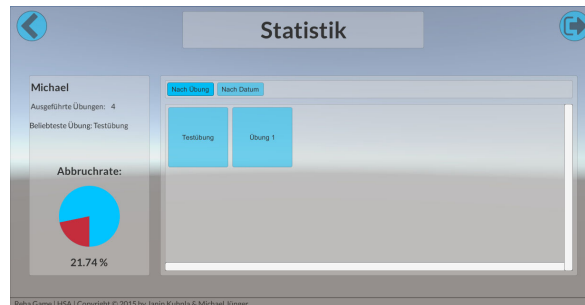


Statistik

Wurde die Übung erfolgreich abgeschlossen, so gelangt der Benutzer automatisch in die Statistik. Dort erhält er neben allgemeinen Daten zu den bislang abgeschlossenen Übungen und der am häufigsten ausgeführten Übung Daten über die Abbruchrate der Übungsschritte. Die Daten können nach Übung oder Datum gruppiert werden und beziehen sich auf das jeweils aktive Objekt. Ist kein Objekt ausgewählt,

werden alle Daten in die Berechnung einbezogen. Wird stattdessen eine Übung gewählt, werden nur die Daten dieser Übung berücksichtigt und bei einem Datum nur die Daten von diesem Tag.

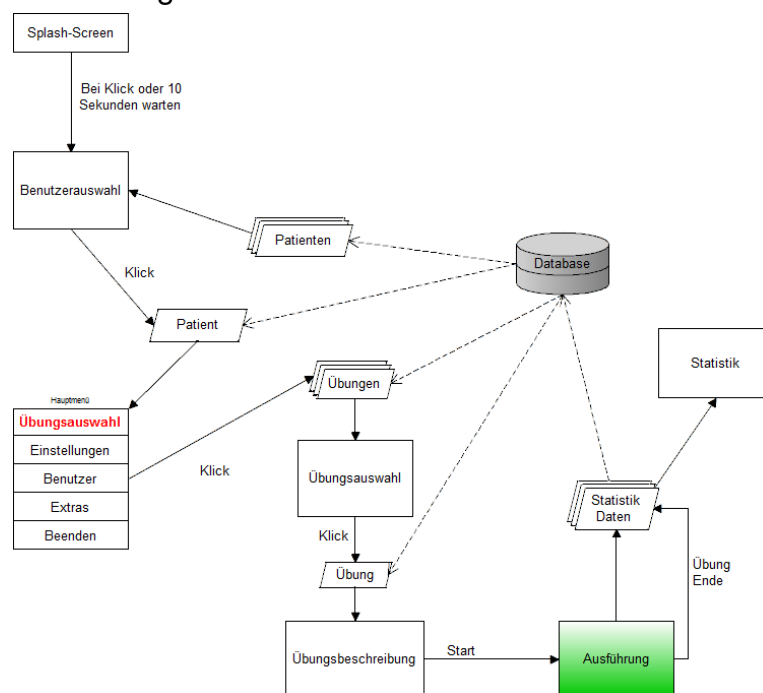
Abbildung 3.7: Statistikmenü



3.1.2 Workflow bei bestehenden Nutzern

Existiert mindestens ein Benutzer, lädt die Anwendung nach dem Splash-Screen direkt die Benutzerauswahl. Von hier aus kann nun entweder ein Benutzer gewählt oder das Menü zum erstellen eines neuen Benutzers aufgerufen werden. Wird ein Benutzer gewählt, wird das Hauptmenü geladen. Ansonsten unterscheidet sich der Workflow nicht von der Erstbenutzung.

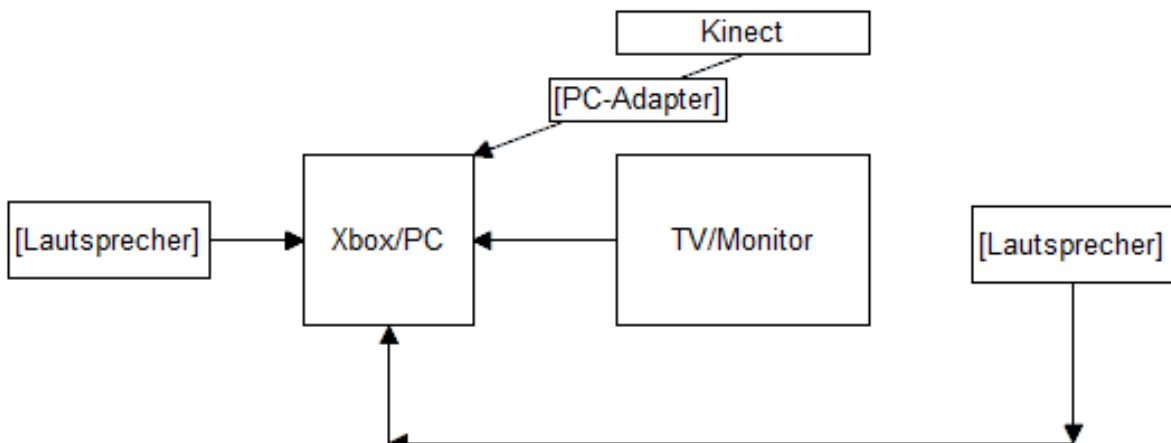
Abbildung 3.8: Flowchart bei existierenden Nutzern



4. Technisches Konzept

Der technische Aufwand sollte möglichst gering gehalten werden, um die Anwendung auch im privaten Umfeld zu ermöglichen, ohne dafür ein besonders hohes technisches Knowhow zu besitzen. Bei den verwendeten Komponenten handelt es sich daher nicht um komplizierte Eigenentwicklungen, sondern um gängige Hardware aus dem Bereich der Consumer-Electronic. Dadurch ist für die meisten Fälle sichergestellt, dass, sollte die betroffene Person selbst nicht dazu in der Lage sein, eine dritte Person aus dem nahen Umfeld wie Angehörige, Freunde oder eine Pflegekraft die Installation übernehmen kann.

Abbildung 4.1: Technischer Aufbau



4.1 Die Komponenten im Überblick

Im Wesentlichen besteht das System aus drei Hauptkomponenten: Dem Rechner, der Kinect als Eingabegerät und einem Bildschirm als Ausgabegerät. Optional können an das System Lautsprecher angeschlossen werden, sie sind zur Verwendung aber nicht zwingend erforderlich.

4.1.1 Xbox/PC

Die Xbox bzw. der PC ist das zentrale Gerät, auf dem die Anwendung schlussendlich läuft. Im Fall, dass ein PC verwendet wird, ist kein besonders leistungsfähiger

Rechner notwendig. Es muss lediglich darauf geachtet werden, dass er - aktuell für die Kinect 2 - mindestens den USB 3.0 Standard unterstützt. Mangels Hardware wurde die Anwendung nicht auf einer Xbox getestet, es sollte jedoch grundsätzlich möglich sein, da die verwendete 3D-Spielengine Unity dies unterstützt.

4.1.2 TV/Monitor

Ein Bildschirm ist für die Darstellung der Anwendung notwendig. Darauf sieht der Patient nicht nur das Menü und die Spielwelt, sondern er erhält zusätzlich visuelles Feedback während der Übungsausführung.

4.1.3 Kinect

Die Kinect ist, neben Maus und Tastatur am PC bzw. dem Controller bei der Xbox, das zentrale Eingabegerät für die Durchführung der Übung.

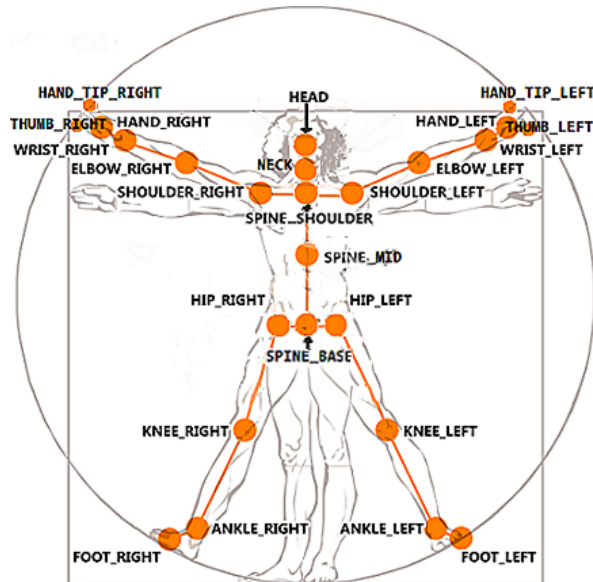
Abbildung 4.2: Kinect v2



Durch sie wird der Anwender und seine Bewegungen erkannt. Die erfassten Daten werden daraufhin an den Rechner gesendet, dort verarbeitet und schlussendlich auf ein virtuelles Objekt übertragen, welches dann der Bewegung folgt. Die Kinect in der zweiten Version kann dabei sechs Personen mit je 25 Gelenkpunkten erkennen und verarbeiten. Über die Application Programming Interface (API) erhält der Programmierer zugriff auf jede Person und deren Gelenke. Darüber erhält er Zugang zu den Daten wie der Position und Drehung des jeweiligen Gelenks und kann diese entsprechend den Anforderungen der Anwendung nutzen.

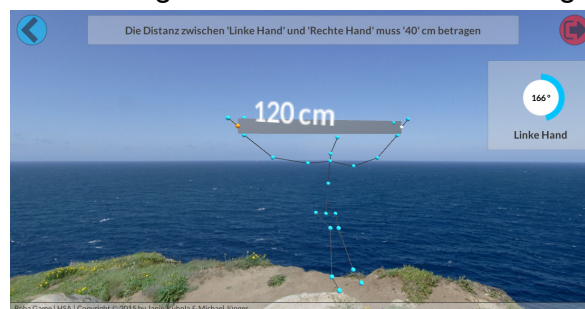
Unter den 25 Gelenken die von der Kinect erkannt werden, sind nicht alle Gelenke des Menschlichen Körpers enthalten. Es fehlen einzelne Finger, die Zehen und einige Wirbel. Die erkannten Gelenke reichen jedoch aus, um die genaue Bewegungsabfolge des Gesamten Körpers zu erfassen.

Abbildung 4.3: Kinect Gelenkmodell



Die erfassten Daten der Kinect sind sehr genau. In einem Test wurde der Abstand beider Hände gemessen und die berechnete Distanz am Monitor ausgegeben. Dazu wurde ein 30cm langes Lineal zu Hilfe genommen, welches mit den Handflächen festgehalten wurde, sodass der Abstand beider Hände der Länge des Lineals entsprochen hat. Die gemessenen Daten schwankten dabei minimal um ein paar wenige Zentimeter.

Abbildung 4.4: Kinect Distanzmessung



4.1.4 PC-Adapter

Anders als beim Vorgänger ist für die Kinect 2 keine zusätzliche Ausführung für Windows-basierte Rechner erhältlich. Soll für die Anwendung ein PC verwendet

werden, ist daher zum Betrieb ein zusätzlicher Adapter notwendig. Dieser versorgt die Kinect mit Strom und bietet die Anschlussmöglichkeit an eine USB 3.0 Schnittstelle am PC.

4.1.5 Lautsprecher

Die Lautsprecher dienen in erster Linie dem auditiven Feedback und der musikalischen Untermalung in den Menüs. Sie sind optional und für die Ausführung der Übung nicht zwingend erforderlich. Im Falle einer Seh- oder Leseschwäche erhält der Anwender darüber jedoch wertvolle Informationen, ohne die eine Ausführung der Übung dann nicht möglich ist.

4.2 Rehabilitation Execution Markup Language

Die größte Herausforderung für die Entwicklung der Anwendung war es, eine Sprache zu definieren, mit deren Hilfe sich Anweisungen, beispielsweise die eines Trainers, in eine für den Computer interpretierbare Form transformieren lassen. Überlegungen hierzu waren, die Anweisungen in Form von JSON zu speichern. Ein Objekt wird dabei in seine Eigenschaften zerlegt, die dann als String beschrieben werden

Abbildung 4.5: REML im JSON-Format

```
{
  "exercise": {
    "-name": "Test",
    "step": [
      {
        "-name": "Left hand above right hand",
        "joint": {
          "-name": "HandLeft",
          "behaviour": {
            "-is": "above",
            "-joint": "HandRight"
          }
        }
      },
      "action": {
        "-name": "hold",
        "-value": "5"
      }
    ]
  },
  {
    "-name": "Right hand above left hand",
    "joint": {
      "-name": "HandRight",
      "behaviour": {
        "-is": "above",
        "-joint": "HandLeft"
      }
    },
    "action": {
      "-name": "hold",
      "-value": "5"
    }
  }
]
```

Besteht ein Objekt aus Eigenschaften und weiteren, kleineren Objekten, so ist auch eine Verschachtelung möglich. Bei vielen Eigenschaften und Objekten muss der

JSON-String allerdings maschinell erzeugt werden. Zudem ist er sehr unübersichtlich und kann vom Menschen nicht ohne weiteres gelesen und verstanden werden.

Daher wurde auf den Einsatz einer XML-Sprache gesetzt, da diese übersichtlicher und daher leichter zu verstehen ist.

Abbildung 4.6: REML im XML-Format

```
<exercise name="Test">
  <step name="Left hand above right hand">
    <joint name="HandLeft">
      <behaviour is="above" joint="HandRight"></behaviour>
    </joint>
    <action name="hold" value="5"></action>
  </step>
  <step name="Right hand above left hand">
    <joint name="HandRight">
      <behaviour is="above" joint="HandLeft"></behaviour>
    </joint>
    <action name="hold" value="5"></action>
  </step>
</exercise>
```

Die Vorteile gegenüber der Speicherung als JSON-String liegen auf der Hand. Während das JSON-Format vom Menschen ohne technische Hilfsmittel kaum mehr erfassbar ist, bleibt die Notation im XML-Format auch in ihrer reinen Textdarstellung ohne Syntax-Highlighting verständlich, da die Definition der Sprache selbst sprechend ist. Es ist dadurch auch einem versierten Laien möglich, eigene Übungen zu definieren.

4.3 Object Relational Mapping für Unity und SQLite

Die Speicherung sämtlicher Daten erfolgt mittels einer SQLite-Datenbank. In dieser sind sowohl die Übungen inklusive der zugehörigen Ressourcen, als auch die Menüeinträge gespeichert. Ebenso die Daten der Anwender wie Name, Alter, Geschlecht, die Gelenke und deren Zustand (Vorhanden oder nicht), sowie Daten über die Ausführung der Übungen. Innerhalb des Programms werden diese Daten in Form von Objekten repräsentiert. Ein solches Objekt, beispielsweise der Patient, hat Eigenschaften wie den Namen, das Alter und sein Geschlecht. Zusätzlich hat dieses Objekt jedoch auch Verweise auf andere Objekte wie Gelenke oder Übungen, die ihm zugewiesen sind, die ihrerseits wiederum aus Eigenschaften und Verweisen auf andere Objekte bestehen können.

In der Datenbank sind diese Daten ähnlich in Form von Tabellen gespeichert. Dort gibt es beispielsweise eine Tabelle Patient mit Spalten für dessen Eigenschaften,

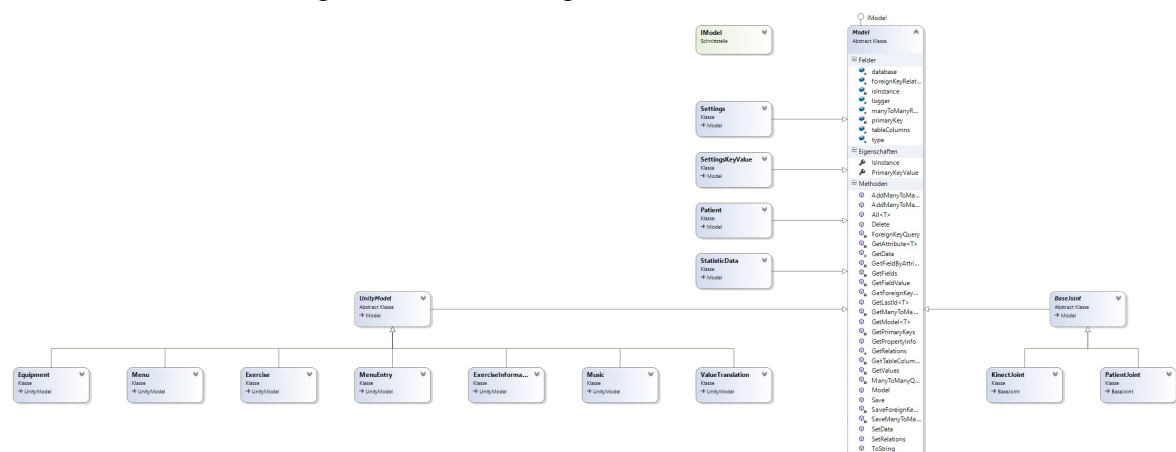
sowie weiteren für die jeweilige eindeutige ID zu dem Datensätzen aus anderen Tabellen, zum Beispiel der Tabelle für Gelenke und Übungen.

Um nun das Objekt Patient mit all seinen Eigenschaften und Abhängigkeiten der anderen Objekte zu erzeugen, müssen die Daten des entsprechenden Patienten aus der Tabelle Patient und den jeweiligen anderen Tabellen eingelesen werden. Dies geschieht mittels eines Object Relational Mapping (ORM). Der ORM weiß, welches Objekt zu welcher Tabelle gehört und welche Eigenschaft der jeweiligen Spalte zugeordnet ist. Im Idealfall wird der ORM angewiesen, das Objekt Patient anhand des Namens zu erzeugen. Der Rückgabewert dieser Anweisung an das ORM ist der gewünschte Patient, befüllt mit all seinen Eigenschaften und Objekten, die von diesem abhängen. Das ORM erzeugt folglich nicht nur ein Objekt, sondern gegebenenfalls eine ganze Reihe von Objekten, eben diese, die zum gegenwärtigen Zeitpunkt benötigt werden.

Während diese Arbeit erstellt wurde war ein ORM für Unity leider nicht verfügbar. Da mit dessen Hilfe die Arbeit jedoch wesentlich vereinfacht wird, wurde im Rahmen der vorliegenden Arbeit ein solches ORM entwickelt. Vorneweg sei jedoch erwähnt, dass es sich nicht um ein vollwertiges ORM handelt, da es die Datenbankstruktur selbst nicht verändern, sondern lediglich Datensätze der bestehenden Struktur erstellen, ändern und lesen kann.

Jedes Datenbank-Objekt erbt dabei von einer gemeinsamen Basisklasse namens Model.

Abbildung 4.7: Klassendiagramm der Datenbank-Modelle



Dieses stellt allen ererbenden Klassen unter anderem die Grundfunktionen bereit, um es aus der Datenbank zu laden, zu speichern oder zu löschen. Jede Spalte wird im Objekt durch einen jeweiligen getter und setter repräsentiert. Über diese kann das

ORM den Wert der Spalte in die entsprechende Eigenschaft des Objektes schreiben oder diesen daraus zum Speichern in der Datenbank lesen. Ob eine Eigenschaft eines Objektes eine Spalte in der Datenbank repräsentiert, wird mittels Annotationen über ihr definiert. Entsprechend markierte Eigenschaften werden durch das Reflection Framework eingelesen. Das ORM ist dadurch flexibel und es können ohne weiteres neue Datenbankmodelle abgebildet werden.

Abbildung 4.8: Auszug des Datenbankmodell Patient

```
30 [PrimaryKey]
31 7 Verweise | Michael Jünger, vor 151 Tagen | 1 Autor, 2 Änderungen
32 public string Name
33 {
34     get
35     {
36         return this.name;
37     }
38     set
39     {
40         this.name = value;
41     }
42 }
43
44 [TableColumn]
45 2 Verweise | Michael Jünger, vor 154 Tagen | 1 Autor, 1 Änderung
46 public long Age
47 {
48     get
49     {
50         return this.age;
51     }
52     set
53     {
54         this.age = value;
55     }
56 }
57
```

Derzeit können folgende Annotationen zugewiesen werden:

- **TableColumn:** Dies ist der Basistyp aller weiteren Annotationen und repräsentiert zugleich eine einfache Spalte der Tabelle.
- **PrimaryKey:** Dabei handelt es sich um den Primärschlüssel des Modells und ist somit in jedem Modell erforderlich. Er ist eindeutig und kann daher nur ein Mal in einer Tabelle vorkommen. Über ihn wird der gesamte Datensatz des Modells in der Tabelle identifiziert.
- **ForeignKey:** Der Fremdschlüssel ist eine Relation auf genau ein Modell in einer anderen Tabelle. In der Datenbank ist in dieser Spalte der Primärschlüssel des Datensatzes gespeichert, die Eigenschaft in der Anwendung beinhaltet das Objekt.
- **ManyToManyRelation:** Wie beim Fremdschlüssel handelt es sich hier um eine Relation auf ein Modell in einer anderen Tabelle. Im Unterschied zum Fremd-

schlüssel kann dieser Wert jedoch mehrere Relationen des gleichen Typs enthalten. In der Datenbank erfolgt die Zuordnung zumeist mittels einer Hilfstabelle, in der jeweils die Primärschlüssel der Relationen gespeichert sind. So kann ein Patient namens Herbert hier mehrfach vorkommen, jedoch mit jeweils einem anderen Wert auf eine Übung. In der Anwendung ist die Speicherung mittels eines Dictionary realisiert, wobei als Key jeweils der Primärschlüssel und als Value das Objekt gespeichert wird.

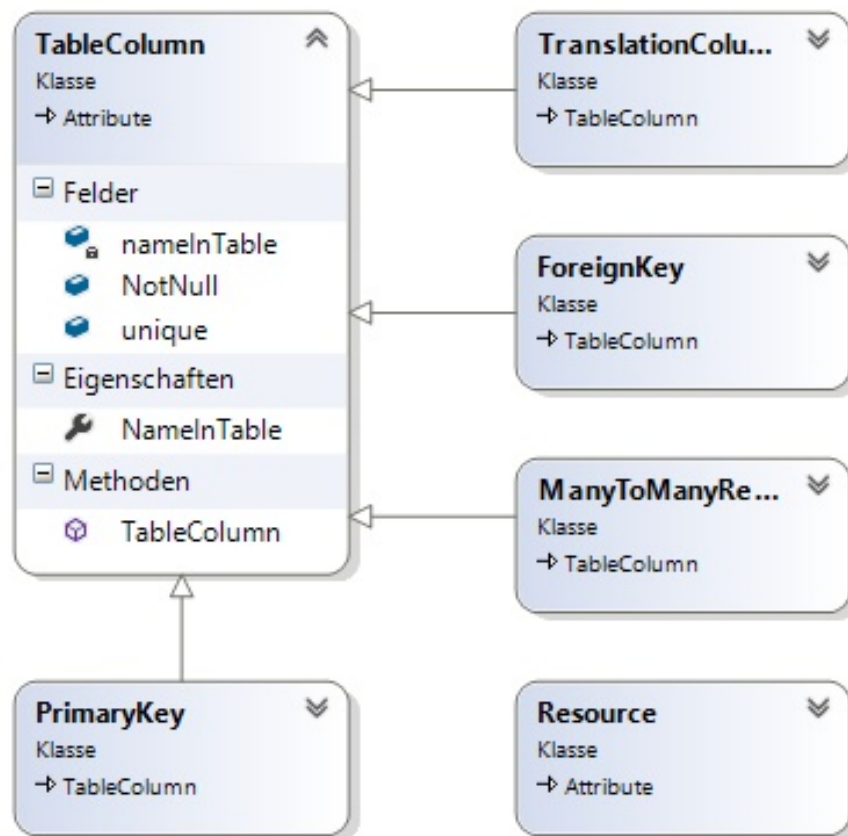
Abbildung 4.9: Many to Many Relation

```
84  
85     [ManyToManyRelation(  
86         "name",  
87         "patient",  
88         "patient_id",  
89         "patient_joints",  
90         "patientjoint_id",  
91         "patientjoint",  
92         "id"  
93     )]  
94     2 Verweise | Michael Jünger, vor 18 Tagen | 1 Autor, 2 Änderungen  
95     public Dictionary<long, PatientJoint> Joints  
96     {  
97         get  
98         {  
99             return this.joints;  
100        }  
101        set  
102        {  
103            this.joints = value;  
104        }  
105    }
```

- TranslationColumn: Da die Anwendung mehrsprachig sein sollte, wurden in der Datenbank Spalten für die jeweilige Sprache angelegt. Die Benennung folgt dabei dem Muster SPALTENNAME_LÄNDERCODE. Am Beispiel einer Übung wäre das folglich name_de_de für Deutschland und name_en_gb für Großbritannien. Ist eine Eigenschaft eines Modells als TranslationColumn gekennzeichnet, lädt das ORM entsprechend der eingestellten Sprache automatisch den Wert aus der jeweiligen Spalte in das Objekt. Im Programmcode muss auf diese Weise nicht weiter nach Sprache unterschieden werden, da immer auf die selbe Eigenschaft zugegriffen wird und sich nur der gespeicherte Wert ändert.
- Resource: Sie ist ein eigenständiger Typ und nicht von TableColumn abgeleitet. Dies war notwendig, da aufgrund der Vererbung alle von TableColumn abgeleiteten Klassen vom Reflection Framework ebenfalls als TableColumn erkannt werden, was auf die Hierarchie der Vererbung zurückzuführen ist. In der Anwendung ist eine Eigenschaft vom Typ Resource aber allgemein und

kann entweder eine Audio-Datei, ein Bild oder Video sein. Es gibt jedoch mehrsprachige Ressourcen wie eine Audio-Ressource für einen Menüeintrag, oder, wie im Falle des Titelbilds für eine Übung, Ressourcen die nicht mehrsprachig sind. Handelt es sich nun um eine mehrsprachige Ressource, müssen beide Annotationen auf die Eigenschaft angewendet werden. Erben nun beide vom gleichen Basistyp, kommt es dabei zu einem Konflikt, da beide als TableColumn identifiziert werden und dadurch nicht mehr unterscheidbar sind. Die einfachste Lösung hierfür war, die Resource als eigenständigen Typen zu behandeln. Anhand des Datentyps der Eigenschaft ist das ORM in der Lage zu entscheiden, ob es die Ressource als Bild, Video oder Audiodatei laden muss.

Abbildung 4.10: Annotationen



4.4 Reha Game Database Editor

Um die Datenbank-Struktur zu erstellen und mit den entsprechenden Daten zu befüllen, wurde eigens der RGDDBE entwickelt. Dabei handelt es sich um eine Internetseite, mithilfe der die benötigten Daten über eine grafische Oberfläche eingegeben und anschließend in die für Unity passende Ordnerstruktur exportiert werden

können. Die Entwicklung der Internetseite war nicht Teil der vorliegenden Arbeit, soll jedoch der Vollständigkeit halber erwähnt werden. Als Webframework wurde Django verwendet, welches ein eigenes ORM mitliefert und zugleich SQLite unterstützt.

Abbildung 4.11: Neue Übung erstellen

Übung bearbeiten

Felder, die mit einem Stern (*) markiert sind, sind Pflichtfelder

Name des Objekts in Unity *:

Der exakte Name in camelCase Notation des Objekts in Unity!

Zugewiesener Tag in Unity:

Der exakte Name des Tags, der diesem Objekt in Unity zugewiesen ist!

Name (Deutsch) *:

Name (Englisch):

Auditiver Name (Deutsch) *:

Datei auswählen | Keine ausgewählt

Auditiver Name (Englisch):

Datei auswählen | Keine ausgewählt

Vorschaubild *:

Datei auswählen | Keine ausgewählt

Schwierigkeitsgrad *:

leicht

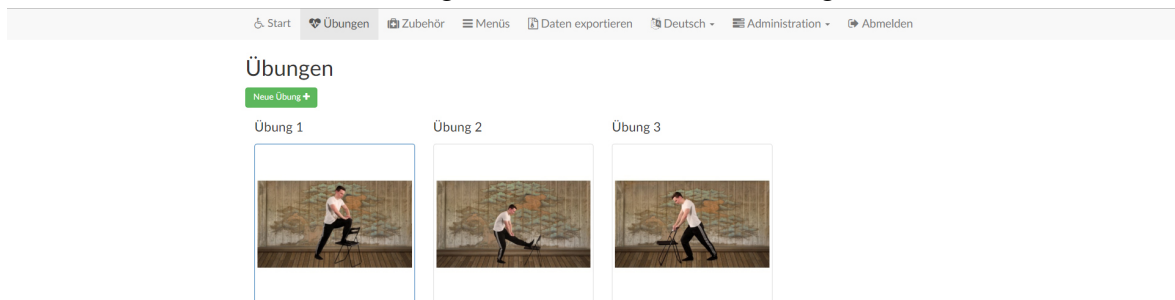
Video (Deutsch) *:

Datei auswählen | Keine ausgewählt

Video (Englisch):

Über die Eingabemaske werden alle benötigten Felder befüllt.

Abbildung 4.12: Übersicht der Übungen



<https://rgbo.mju.de/showExercise/exercise1>

Über die Übersicht kann die Eingabemaske für eine neue Übung aufgerufen werden oder eine bestehende bearbeitet und gelöscht werden.

5. Visuelles und auditives Designkonzept

Da es sich um eine medizinische Anwendung handelt, galt es, den Zugang möglichst barrierefrei zu gestalten. Aus technischer Sicht ist dies jedoch nicht vollumfänglich realisierbar. Zum einen gibt es ungelöste technische Probleme mit bei Steuerung der Anwendung, bei eingeschränkter Mobilität der oberen Extremitäten. Zwar wäre es technisch möglich, beispielsweise eine Fußsteuerung zur Verfügung zu stellen. Diese wäre jedoch vergleichsweise teuer, was dem Ziel - ein System zur Verfügung zu stellen - dass sich eine breite Bevölkerungsgruppe leisten kann, entgegenwirkt. Gleichzeitig deckt eine solche Steuerung lediglich eine Teilmenge der Nutzer ab, denen dies mit den Händen nicht möglich ist. Folglich würde dies unweigerlich dazu führen, weitere Eingabemethoden zur Verfügung zu stellen. Darauf konnte auch im Hinblick auf die zeitliche Begrenzung keine Rücksicht genommen werden.

5.1 Farbgebung

Bei der Entwicklung des Prototyps wurde insbesondere auf eine klare Farbgebung mit scharfen Kontrasten geachtet. Der Anwender muss während der Übung Informationen vom Bildschirm ablesen können. Ist er dabei durch die Übung angestrengt, soll er nicht noch zusätzlich belastet werden, die eingeblendeten Texte zu erkennen. Doch auch für Personen mit einer Sehschwäche ist die Lesbarkeit dadurch erhöht.

5.1.1 Farben in Menüs

Die Hintergründe der Menüs sind milchig-transparent, die Textfarbe ist schwarz. Dadurch entsteht ein hoher Kontrast zwischen Hintergrund und Text, was zu einer besseren Lesbarkeit der Texte führt.

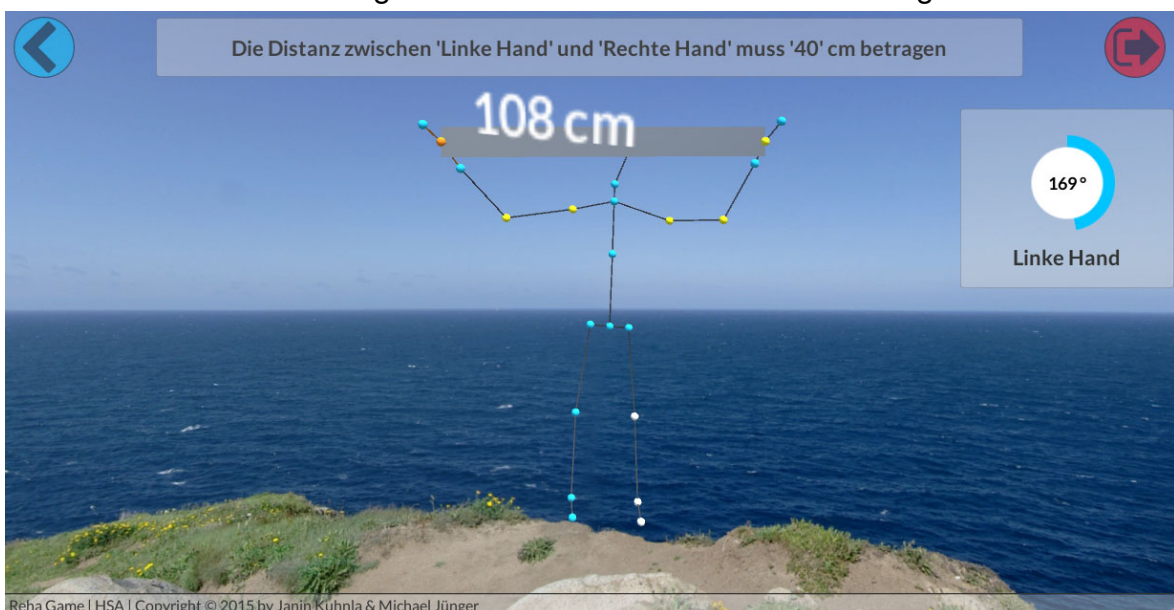
Abbildung 5.1: Das Einstellungs Menü



5.1.2 Farben in der Übung

Die Gelenke des Benutzers werden als Kugeln in unterschiedlichen Farben dargestellt. Die Grundfarbe ist türkis, die in der Übung beanspruchten Gelenke sind gelb, das aktive orange und nicht vorhandene sind weiß. Da grau als Farbe für die Knochen vorkommt, wurde auf die Verwendung von rot und grün an dieser Stelle gänzlich verzichtet, um Unklarheiten bei einer vorhandenen Rot-Grün-Schwäche vorzubeugen.

Abbildung 5.2: Farben der Gelenke in der Übung



5.2 Auditives Feedback

Neben den grafischen Elementen gibt die Anwendung zusätzlich Feedback in Form von Ton wieder. Dies ist vor allem dann sinnvoll, wenn eine Sehschwäche beim Anwender vorliegt und die Texte nicht selbst gelesen werden können.

Die auditiven Elemente der Anwendung sind:

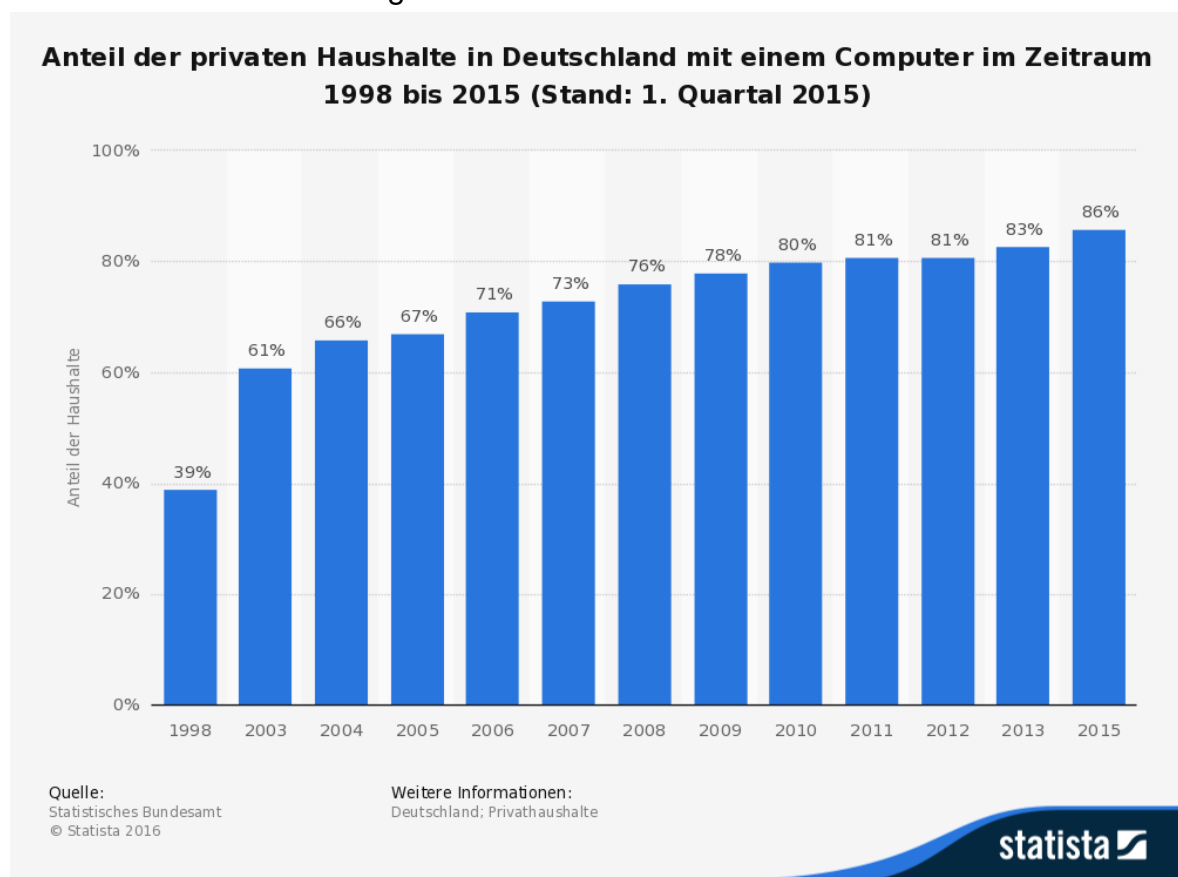
- Menüs: Der Menütitel beim Eintreten in das Menü, Menüpunkte sowie Eingabefelder bei Auswahl.
- Übungsauswahl: Name der Übung wenn diese ausgewählt wird.
- Detailansicht der Übung: Übungsbeschreibung und wichtige Informationen zur Übung.
- Übungsausführung: Bei einer Fehlstellung des aktiven Gelenks ertönt ein hoher Signalton, der tiefer und leiser wird, je näher der Anwender sich in der Idealposition befindet. Wurde die Idealstellung erreicht, ertönt ein positiver Quittingston.

6. Wirtschaftliches Konzept

Die Anwendung sollte von Beginn an ein System sein, dass sich eine große Mehrheit finanziell leisten kann. Somit war es das Ziel, zum einen die Kosten möglichst gering zu halten und zum anderen, Komponenten zu verwenden, welche über die Reha hinaus Anwendung finden können.

Ein PC ist in nahezu jedem Haushalt vorzufinden, es wird allerdings ein aktueller Rechner mit einer USB 3.0 Schnittstelle benötigt. Laut Umfragen besaßen im Jahre 2015 über 85 Prozent der Haushalte in Deutschland einen PC.

Abbildung 6.1: Privathaushalte mit PC bis 2015



Alternativ zum PC ist jedoch auch eine Xbox einsetzbar. Ist allerdings bereits ein PC mit ausreichender Hardware vorhanden, ist dies die kostengünstigste Variante. Der Anwender benötigt neben der Software lediglich eine Kinect 2 und einen speziellen Adapter. Entscheidet sich der Anwender für eine Xbox, entfällt zwar der Zusatzadapter, jedoch kostet die Xbox entgegen dem Adapter einige hundert Euro mehr. Ist weder ein PC noch eine Xbox vorhanden, so ist die Anschaffung der Konsole die günstigere Variante, da diese entgegen eines geeigneten Rechners kostengünsti-

ger ist. Zum Vergleich: Eine Xbox One kostet derzeit zwischen 250 bis 300 Euro. Ein PC oder Laptop mit ausreichender Hardware schlägt mit etwa dem doppelten zu Buche.

Neben den Kosten für die Anschaffung solle jedoch – so weit möglich – auch der Nutzen des Systems nach Abschluss der Reha in Betracht gezogen werden. Steht eine komplette Neuanschaffung der Hardware an, ist der zwar teurere aber vielseitiger einsetzbare PC womöglich dennoch die bessere Wahl, da eine Xbox lediglich als Spielekonsole einsetzbar ist und ggf. nach den Reha-Maßnahmen keine weitere Verwendung mehr findet.

Aus diesen Überlegungen lassen sich grundsätzlich zwei Gruppen potentieller Kunden ableiten. Privathaushalte und medizinische Einrichtungen. In beiden Fällen wäre der Verkauf von Zusatzprodukten, wie etwa weiteren Übungen, denkbar. Diese könnten, analog zu den sogenannten Inapp-Käufen bei Apps für Smartphones, über einen Store zum Download angeboten werden.

6.1 Privathaushalte

In Privathaushalten wurden die Grundvoraussetzungen zum Einsatz des Systems bereits geschaffen. Mögliche Anwender haben mit hoher Wahrscheinlichkeit schon die notwendige Hardware. Ihnen fehlt lediglich die Software mit den entsprechenden Übungen und die Kinect. Beides können sie direkt im Handel erwerben.

Für Privathaushalte ist auch eine Verleihung der Hardware in Kooperation mit Reha-Zentren denkbar. Für eine temporär andauernde Reha-Maßnahme müssen so vor allem ältere Patienten nicht die gesamten Kosten für die Hardware aufwenden, für die sie ansonsten keine Verwendung mehr haben. Neben ökonomischen Aspekten wäre dies zudem die ökologisch sinnvollste Variante. Auf diese Weise können auch jene Kunden erreicht werden, für die eine Anschaffung des Systems aus wirtschaftlichen Gründen nicht in Frage kommt.

6.2 Medizinische Einrichtungen

Diese können sowohl als Kunde wie auch als Vertriebspartner angesehen werden. Zum einen gelten sie als potentielle Anbieter des Systems in ihrer Einrichtung und sind somit Kunden, da sie das Produkt dazu erwerben müssen. Speziell im Falle von Reha-Einrichtungen können sie aber auch Vertriebspartner sein, da sie das Produkt dem Kunden näher bringen.

Als Kunden wäre es für medizinische Einrichtungen auch sinnvoll, entgeltlich Serviceverträge abzuschließen und Schulungen zum Produkt anzubieten.

7. Schlussteil

Die Arbeit hat das Genre der Exergames näher beleuchtet. Es wurde aufgezeigt, welche vielfältigen Einsatzmöglichkeiten es für einen digitalen Assistenten für Reha-Anwendungen gibt, sei es Zuhause oder in einer medizinischen Einrichtung. Die Kosten für ein solches System sind vergleichsweise gering. Somit handelt es sich nicht um ein utopisches Zukunftsprodukt, denn es könnte bereits heute produziert werden. Auch wurde ein möglicher gesellschaftlicher Nutzen aufgezeigt, den ein solches System mit sich bringen kann. Die Gesellschaft ist durch die steigende Lebenserwartung im Wandel. Gegen die mit dem Altern verbundenen körperlichen Gebrechen ist dennoch niemand gewappnet. Die Anwendung kann an dieser Stelle ansetzen und sowohl die medizinischen Einrichtungen, deren Personal sowie die Krankenkassen und nicht zuletzt die Patienten entlasten.

Der vorliegende Prototyp bietet dazu bereits viel Potential, wenngleich er auch noch nicht ausgereift ist. Zwar ist es schon jetzt möglich, einzelne Übungen zu absolvieren, für ein ausgereiftes Übungsprogramm allerdings ist er noch nicht geeignet. Nach wie vor fehlt der Anwendung ein Modul zur Kalibrierung, in dem Patienten mit irreversiblen körperlichen Beeinträchtigungen den Soll-Zustand der Übung auf ihren persönlichen Ist-Zustand anpassen können. Dieses Feature wurde zwar vorbereitet, konnte jedoch noch nicht abgeschlossen werden.

Weiterhin werden die Übungen noch nicht gefiltert. Jedem Benutzer können eine oder mehrere Anwendungen zugewiesen werden, was dann seinem individuellen Trainingsplan entsprechen würde. In der Übungsauswahl werden diese allerdings nicht berücksichtigt und stattdessen alle Übungen angezeigt. Auch wird eine Übung, die ein bestimmtes Gelenk voraussetzt, noch nicht aus der Übungsauswahl entfernt, wenn dem Benutzer die entsprechende Gliedmaße fehlt.

Ein weiteres Feature, welches in einer kommenden Version nicht fehlen sollte, ist die Möglichkeit einen Benutzer per QR-Code zu erstellen. Im Reha-Zentrum könnte dem Patienten ein Schein ausgedruckt werden, ähnlich einem Rezept beim Arzt. Auf dem Schein befindet sich neben allgemeinen Angaben zum Reha-Zentrum lediglich der QR-Code. In ihm enthalten können die Stammdaten des Patienten sein, sowie etwaige fehlende Gliedmaße und Übungen, die er ausführen soll. Über die integrierte Kamera der Kinect könnte dieser schließlich eingelesen werden, woraufhin der Benutzer im System angelegt wird.

Da der Benutzer persönliche Daten wie Name, Alter und Geschlecht enthält, müssen für die Benutzung in medizinischen Einrichtungen Vorkehrungen zum Schutz dieser Daten getroffen werden. Hier haben unter Umständen mehrere Nutzer Zugang auf das System. Nach aktuellem Stand aber ist ein Benutzerprofil von jedermann einsehbar. Denkbar wäre hier die Verwendung eines Pseudonyms als Name, der öffentlich sichtbar ist. Die Aktivierung eines Nutzers ist dann nur durch die Eingabe einer PIN oder eines Passwortes möglich, welches vom Nutzer festgelegt werden kann. Insgesamt wären vor dem produktiven Einsatz in einer medizinischen Einrichtung die gesetzlichen Rahmenbedingungen zur Speicherung und Verarbeitung der Daten noch zu klären.

Der jedoch schwerwiegendste Kritikpunkt der Anwendung ist, dass die Kinect alleine nicht ausreichen wird, die Korrektheit der ausgeführten Teilschritte einer Übung gänzlich zu erfassen. Vor allem bei Gewichtsverlagerungen, welche die Position des Körpers nur minimal verändern, gibt es noch große Schwierigkeiten, diese Veränderung zu erfassen. Zudem ist die Kinect nur dazu ausgelegt den Körper Frontal zu erfassen. Liegt der Patient auf dem Boden oder dreht sich mit dem Rücken zur Kinect, kommt es zu einer erhöhten Ungenauigkeit bei den Messungen der Gelenke am Körper.

Wenngleich es noch einige fehlende Punkte im Prototypen gibt, so kann er dennoch als Basis für ein künftiges Produkt oder weitere Forschungen herangezogen werden. Nach aktuellem Stand der Technik muss allerdings über die Verwendung von zusätzlichen Eingabemethoden nachgedacht werden, um die Genauigkeit der Messergebnisse und damit die Qualität der Beurteilung über die Korrektheit der Übungsausführung zu erhöhen. Es soll schließlich eine Verbesserung der Physiologie am Patienten erreicht werden.

Danksagungen

Zum Schluss möchte ich mich bei allen Bedanken, die mich über die letzten drei Semester bei meiner Arbeit unterstützt haben.

Der besondere Danke geht dabei in erster Linie an das Trainer-Team von ProVita, die stets ein offenes Ohr für Fragen zu den Übungen hatten und mit Begeisterung über die Anwendung an einem Gelingen mitgewirkt haben und auch einen ihrer Trainer für die Aufnahme der Videos zur Übungsausführung zur Verfügung gestellt haben. Vielen Dank daher insbesondere an Patrick

Ebenso gebührt meinem betreuendem Professor - Herrn Müller - ein großes Dankeschön für konstruktive Betreuung und Vorschläge zur Anwendung, vor allem in Fragen zum Design. Ebenso nicht unerwähnt bleiben darf an dieser Stelle Herr Seifert, der neben Herrn Müller die Arbeit in technischen Fragen betreute. Auch hier ein großes Dankeschön.

Zuletzt danke ich noch allen Freunden und meiner Familie für die seelische Unterstützung und für das in Ruhe lassen und nicht beleidigt sein, wenn es einmal etwas stressiger wurde.

Literaturverzeichnis

- [Bassett 2015] BASSETT, Lindsay: *Introduction to JavaScript object notation*. O'Reilly Media, 2015. – ISBN 978-1-4919-2948-3
- [Dentler 2010] DENTLER, J.: *NHibernate 3.0 Cookbook: 70 Incredibly Powerful Recipes for Using the Full Spectrum of Solutions in the NHibernate Ecosystem*. Lightning Source, 2010 (Quick answers to common problems). – URL <https://books.google.de/books?id=zaV0mAEACAAJ>. – ISBN 9781849513043
- [Froeb 2011] FROEB, Maik S.: *Gleichgewichtstraining mit Nintendo® Wii fit™*. 2011. – URL http://home.uni-leipzig.de/lb/?wpfb_dl=301
- [Koch 2010] KOCH, Daniel: *XML für Webentwickler*. Hanser, 2010. – ISBN 978-3-446-42356-5
- [Mical 1987] MICAL, Robert J.: *Info Magazin - How We Created the AMIGA Computer*. 1987. – URL http://ia802708.us.archive.org/20/items/info-magazine-13/Info_Issue_13_1987_Jan-Feb.pdf
- [Viktor Wendel, Sandro Hardy, Stefan Göbel, Ralf Steinmetz 2010] VIKTOR WENDEL, SANDRO HARDY, STEFAN GÖBEL, RALF STEINMETZ: *Adaption und Personalisierung von Exergames*. 2010. – URL <ftp://ftp.kom.tu-darmstadt.de/papers/WHGS10.pdf>
- [Wikipedia 2014] WIKIPEDIA: *Wii Fit* — *Wikipedia, Die freie Enzyklopädie*. https://de.wikipedia.org/w/index.php?title=Wii_Fit&oldid=132293676. 2014. – [Online; Stand 11. Juli 2016]