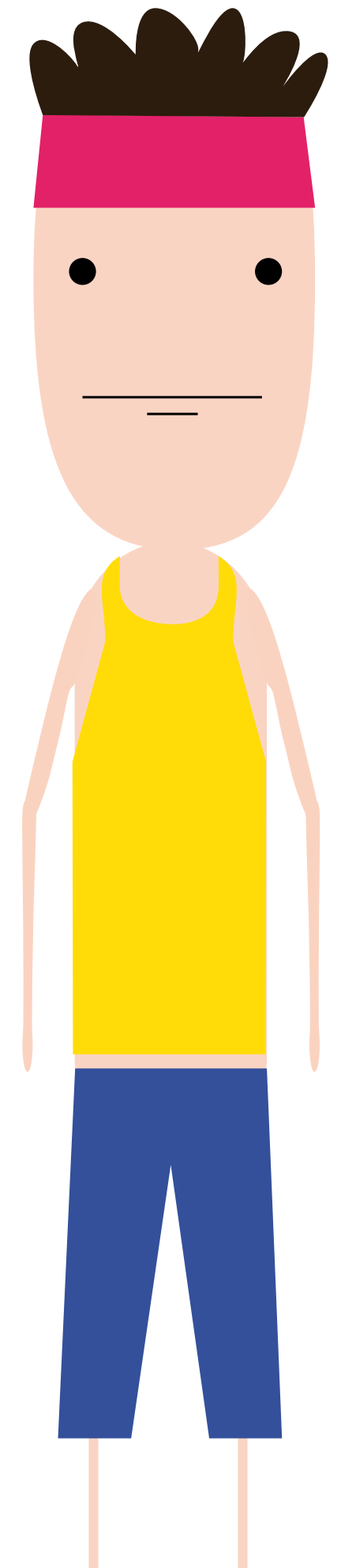


meet
WUTZ



Konzeption & Entwicklung

Carlos Niermeier 667410
Jannik Bussmann 667887
Sebastian Galetzka 667263

06

Konzept

12

Projektplanung

20

Applikation

38

Elektronik & Prototyp

50

Ausblick und Fazit

51

Credits

Inhalt

Konzept

Einführung

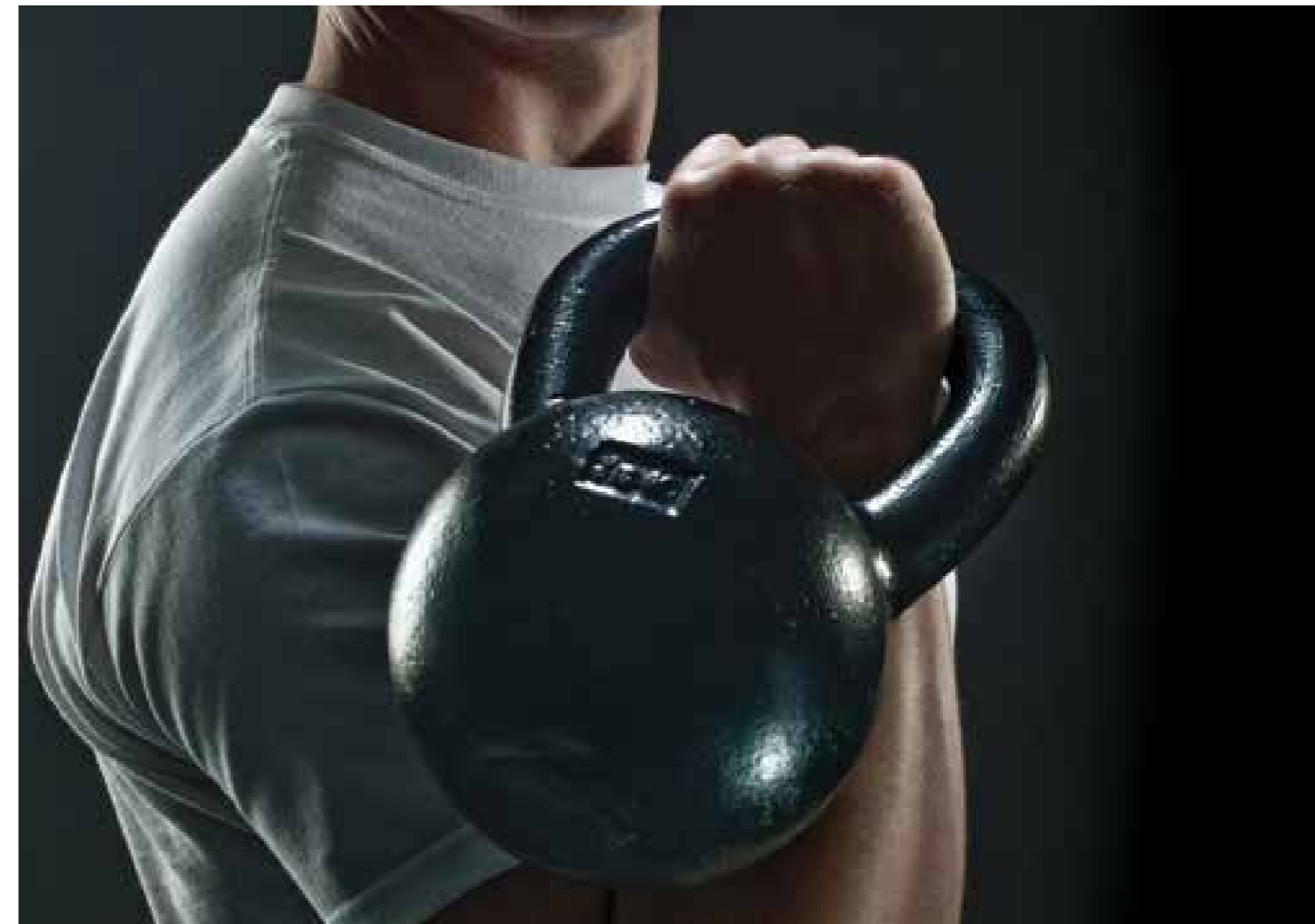
In den letzten Jahren ist vermehrt ein Hype um Kraftsport in der Gesellschaft und im Internet zu beobachten. Auf YouTube sehen wir Body-Transformationen, im Park schwitzen sich immer mehr Leute zum Traumkörper und bei Instagram veröffentlichen Weltstars Fotos und Videos von ihrem morgentlichen Workout. Wir wollten uns mit diesem populären Thema in dieser Hausarbeit auseinandersetzen. Bei WUTZ handelt es sich um einen Dienst, der Fitnessgeräte mit einer Applikation vernetzt und in Form von Minigames den Nutzer dazu animiert, sich zu bewegen.

Viele Menschen waren schon an dem Punkt, etwas an ihrem Körper verändern zu wollen. Man möchte dünner werden und/oder Muskeln aufbauen, um einem (persönlichen) Schönheitsideal zu entsprechen oder gesünder zu leben. Ebenso sind viele bereits an dem Punkt gewesen, zu scheitern, aufzugeben oder gar nicht erst anzufangen, ohne jegliche Strategien, um den inneren Schweinehund alleine überwinden zu können. Jedoch von anderen Leuten mit ähnlichen Problemen befragt, entwickelt man sich plötzlich zu einem regelrechten Lebenscoach. Wer hat ihn nicht, den einen Freund, der die besten Beziehungstipps parat hat, aber selbst immer Single ist oder fundiertes Wissen über Sport und Ernährung teilt, aber selbst lieber auf der Couch sitzt und Pizza isst. Es scheint vermeintlich einfacher zu sein, das Problem von außen (objektiv) zu betrachten und entsprechende Tipps zu geben, als selbst zu handeln. In der Psychologie spricht man in diesem Zusammenhang von der Projektion - dem Verfolgen eigener Wünsche in anderen.

So soll der Nutzer seine persönlichen Wünsche auf WUTZ, einen Avatar projizieren und ihn spielerisch coachen. Durch das Absolvieren der Minigames entwickelt sich der Avatar weiter. Dass der Coach dabei eigentlich an sich selbst arbeitet, rückt in den Hintergrund und sorgt für eine entspanntere Trainingsatmosphäre und soll auf diese Weise zu besseren Ergebnissen führen.

Name

WUTZ ist eine Abwandlung von dem englischen Begriff „wuss“, was so viel wie Schwächling bedeutet. Ursprünglich sollte der Begriff durch eine alternative Buchstabierung in „WUZZ“ umgewandelt werden – die Phonetik bliebe gleich und es hätte eine optische Nähe zu verwandten Spitznamen wie „Buzz“ (vgl. z. B. Buzz Lightyear) gegeben. Nach umfangreicher Recherche hatte sich jedoch ergeben, dass sich laut dem „Urban dictionary“ u. A. auch eine obszöne Wortbedeutung hinter dem Begriff wuzz verbirgt, sodass an dieser Stelle umgedacht wurde und WUTZ entstanden ist. WUTZ klingt „verdeutsch“ und orientiert sich an einem Trend, der bereits durch Dienste wie UBER Anwendung findet.



Zielgruppe

Das Spiel richtet sich an Männer, die an Mini-Games in Verbindung mit Fitness-Sport und neuen Interaktionstechnologien mit hohem Unterhaltungswert interessiert sind. Das Alter der Zielgruppe beschränkt sich primär auf Teenager und Erwachsene zwischen 14 und 35 Jahren, allerdings können auch ältere Menschen Motivation und Spaß an dieser App finden. Durch die Wahl eines englischen Titels und englischen Anleitungstexten innerhalb der App, ist die Zielgruppe global und beschränkt sich nicht nur auf Deutschland. Durch die intuitive Benutzerführung mittels Körper und Armbewegungen, werden keine großen Vorkenntnisse in der Bedienung benötigt, was den großen Vorteil hat, dass eine große Zielgruppe verschiedenen Alters erreicht werden kann. Die Spiele werden vor dem Spielbeginn verständlich für den Nutzer erklärt, sodass eine korrekte Spielweise gegeben ist. Für die Benutzung wird ein Smartphone oder ein Tablet mit einem Android Betriebssystem vorausgesetzt.



**Projekt-
planung**

Projektplanung

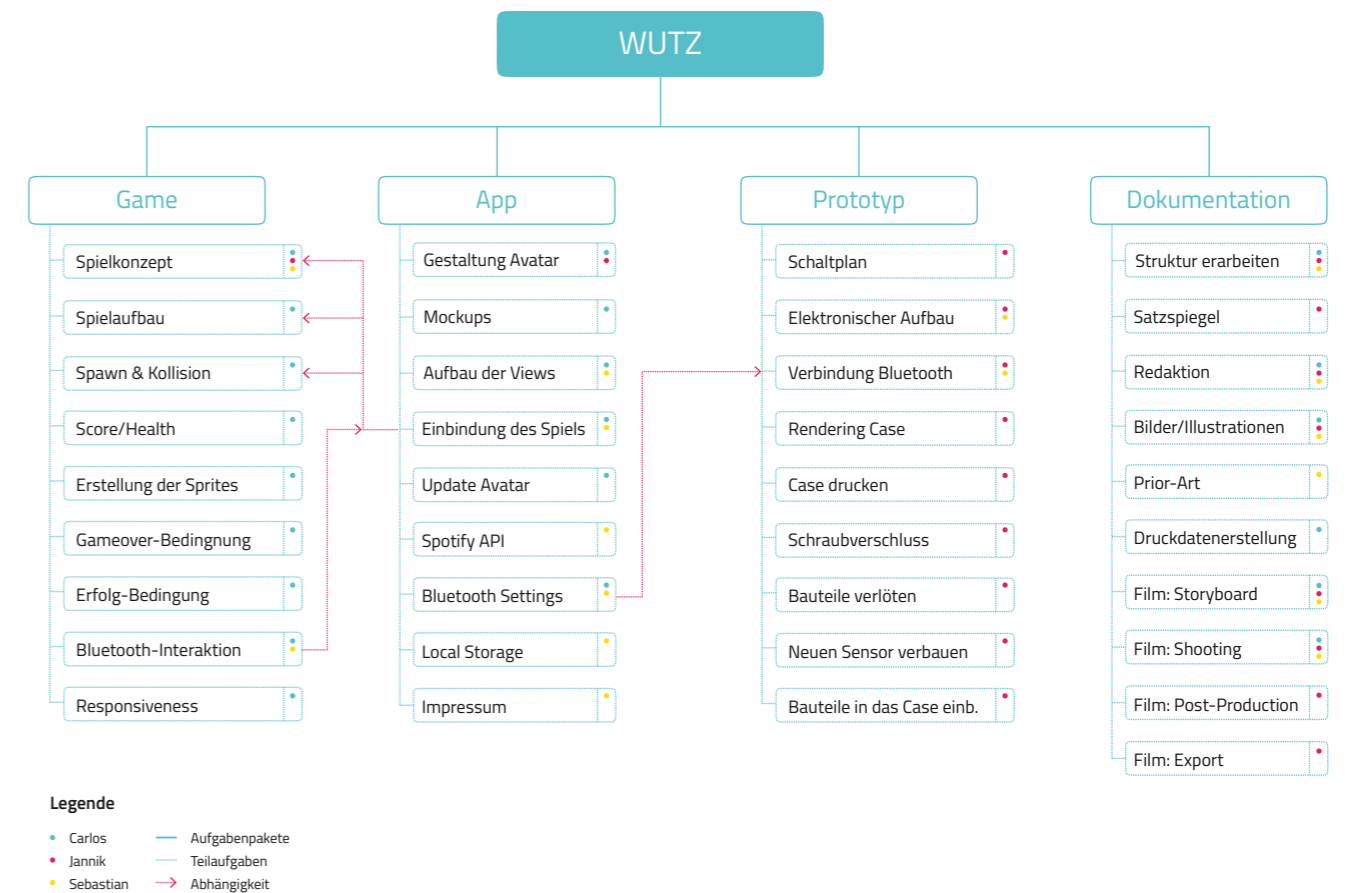
Am Anfang eines jeden Projekts steht die Projektplanung. Aufgabenpakete müssen definiert und Ressourcen aufgabenangemessen eingeplant werden.

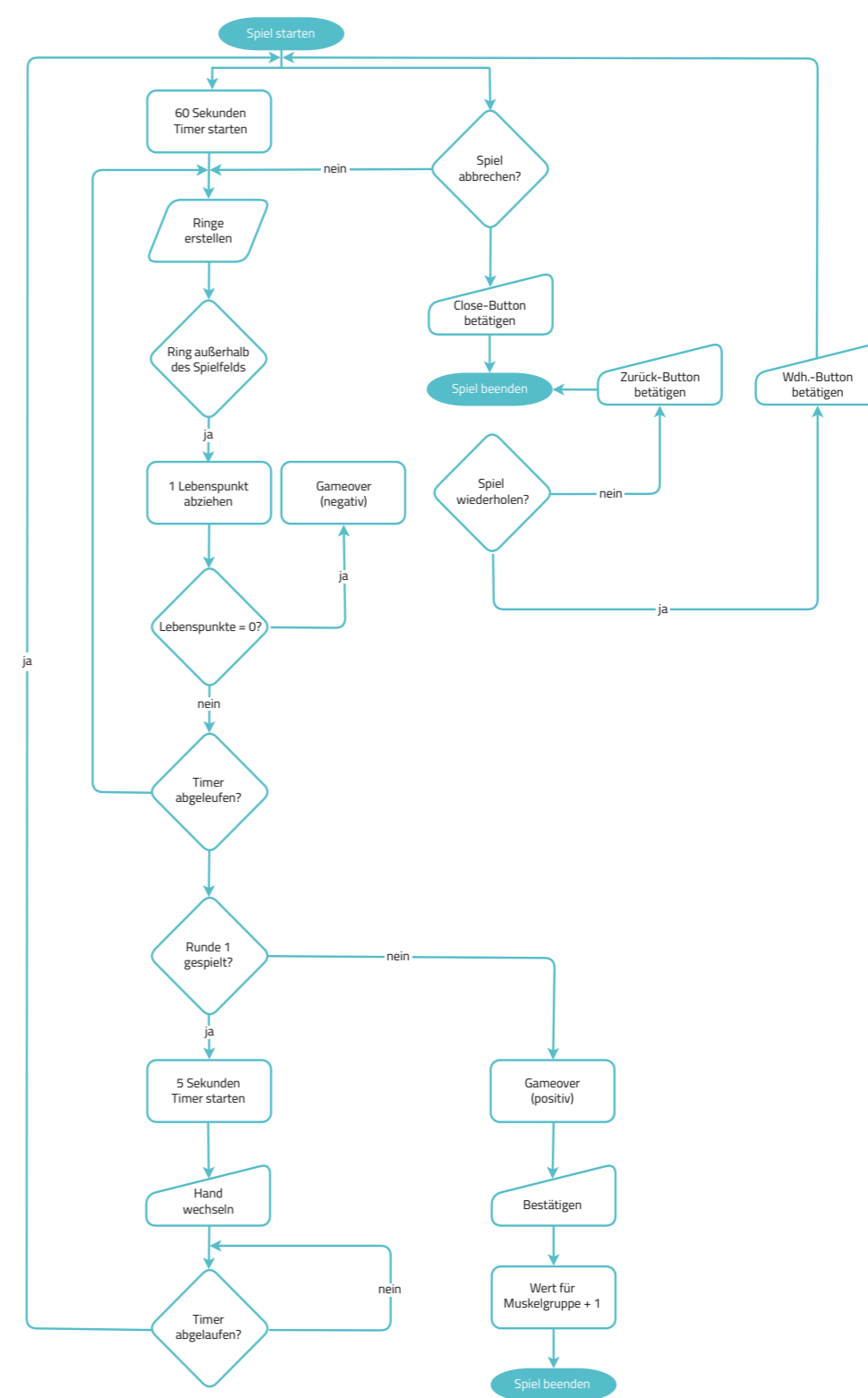
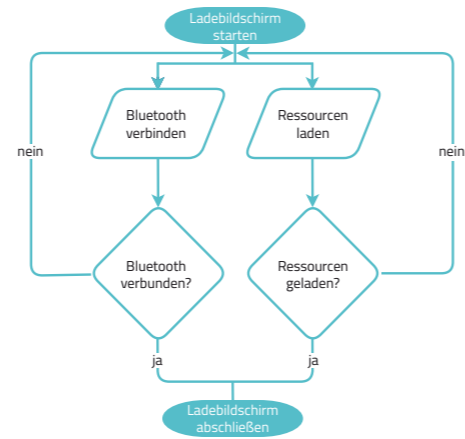
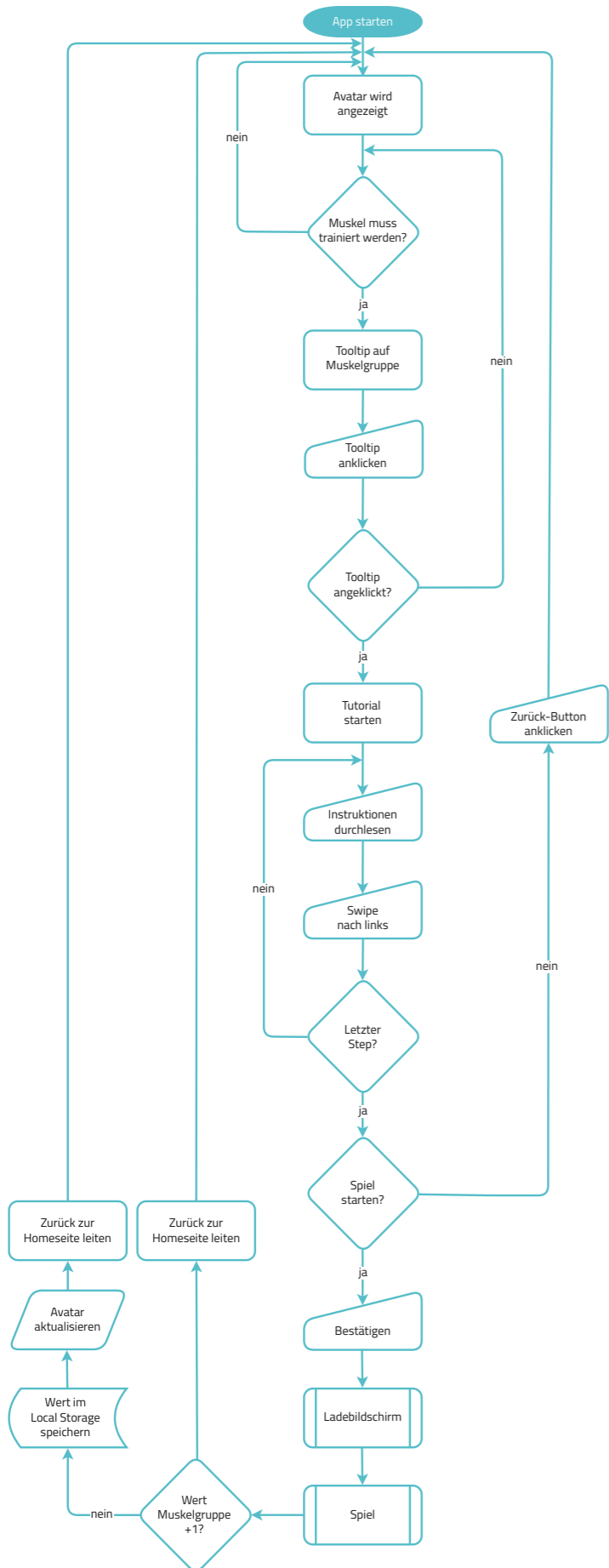
In diesem Semester haben wir uns mit verschiedensten Methoden zum Strukturieren und Planen von Projekten auseinandergesetzt, die im Rahmen des Semesterprojekts Anwendung fanden. Initial wurde ein Projektplan erstellt, der einen groben Rahmen für das Projekt vorgab. Grob in der Hinsicht, als dass ein solcher Projektplan ein lebendiges Dokument ist, das sich im Projektverlauf aufgrund verschiedener Einflüsse – internen wie externen Ursprungs – stets verändert und erweitert. Grundsätzlich ist jedes Projekt individuell.

Neue Aufgabenpakete kommen im Entwicklungsprozess hinzu, die zuvor gar nicht absehbar waren, andere werden unter Umständen verworfen.

Es gibt kaum vordefinierte Schablonen sodass bei der anfänglichen Erstellung eines Projektplans von greifbaren Parametern wie Start- und Endzeitpunkt sowie allgemeinen Arbeitspaketen unter Berücksichtigung von Pufferzeiten ausgegangen werden muss. Nachdem das Thema erschlossen und sich auf ein konkretes Konzept geeinigt werden konnte, wurde auf Basis dessen ein Strukturplan erstellt, der alle für die Umsetzung relevanten Arbeitspakete, Teilaufgaben und Verantwortlichkeiten enthielt, um die noch verbleibende kurze Zeit möglichst produktiv und effizient zu nutzen. Unterstützend wurde ein Flowchart erstellt, um für die Erstellung der App und des Spiels relevante Teilaufgaben zu identifizieren.

Diese wurden darüberhinaus auf einem Whiteboard festgehalten, nach erfolgreicher Bearbeitung durchgestrichen und während des Entwicklungsprozesses um weitere Teilaufgaben erweitert. Abgesehen von der Tatsache, dass so jeder Projektteilnehmer den aktuellen Stand stets im Blick behalten konnte, hatte diese Vorgehensweise auch einen äußerst motivierenden Effekt, da Projektfortschritte direkt und anschaulich sichtbar wurden.





Prior art

Im Zuge der Projektentwicklung wurde selbstverständlich auch recherchiert, ob ähnliche Projekte existieren, die Fitnessgeräte mit Sensorik ausstatten, um diese noch stärker auf die Bedürfnisse von Nutzern zuzuschneiden. Drei dieser Projekte werden im Folgenden kurz vorgestellt.

Push



Push ist ein Fitness-Tracker, der die Ausübung verschiedener Kraftsport-Übungen tracken kann. Push wird von mehr als 50 professionellen Teams in den USA genutzt. Den Tracker bindet sich der Athlet entweder um den Oberschenkel oder trägt ihn am Arm. Der Tracker arbeitet mit einem Accelerometer und einem Gyroscope, um die richtige Ausübung zu kontrollieren. Die Daten werden dann über Bluetooth an die App übertragen. Betrieben wird das Wearable über einen Akku. Ebenso kann man durch die verbauten LEDs den Status des Gerätes jederzeit überprüfen. Die App ist eine Datenbank der ausführbaren Übungen. Die Bewegungen werden analysiert und als Ergebnis in der App gespeichert. Eine Option zum sozialen Teilen ist ebenfalls vorhanden. Ein weiteres Extra zum Motivieren ist das PUSH Portal. Darin werden Bestenlisten zu den verschiedensten Übungen aufgeführt.

Amiigo



Amiigo erfasst sensorisch die technisch richtige Ausführung einer Übung sowie den Gesundheitszustand des Körpers. Den Sensor gibt es in 2 Ausführungen mit unterschiedlichen Ausstattungen. **Armband:** Mit dieser Version wird der obere Körper getrackt. Unter anderem wird dies mit einem 3-Achsen-Accelerometer, Temperatur Sensor und einem Pulsoximeter erreicht. **Schuh-Clip:** Die Beinbewegung wird mit einem kleineren Sensor gemessen. Dieser hat ebenso einen 3-Achsen-Accelerometer und einen Temperatur Sensor.

Die Daten werden von beiden Sensoren mittels Bluetooth an die App übertragen. Die App überwacht und speichert Übungen aus drei großen Sportbereichen: dem Schwimmen, Radfahren und beinahe alle Übungen, die man im Fitnessstudio trainieren kann. Dabei wird eine Übersicht angelegt mit allen relevanten Daten: verbrannte Kalorien, Puls, Hauttemperatur, Blutsauerstoffsättigung, Atemfrequenz usw.

Beast

Beast erfasst Daten zur Verbesserung von Übungen im Krafraum. Als Sensoren kommen hier folgende zum Einsatz: Accelerometer, Gyroscope und Kompass. Die Datenübertragung erfolgt über Bluetooth. Der Sensor wird wahlweise entweder am Körper getragen (Arm, Bein oder Rückenbereich) oder durch die magnetische Funktion des Gerätes direkt an dem Trainingsgerät angebracht. Die Übungen werden in der App gespeichert und analysiert. Pro Übung gibt es einen app-internen Score. Dieser setzt sich aus mehreren Parametern einer Übung zusammen, wie z.B. die Kraft oder die Geschwindigkeit. Während man nach einer Übung eine Zusammenfassung vor sich hat, kann man gleichzeitig zur Übung den Live-Monitor aktivieren und ein visuelles Live-Feedback erhalten. Zusätzlich ermöglicht die App dem Nutzer, seinen Trainingserfolg auf seine persönlichen Bedürfnisse auszurichten. So kann er wählen, ob sich das Workout z. B. auf Kraftsteigerung, Muskelwachstum oder Fettverbrennung ausrichten soll.



Fazit

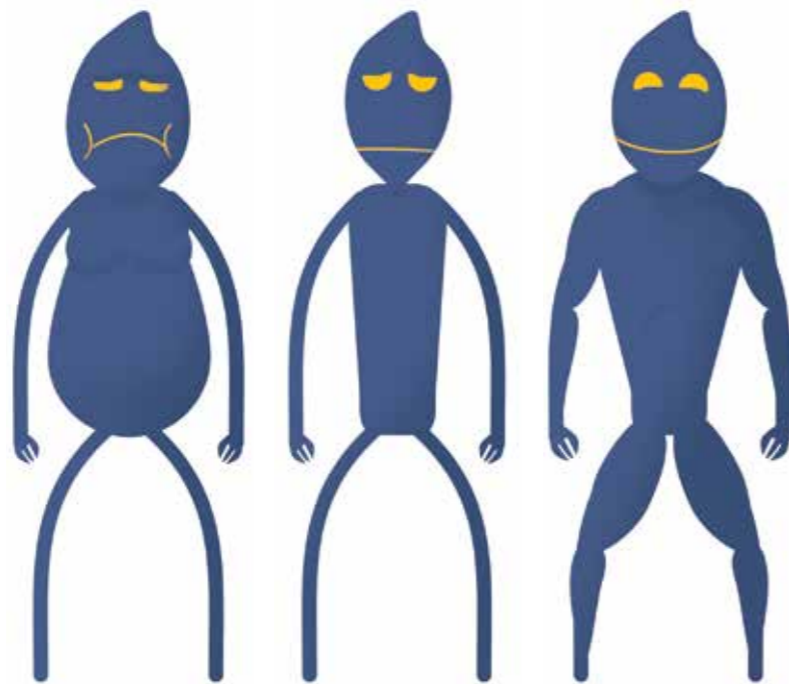
Der Markt bietet viele unterschiedliche Arten von Sensorik, die sich damit auseinandersetzt, das Training von Menschen, die regelmäßig Sport treiben, zu erfassen, auszuwerten und zu optimieren. Die meisten davon werden am Handgelenk getragen. Beast stellt dabei eine Ausnahme dar, da es sich über Magneten zusätzlich auch an den Trainingsgeräten selbst anbringen lässt, ähnlich wie WUTZ. Auffällig ist, dass die verbaute Sensorik sowie die Datenübertragung die gleiche ist, die auch bei WUTZ verbaut ist. WUTZ zeichnet sich allerdings im direkten Vergleich vor allem dadurch aus, dass es sich in erster Linie an Sportnovizen richtet, die normalerweise nicht trainieren. Es geht nicht darum, das Training zu optimieren, sondern Menschen dazu zu motivieren, sich überhaupt sportlich zu betätigen. Die verbaute Sensorik dient hier nicht der Erfassung von qualitativen Daten aus dem Workout, sondern ermöglicht es dem Trainingsgegenstand, die Funktion eines Eingabegerätes einzunehmen und die Interaktion mit dem selbigen um eine Ebene zu erweitern.

Applikation

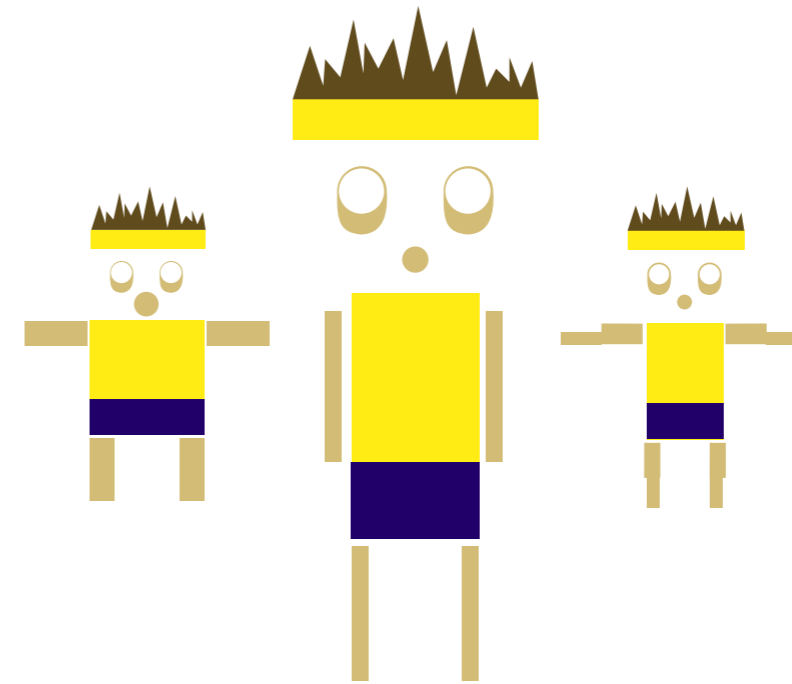
Avatar

Als einen zentralen Gestaltungspunkt der App ist der zu trainierende Charakter anzusehen. Bei der Gestaltung wurde auf eine minimale und abstrakte Ästhetik wert gelegt, die dem Cartoon-Genre zuzuordnen ist. Die auffällige Form und Farbgebung soll dem Spieler ein humoristisches Feeling vermitteln, um das Spielvergnügen visuell zu transportieren.

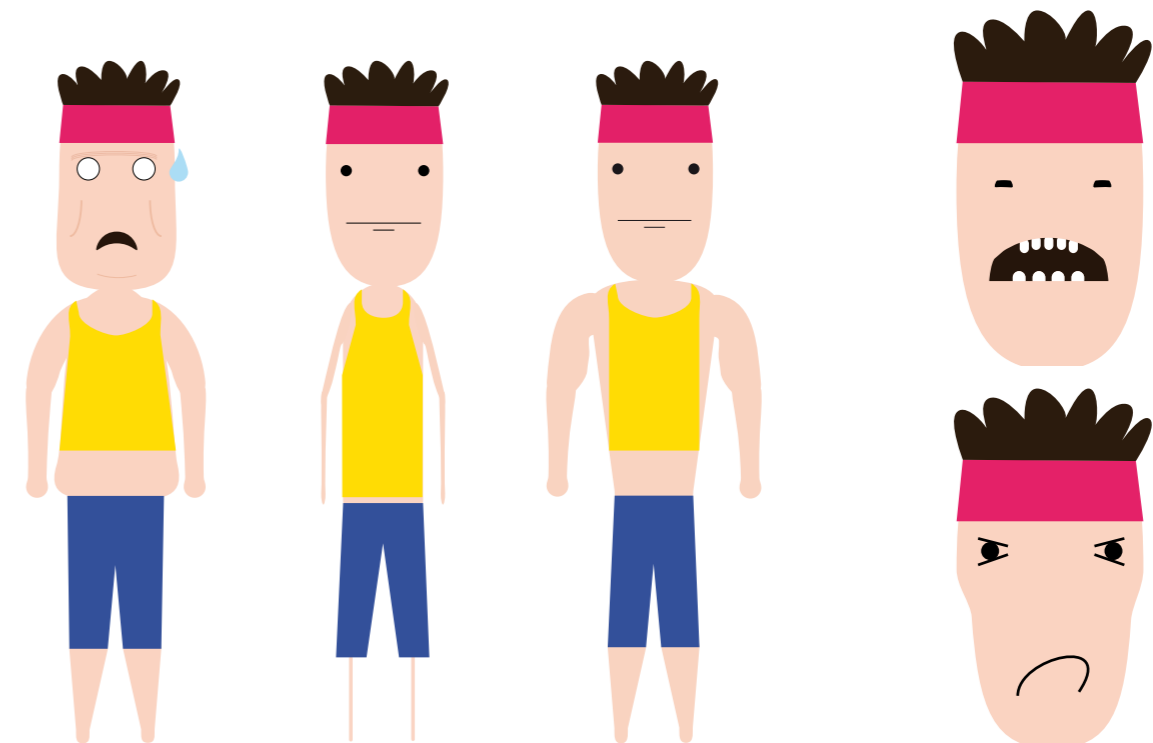
Bei der Gestaltung des Avatars wurde von Anfang an Wert darauf gelegt, eine Visualität zu schaffen, die modellierbar ist. Schließlich sollte der Avatar unterschiedliche Körperproportionen und Emotionen darstellen können. In der ersten Entwurfsphase wurde versucht, ein Wesen zu kreieren, das vermenschlicht, aber kein Mensch ist. Der initiale Gedanke dahinter war, eine geschlechterübergreifende Zielgruppe anzusprechen. Da die Ergebnisse nicht wirklich zufriedenstellend waren, und sich sowohl konzeptionell (insbesondere im Hinblick auf die Zielgruppe) als auch visuell immer mehr eine maskuline Form herauskristallisierte, wurde in folgenden Entwurfsschritten der Fokus auf einen Menschen gelegt. Dieser zeichnet sich durch eine reduzierte Formgebung und knalligen, flachen Farben aus. Auch die Mimik besteht weitestgehend aus sehr einfachen Linien und Flächen, die den Zustand des Avatars auf einfache und humoristische Art und Weise transportieren.



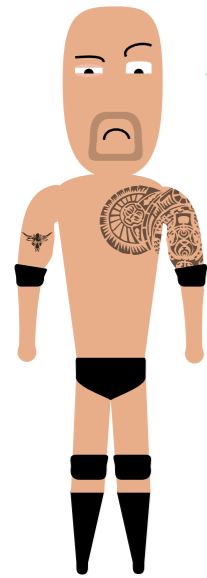
Erste Entwurfsphase.



Gegenentwurf mit einem geometrischeren Ansatz.



Finaler Entwurf.



The Rock wants your little Candy-Ass to just bring it!
– The Rock –



Dann mach deinen Scheiß doch selber, du A@\$ç}{[]ç}!
– Klaus Kinski –



Get over heeeeeere!
– Scorpion –



It's morphing time!
– Power Rangers –



Whatcha gonna do when calories hit you, brother?
– Hulk Hogan –



De workaut is derminaded.
– TX-800 –



Am I gonna die or am I stronger than ever?
– Stan Smith –

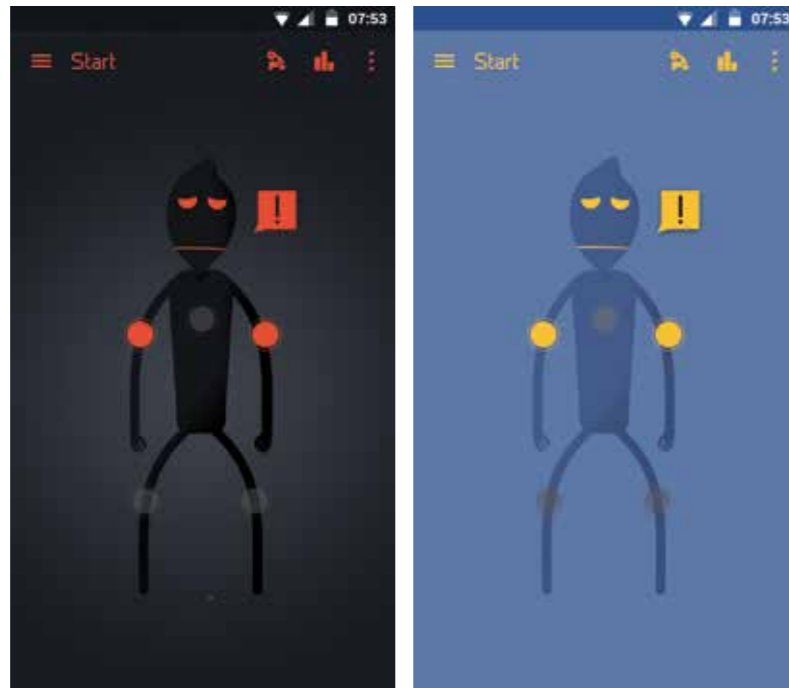


I find your lack of determination disturbing.
– Darth Vader –

Eine Auswahl prominenter Skins, die als Bonus-Content angedacht, aber im Laufe der Umsetzung verworfen wurden.

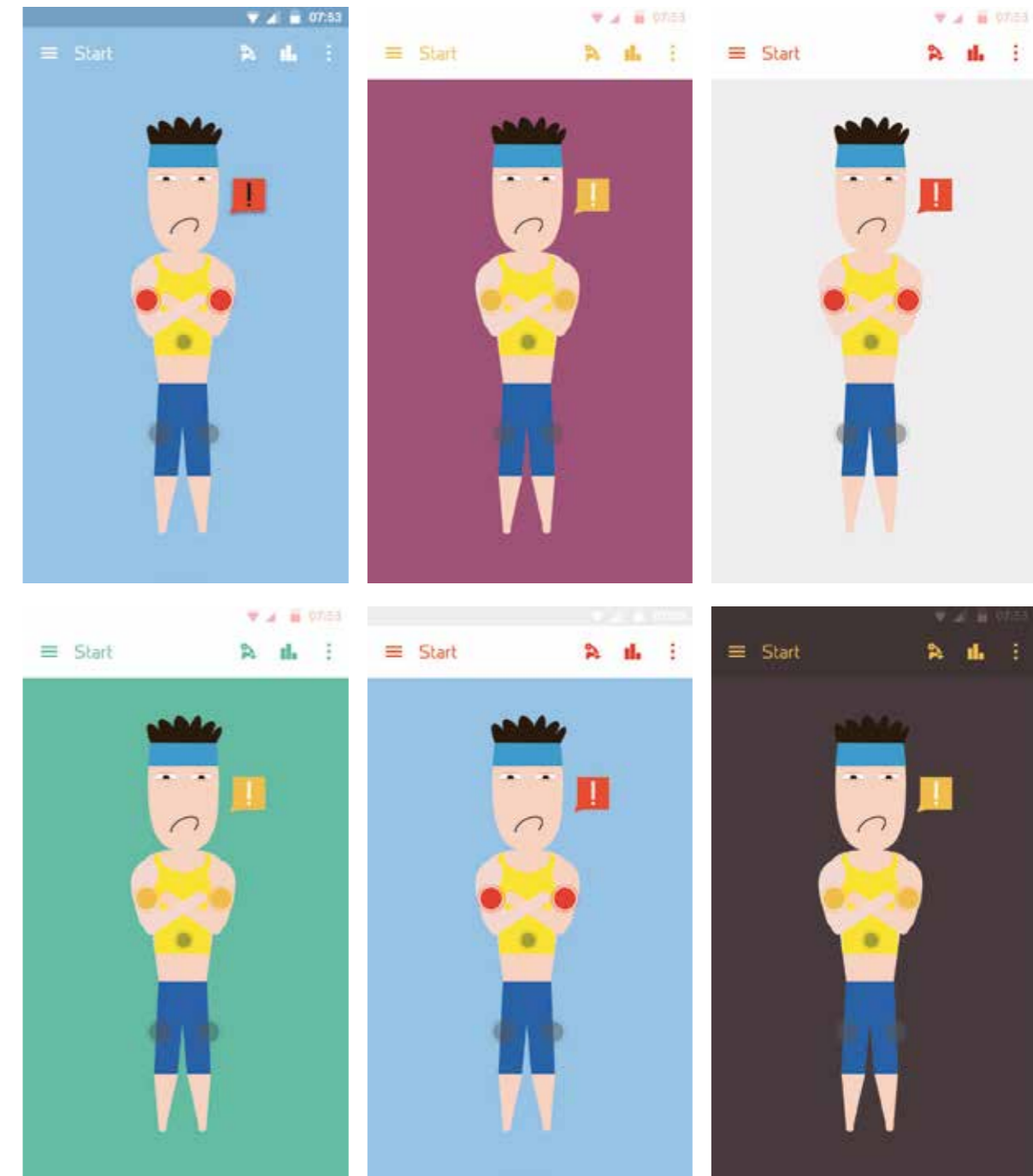
Aufbau der App

Parallel zum Avatar wurde auch die App in einem iterativen Gestaltungsprozess entwickelt. Dazu wurden im ersten Schritt Mockups erstellt, deren Basis der Anfangsscreen mit dem Avatar war.



Mockup Erste Entwurfsphase

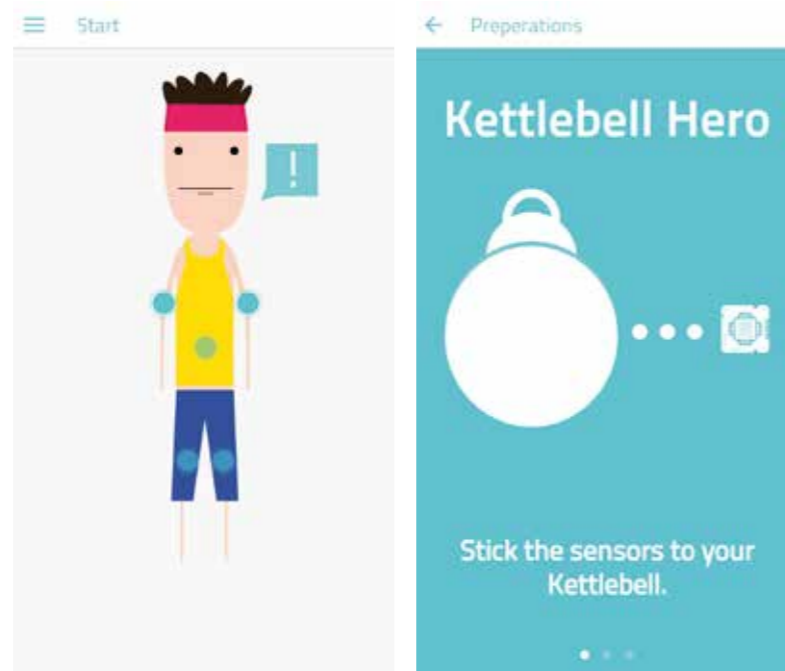
Nachdem die Formfindung des Avatars vollendet war, ging es im nächsten Schritt darum, ein stimmiges Farbkonzept zu finden. Dazu wurde ein Exposé mit verschiedenen Farbvarianten erstellt, im Team diskutiert, bewertet und die finale Farbgebung ausgearbeitet. Im Zuge der Entwicklung wurde das Mockup weiter reduziert.



Ein Exposé verschiedener Farbvarianten.

Der Aufbau der App ist sehr minimalistisch gehalten worden, um den Zweck der App in den Vordergrund zu rücken. Nach dem Öffnen der App befindet man sich auf dem Start- bzw Avatar-Bildschirm. Hier erscheint der Avatar und dient als Auswahlmenü. Bei dem Verlangen zu trainieren, erscheinen auf den betroffenen Muskelgruppen Hotspots, die die jeweiligen Spiele repräsentieren. Nach dem Klick auf einen der Punkte startet zuerst ein Tutorial, dass das nachfolgende Spiel erklärt. Am Ende des Tutorials erscheint der Start-Button. Mit diesem startet man die Bluetooth-Verbindung und das Spiel. Auf diese zwei Punkte wird noch später in der Dokumentation eingegangen.

Oben links in der App befindet sich ein Burger-Button, der das Hauptmenü öffnet. In diesem befinden sich weitere Views, im Einzelnen wären das: Game, Spotify, Settings und Impressum.

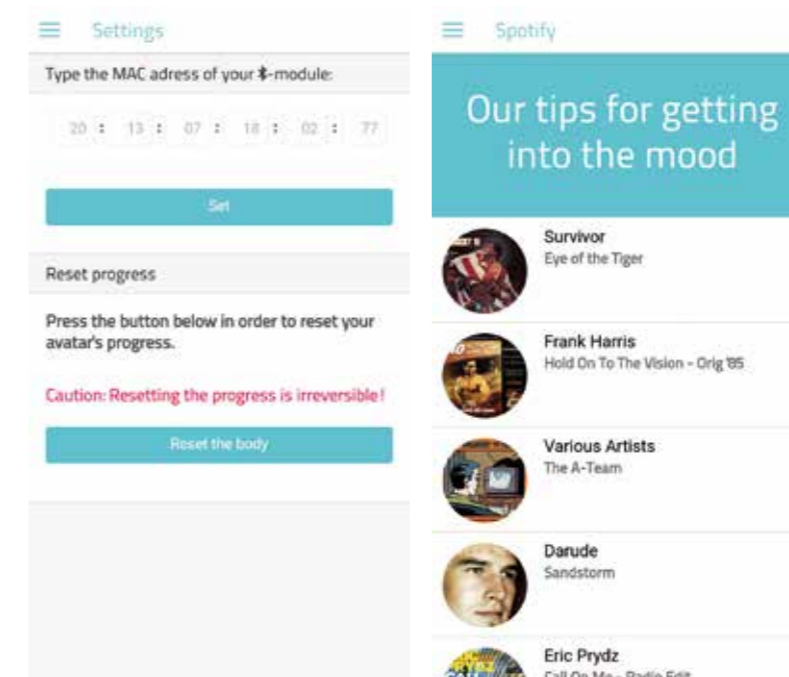


Auswahlbildschirm der zu trainierenden Muskelgruppen.

Beginn des Tutorials vor dem Spiel.

In der View „Spotify“ findet man Musikvorschläge, die in einer Spotifyplaylist abgelegt sind. Diese werden von der Web API vom genannten Musikdienst bezogen. Darauf wird ebenso noch einmal im folgenden eingegangen.

In den Settings wurden zwei verschiedene Einstellungsmöglichkeiten eingebaut. So ist es möglich, sollte man ein anderes Bluetooth-Modul benutzen, die Mac-Adresse des Empfängergerätes zu ändern, damit sich diese letztendlich mit der App verbinden können. Außerdem ist es möglich, den Fortschritt des Avatars auf die Ursprungsform zurückzusetzen. Als letzten Menüpunkt findet sich in der App das Impressum. In diesem sind nochmal alle relevanten Daten aufgelistet.



Einstellungsbildschirm der Bluetooth-Verbindung und Reset des Avatars.

Spotify Playlist innerhalb der App.



Ionic

Die App wurde unter Verwendung des IONIC-Frameworks erstellt. IONIC bietet die Möglichkeit, via HTML5, CSS, Angular-JS und der Einbindung von Cordova-Plugins hybride Web-Apps zu erstellen, die auf Smart-Devices aufgerufen werden können und deren Funktionalität simulieren. Hybride insofern, als dass sich der erstellte Quellcode in native Programmierumgebungen integrieren lässt und nach dem Export als .apk (Applikations-Dateiformat) die Apps so in den Stores der jeweiligen Anbieter wie z. B. Apple oder Google vertrieben werden können. Durch die Integration von Cordova lässt sich sogar auf native Geräte-Funktionen, wie z. B. die in den Geräten verbaute Sensorik und Kamera, aber auch auf das Adressbuch oder den internen Gerätespeicher zurückgreifen, um diese in die Software einzubinden.

PhaserJS



PhaserJS ist ein Open-Source-Framework auf Basis von JavaScript, mit dem man interaktive Spiele für Desktop- und mobile Browser programmieren kann. Dabei bietet das Framework zahlreiche Funktionen, die das Erstellen von Spielen gegenüber herkömmlichen JavaScript-Bibliotheken um einiges erleichtern. Neben Funktionen wie dem Einbinden von Bildern (Sprites), Audio-Dateien, Videos und Texten verfügt PhaserJS über eine umfangreiche Physik-Engine sowie vielen vordefinierten Animationsfunktionen. Desweiteren lassen sich verschiedene Systemzustände anlegen, wie z.B. Menüs, Lade-Screens, Game-Over-Screens und natürlich das eigentliche Spiel selbst. Die Ausgabe erfolgt wahlweise über WebGL oder HTML-5 Canvas.

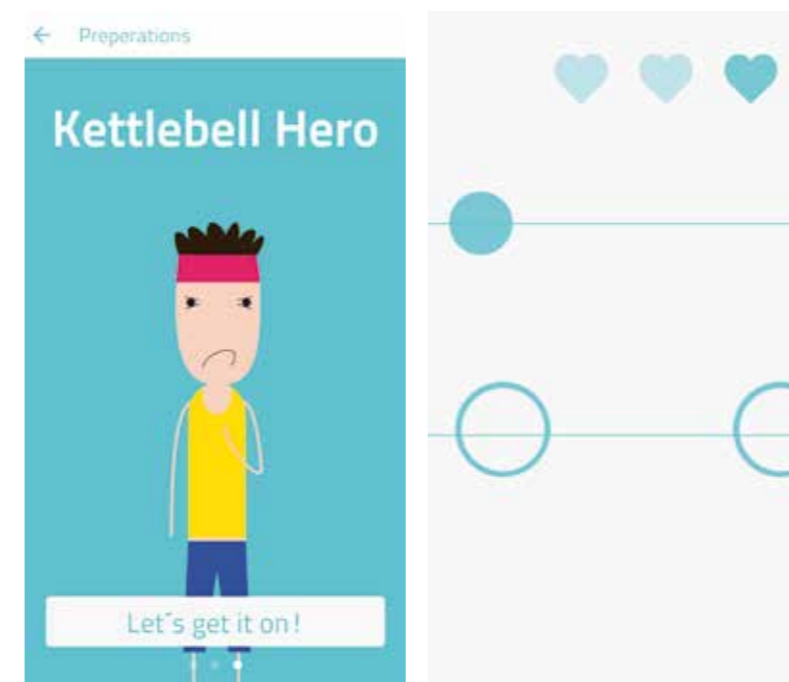
Zunächst wurde ein Prototyp des Spiels, unabhängig von einer IONIC-Instanz, entwickelt. Zur verbesserten Übersichtlichkeit wurden die einzelnen Systemzustände auf einzelne JavaScript-Dateien aufgeteilt, ähnlich wie man das bei der objektorientierten Programmierung praktiziert. Das Spiel setzt sich aus den folgenden Systemzuständen zusammen:

loadState

Hier werden alle für das Spiel notwendigen Dateien in den Cache geladen. Das beinhaltet sämtliche Grafiken sowie Audio-Dateien. Außerdem werden an dieser Stelle Textstile definiert.

bootState

Dieser Zustand wird vor dem eigentlichen Beginn des Spiels für zehn Sekunden eingeblendet und dient zur zeitlichen Überbrückung des Verbindungsaufbaus zwischen dem Bluetooth-Modul (notwendig für die Steuerung des Spiels) und dem Dekodieren der Audio-Datei. Darüber hinaus hat der Spieler die Möglichkeit, das Display so zu platzieren, dass er es gut im Blick hat, das Trainingsgerät in die Hand zu nehmen und sich auf das bevorstehende Workout einzustellen. Nach Ablauf der zehn Sekunden wechselt der Systemzustand automatisch in das eigentliche Spiel.



Ende des Tutorials und In-Game-Footage.

playState

In diesem Zustand findet das eigentliche Spiel statt. Auf zwei Linien erscheinen zufällig in einem vordefinierten Abstand Ringe, die es mit dem Ball einzusammeln gilt. Der Ball wird über das Trainingsgerät (in diesem Fall eine Kettlebelle), das mit der Elektronik versehen ist, gesteuert. Mittels einer Aufwärts- bzw. Abwärtsbewegung springt der Ball zwischen den Linien hin und her. Gerät ein Ring außerhalb des Spielbereichs, wird dem Spieler ein Lebenspunkt abgezogen. Nach 60 Sekunden wechselt der Systemzustand für fünf Sekunden in den sogenannten menuState. Der Spieler wird darum gebeten, den anderen Arm einzusetzen.

Eine Interaktion ist in diesem Zeitintervall nicht möglich. Nach Ablauf der fünf Sekunden wechselt der Systemzustand zurück in das Spiel, die zweite Runde beginnt nach den gleichen Regeln. Zu Demonstrationszwecken auf einem Desktop-Computer wurde die Möglichkeit beibehalten, den Ball auch mit den Pfeiltasten steuern zu können.

failState

Verpasst der Spieler zu viele Ringe, bricht das Spiel ab und der Fail-Screen wird eingeblendet, der dem Spieler visualisiert, dass er der Herausforderung nicht gerecht wurde. Ihm wird im Zuge dessen geraten, eine kurze Pause einzulegen und es erneut zu versuchen. Wenn der Spieler nicht möchte, kann er auf die Startseite der App zurückkehren.

gameoverState

Dieser Zustand wird nach erfolgreichem Abschluss beider runden angezeigt. Durch das Auslösen des Buttons wird der Spieler zur Startseite der App zurückgeleitet. Im Hintergrund wird ihm ein Punktwert für die entsprechende Muskelgruppe gutgeschrieben. Der Fortschritt wird dem Spieler durch den Avatar angezeigt, dessen Muskel entsprechend gewachsen ist.

Die zuvor beschriebene Dateistruktur musste zwar im späteren Projektverlauf verworfen und die Inhalte der einzelnen Dateien für das Spiel in den Controller innerhalb der IONIC-App eingefügt werden, dennoch trug die Segmentierung zu einer effizienteren Erstellung des Spiels bei. In diesem Projekt wurde auf Canvas zurückgegriffen, um eine verbesserte Performance zu gewährleisten, da WebGL sehr hardwarelastig ist und die verfügbaren Testgeräte der Belastung nicht oder nur in Einschränkungen gerecht wurden. Ein weiteres Problem entstand im Zusammenhang mit der Einbindung der Hintergrundmusik. Ursprünglich war dafür eine Audio-Datei im MP3-Format vorgesehen. Der Dekodierprozess dauerte allerdings so lange, dass die Musik auf den Testgeräten erst nach einer erheblichen Verzögerung abgespielt wurde. Nach einigen Experimenten mit dem OGG-Format und dem M4A-Format wurde schließlich letzteres eingesetzt und konnte die Verzögerung um einiges verkürzen.

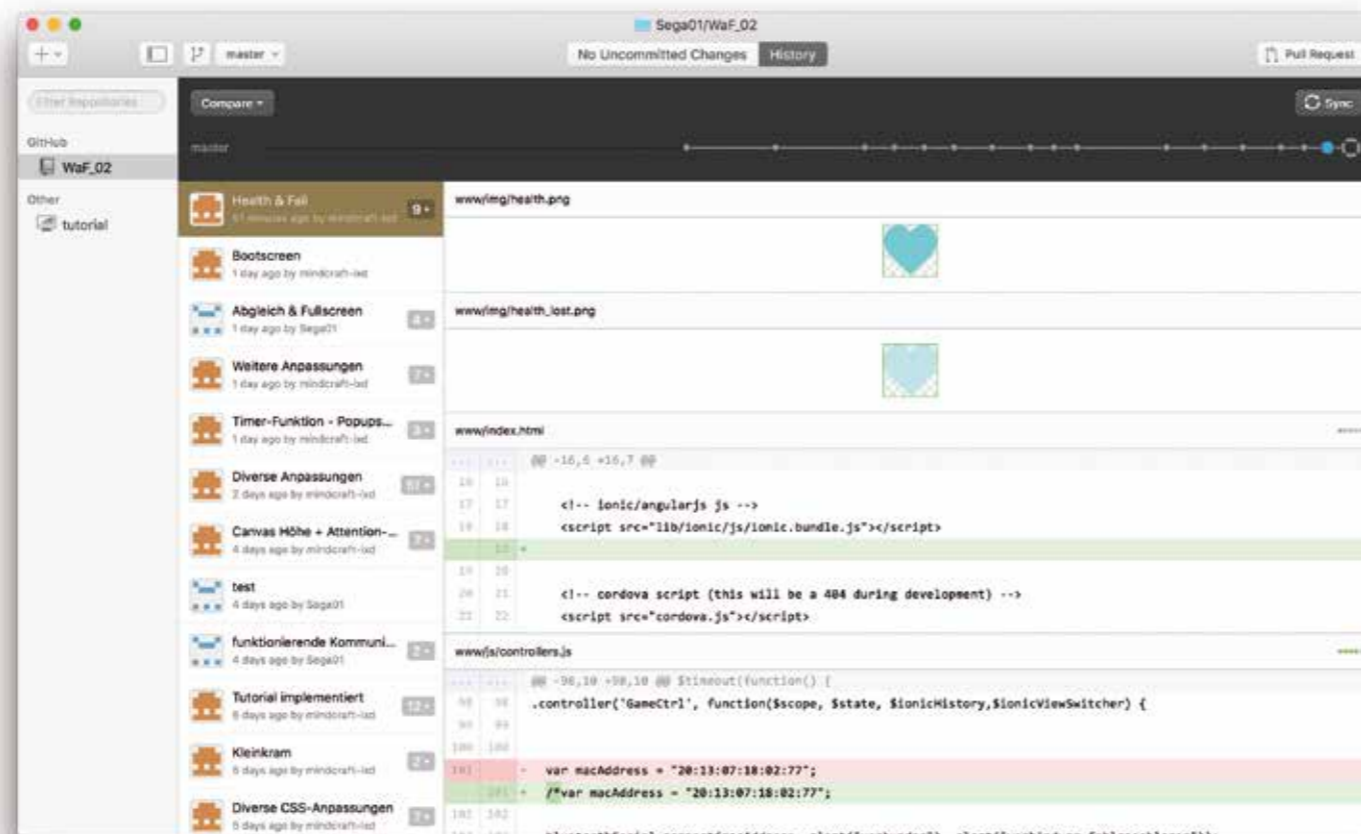
Eine Schwachstelle von PhaserJS stellt die fehlende Unterstützung für eine durchweg responsive Anpassung des Spiels auf unterschiedliche Endgeräte und Bildschirmauflösungen seitens des Frameworks dar. Die Skalierung und Positionierung von Bildern gestaltet sich sehr schwierig, gerade wenn man mit randabfallenden Grafiken, wie z. B. im gamoverState, arbeitet. Das Spiel wurde daher zunächst für ein Tablet im Hochformat mit der Auflösung von 800 x 1280 Pixeln optimiert.

Auszug aus der controller.js.

GitHub



Da ab dem Zeitpunkt, an dem ein Zwischenstand des Spiels, in die App integriert war, von mehreren Leuten gleichzeitig an der App und dem Spiel gearbeitet wurde, war ein Dienst wie GitHub bei der Entwicklung des Projekts äußerst wertvoll. GitHub ist eine Plattform, die es Entwicklern ermöglicht, alleine oder im Team an code-basierten Projekten zu arbeiten. Projekte lassen sich sowohl privat (weitere Teilnehmer werden explizit eingeladen) als auch öffentlich (Forks) bearbeiten.



Auszug aus der controller.js.

Über die Software GitHub Desktop konnten Projekte lokal abgespeichert und bearbeitet werden. Hat ein Projektteilnehmer Änderungen am Quellcode oder der Dateistruktur vorgenommen, beispielsweise durch das Hinzufügen neuer Grafiken, so konnte er über die Funktion „Commit to master“ seinen Stand auf die Plattform bringen. Dadurch wird eine neue Version des Projekts erstellt.

Über die Sync-Funktion lassen sich alle Änderungen zu einer Version verbinden und werden lokal synchronisiert. Sollten sich zwischen den Versionen Konflikte ergeben haben, wird der Nutzer direkt davon in Kenntnis gesetzt, durch eine Warnmeldung innerhalb der Software sowie durch Kommentare an den betroffenen Stellen im Quellcode. Eine genaue Auflistung von Änderungen innerhalb einer Version kann jederzeit eingesehen werden. Zusätzlich lässt sich über einen Zeitstrahl innerhalb der Software jede vergangene Version des Projektes ansteuern und einsehen. Sämtliche Funktionen der Software können natürlich auch über das GitHub-Web-Interface genutzt werden, allerdings integriert sich GitHub Desktop sehr gut in den Workflow und spart dadurch viele Klicks und sehr viel Zeit.

Die Einarbeitungszeit in GitHub war wirklich sehr kurz und hat das Projekt positiv beeinflusst. Musste man bei vergangenen Projekten zum Teil darauf warten, dass ein Projektteilnehmer eine bestimmte Funktion implementiert oder ein bestimmtes Problem behoben hat, so konnten mit der Hilfe von GitHub mehrere Personen gleichzeitig am Quellcode arbeiten. Durch die automatisch dokumentierte Versionierung, Kommentare innerhalb des Quellcodes sowie das automatisierte Zusammenführen (Mergen) konnten Konflikte im Code sehr schnell gelöst werden.

Corporate Design

Jeder Service braucht natürlich einen einheitlichen Gestaltungsrahmen, der ihn auszeichnet und den Nutzer aufgabenangemessen durch die korrespondierende Benutzeroberfläche leitet. Bei der Gestaltung wurde sehr bedacht in einem iterativen Prozess eine zufriedenstellende Lösung erarbeitet.

Schrift

Als Schrift für den Service wurde die Titillium gewählt. Die Serifenlose Schrift wirkt modern und zeitgemäß in ihrer Formensprache mit konsequenter, aber unaufdringlicher Ausprägung. Mit 16 Schriftschnitten ist sie vielseitig einsetzbar und eignet sich sowohl für Überschriften als auch für den Fließtext. Desweiteren steht die Titillium unter der OpenFont-Lizenz und ist daher für nicht-kommerziellen Projekte kostenlos nutzbar. Aufgrund dieser Faktoren wurde die Entscheidung getroffen, die Titillium für WUTZ zu nutzen.

Motivation

The spirit is willing but the flesh is weak.

RESOLUTIONARY

700 muscles, 215 bones, 915 reasons.

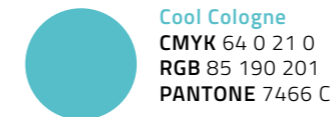
Abyss is a metaphorical barrier which stands between knowing what needs to be done and actually doing it.

Formen

In der Gestaltung mit Formen wurde sich besonders auf Rundungen konzentriert. Zurückführend auf die Natur werden runde Formen als besonders weich und angenehm wahrgenommen. Da die Formen nicht aufdringlich wirken, sowie dem Spieler vom eigentlichen Ausübung der Sportübung ablenken sollen, war dies eine adäquate Gestaltungslösung.

Farbwelt

Bei der farblichen Gestaltung der App wurde als Signalfarbe der Farbton „Cool Cologne“ gewählt. Der Farbton setzt sich als subtraktive Farbmischung signifikant aus den Farben Cyan und Yellow zusammen und wirkt jung, modern und sportlich. Aufgrund der Eigenschaften empfanden wir den Farbton als besonders geeignet für unser Projekt. In vielen Elementen der App sowie dem Prototyp findet sich diese Nuance konsequent wieder. Als zusätzliche Auszeichnungsfarben finden sich insbesondere im Avatar das Power Pink sowie das Kicking Yellow wieder. Dadurch entsteht ein harmonischer dreiklang (optisch, nicht metrisch).



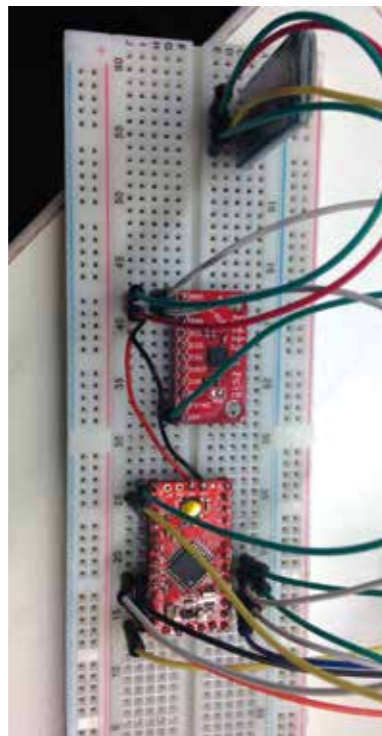
**Elektronik &
Prototyp**

Elektronik

Um den Prototypen mit der App zu verbinden bedarf es einem elektronischen Aufbau, der die App mit dem Artefakt verbindet. Im ersten Schritt wurde identifiziert, mit welche Komponenten für das Vorhaben gearbeitet werden konnte. Wir entschieden aufgrund der kompakten Bauweise mit einem Arduino Mini, einem Neun-Achsen Gyrossensor Beschleunigungen zu messen, einem Bluetooth Modul sowie einer LED und einem Schalter. Die Stromversorgung sollte durch einen Akku gegeben werden, der alle Komponenten mit Strom versorgt. In dem folgenden Schaltplan wird der technische Aufbau sichtbar.

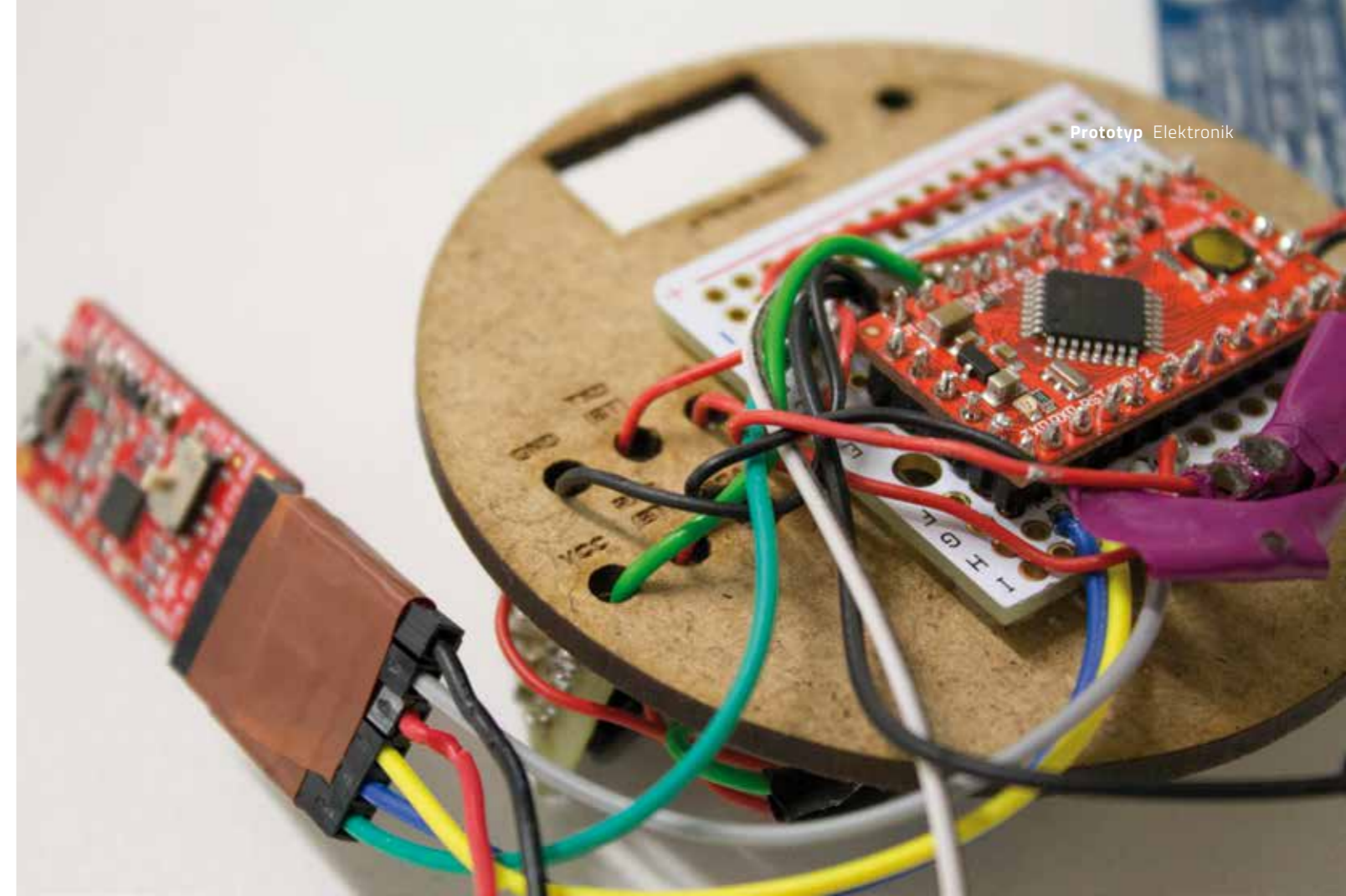
Als Herzstück des Prototypen wurde der Arduino Mini ausgewählt. Die kompakte Bauweise sowie digitale und analoge Anschlüsse gewährleiten in der Arbeit die nötige Flexibilität.

Als Akku nutzen wir die Einheit einer PowerBank. Um den Akku in das schlanke Gehäuse zu integrieren wurde das Gehäuse entfernt und nur das Innenleben genutzt. Der 2500mAh Akku der PowerBank betreibt alle Bauteile mit einem 5V Stromanschluss. Durch den Micro USB Slot der Powerbank lässt sich der Akku leicht aufladen, was von großem Vorteil für den Anwender ist.



links: erster elektronischer Probeaufbau.

rechts: Verlötung der Kontakte auf einer angefertigten Schablone.



Mit dem Beschleunigungssensor „MPU-9150“ werden unter anderem die Beschleunigungen auf 3 verschiedenen Achsen gemessen und ausgewertet. Die Zahlenwerte bleiben im ruhigen Zustand im Bereich von 0 bis 300. Wenn eine Beschleunigung stattfindet, gehen die Zahlenwerte je nach Beschleunigungskraft bis auf mehrere 1000. Der Programmcode analysiert diesen Beschleunigungswert und prüft nach ob ein bestimmter Wert erreicht wurde. Wurde der Grenzwert überschritten, wird im Programmcode eine boolean-Variable auf „true“ gesetzt, was gleichzeitig heißt, dass eine Beschleunigung stattfindet. In diesem Fall wird dann eine 1 über das Bluetooth-Modul an die App geschickt. Im anderen Fall ist die Variable „false“ und es wird eine 0 übertragen.

Die Kommunikation vom Wearable zur App wurde über ein Bluetooth-Modul gewährleistet. Zur Einbindung wurde ein Plugin benutzt, das für Cordova entwickelt wurde. Die Entscheidung fiel auf das Plugin „BluetoothSerial“, da dieses, im Gegensatz zu den anderen Plugins, die serielle Kommunikation zwischen dem Arduino und der App unterstützt hat.

Verbindung

Die Kommunikation vom Wearable zur App wurde über ein Bluetooth-Modul gewährleistet. Zur Einbindung wurde ein Plugin benutzt, das für Cordova entwickelt wurde. Die Entscheidung fiel auf das Plugin BluetoothSerial, da dieses, im Gegensatz zu den anderen Plugins, die serielle Kommunikation zwischen dem Arduino und der App unterstützt hat. Im Projekt wurden einige der Plugin-internen Methoden verwendet, die im folgenden dargestellt werden:

BluetoothSerial.connect

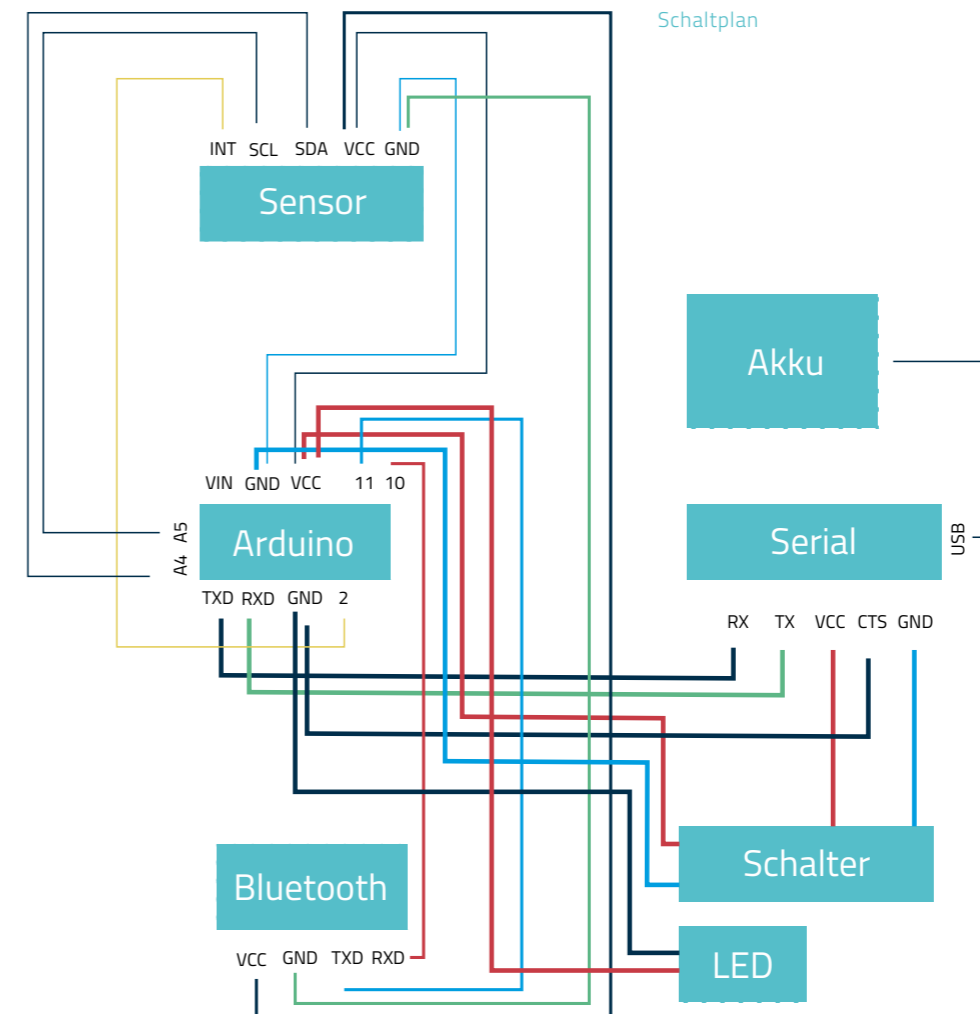
Mit dieser Funktion wird eine Verbindung zwischen der App und dem Wearable aufgebaut. Dazu muss das Gerät allerdings vorher über die Bluetooth-Einstellungen des Smartphones gekoppelt werden, damit eine Verbindung hergestellt werden kann. Sollte dies nicht der Fall sein, wird keine Verbindung aufgebaut. Die Verbindung wird mittels der MAC-Adresse des Bluetooth-Moduls hergestellt.

BluetoothSerial.disconnect

Diese Funktion löst die Verbindung mit dem Gerät wieder auf. Dies findet statt, nachdem der Spieler nach einem Spiel wieder in das Hauptmenü wechselt. Dies geschieht, da bei einer bestehenden Verbindung und dem erneuten Aufruf der connect-Funktion die Verbindung verloren ginge.

BluetoothSerial.read

Damit lassen sich die Bytes, die über das Bluetooth-Serial des Wearables gesendet werden, empfangen. Auf dem Wearable wird ein boolean-Wert ausgegeben. Dieser wird dann entweder in 1 oder 0 umgerechnet und an die App gesendet. Allerdings wurde hier auch schon ein Fehler festgestellt. Die Funktion liest alle Daten aus, die sich in dem Buffer befinden. Bei einer unterschiedlichen Übertragungsrate bedeutet dies, dass man nicht nur einen Wert, sondern mehrere empfängt, was bei dem Projekt nicht sein darf. Eine Annäherung der Übertragungsrate und Empfangsrate von Gerät und App konnte die Werte etwas stabilisieren, dennoch wurden an einigen Stellen doppelte Werte gesendet. Aus diesem Grund wurde diese Funktion wieder verworfen und eine Alternative gesucht.



BluetoothSerial.readUntil

Diese alternative Funktion liest ebenso die Daten aus, nur mit dem Unterschied eines weiteren Parameters. Hier wird ein sogenannter Delimiter eingesetzt. Das bedeutet diese Funktion liest die Werte aus, bis ein bestimmter Datensatz übertragen wird. In diesem Fall sollte nur ein Wert gelesen werden und bei der ersten Leerzeile sollte die Übertragung abgebrochen werden.

Zum Schluss wurde die Elektronik verlötet und in das Gehäuse verbaut.

Case

Um die Elektronik vor äußerlichen Einflüssen zu schützen und um der Arbeit einen visuellen Mehrwert zu geben, haben wir uns entschieden, ein Gehäuse für unser Wearable zu bauen. Im ersten Schritt wurde erfasst, in welcher Form das Case gebaut werden sollte. Die Entscheidung fiel auf eine runde Formgebung, da von Beginn an hauptsächlich mit einer Kugelhantel gearbeitet wurde. Dazu fügt sich die Form in das grafische Gesamtkonzept. Gleichzeitig hebt sich das Wearable von vergleichbaren Produkten ab.

Planung

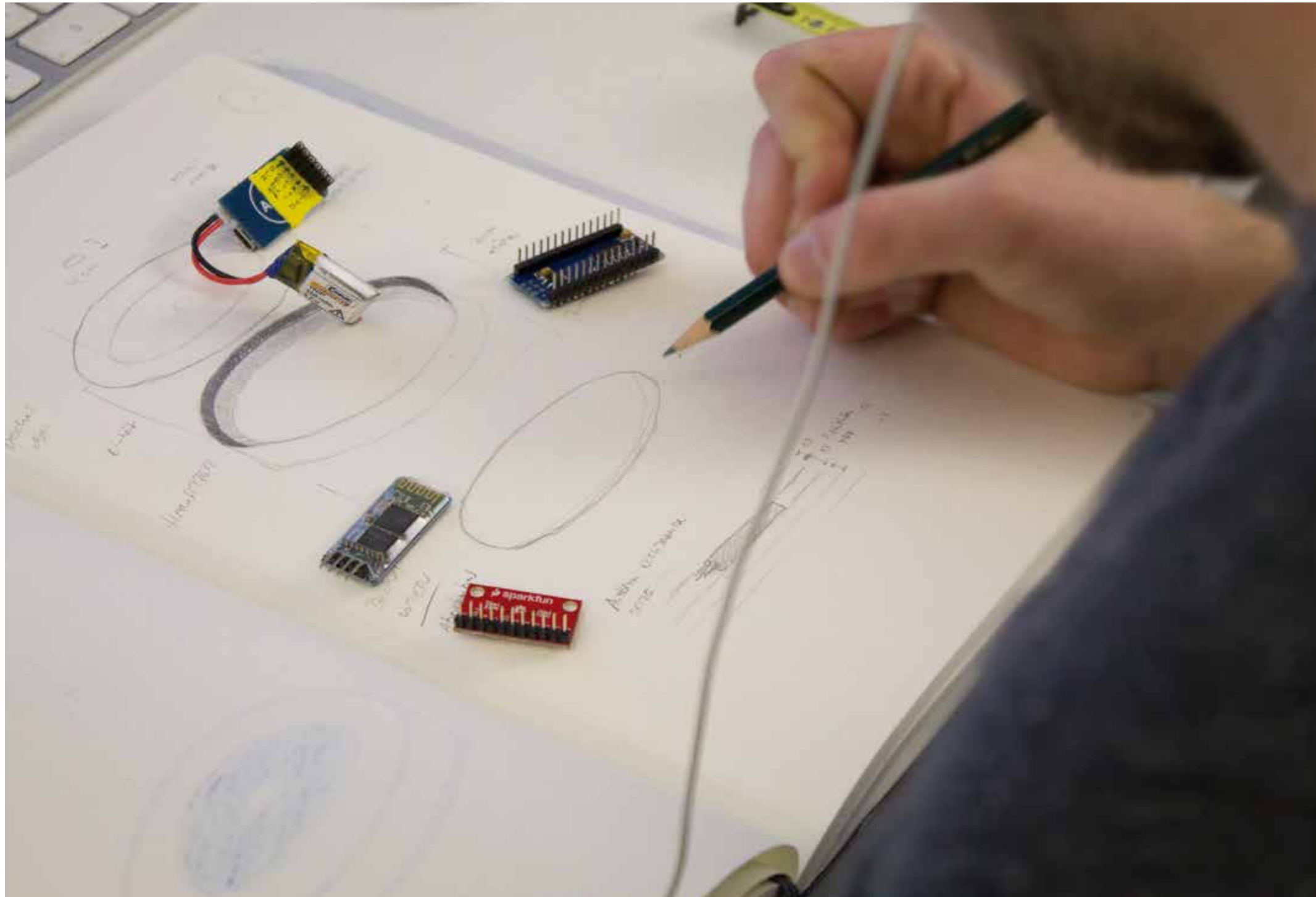
Nachdem eine passende Form gefunden wurde, konnte mit der Planung für die Integration der Elektronik begonnen werden. Um das Artefakt mobil und komfortabel beim Transport und Training zu halten, bestand die große Herausforderung darin, die einzelnen Bauteile in ein möglichst kompaktes Gehäuse zu verbauen. Hierzu wurde in iterativen Schritten eine Anordnung von Komponenten in Verbindung mit verschiedenen Gehäusegrößen vorgenommen. Es entstanden innerhalb des Prozesses eine Reihe von Gehäuse Prototypen die mithilfe des Fused Deposition Modeling (FDM) 3D Druckverfahrens angefertigt werden konnten.

Umsetzung

Als besonders effektiv für das Öffnen des Gehäuses erwies sich das Gehäuse in zwei Schalenhälften zu teilen und diese mithilfe eines Schraubverschlusses zusammenzuführen. Dies eignete sich vor allem dafür, um im Arbeitsprozess einen stetigen Zugriff auf das Innenleben mit der Elektronik zu haben. Der Schraubverschluss wurde zunächst in einer Größe von 1:10 modelliert, um einen geeigneten Reibungskoeffizienten zu ermitteln. Nachdem durch Versuche eine optimale Gewindegröße identifiziert werden konnte, wurde der Verschluss in Originalgröße in das Case integriert und gedruckt.

Um dem Benutzer eine intuitive Bedienung des Gegenstandes zu ermöglichen, entschieden wir uns einen Eingang zur Ladung der Batterie, einen Druckknopf zum An- und Ausschalten, sowie eine Öffnung für eine LED, um den aktuellen Zustand sichtbar zu machen, in das Gehäuse einzufügen.

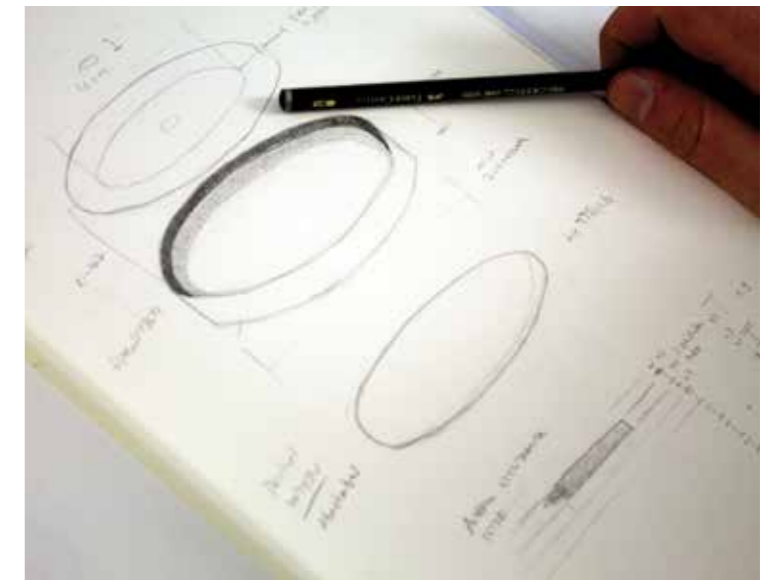




Erste Planungen.

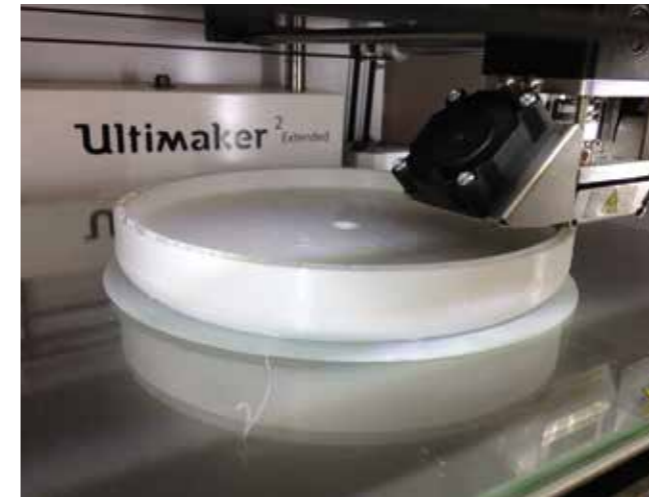


Erste Teamsitzungen.



Erste Planungen

Im nächsten Schritt begannen wir mithilfe von Modelliermasse, unebene Stellen die durch den 3D-Druck entstanden sind, auszugleichen. Die Arbeit mit der Modelliermasse erwies sich als nicht sehr effektiv, da unebene Stellen zwar ausgeglichen wurden, jedoch durch die Masse sich weitere Unebenheiten bildeten, die sich durch Abschleifen und erneuten Auftragen nicht ausgleichen ließen. Nachdem der Prototyp in einem hinreichenden Zustand war, wurde das Gehäuse mithilfe von Primer grundiert. Wir verwendeten dazu T2000 Plastic Primer von der Marke Montana. Primer ist für Oberflächen aus Plastik zur Vorbehandlung von Plastikteilen geeignet. Nach der Trocknung der Grundierung konnte die Oberfläche mit Lacken auf Synthetik-, Nitro- oder acrylbasis überlackiert werden. Durch mehrmaliges Auftragen konnten noch vorhandene Unebenheiten etwas ausgeglichen werden. Zum Schluss wurde der Prototyp in seinem Endzustand mit weißer Farbe lackiert, sowie der farbliche Akzent „Cool Cologne“ hinzugefügt.



**Ausblick &
Fazit**

Ausblick & Fazit

Im Rahmen des Semesterprojekts haben wir intensiv im Team zusammengearbeitet. Neue Inhalte und Technologien wurden erschlossen, Ideen entwickelt und anschließend wieder verworfen. Begleitet durch die Veranstaltungen haben wir Methodiken aus dem Bereich Projektmanagement kennengelernt und versucht, diese in unseren bisherigen Workflow einzubinden. Im Zuge dessen haben wir das Konzept im (zeitlichen) Rahmen unserer Hausarbeit so weit wie konnten ausgearbeitet. Wie jedoch bei jedem Project, gibt es auch bei WUTZ sicherlich noch Verbesserungs- und Erweiterungspotenzial.

Das umgesetzte Spiel ist sehr minimalistisch, bedient sich zugegebenermaßen den rudimentären Funktionen des PhaserJS-Frameworks und hätte sicherlich auch mit anderen JavaScript-Frameworks wie jQuery oder AngularJS umgesetzt werden können. Dennoch hat der integrierte Funktionsumfang nach einer moderaten Einarbeitungszeit zu einem erheblichen Zeitersparnis bei der Umsetzung der Spiels beigetragen. Außerdem ist das Konzept der App auf langer Sicht darauf ausgelegt, unterschiedliche Muskelgruppen mit mehreren kleinen Spielen zu trainieren, sodass in Zukunft auch komplexere Spielkonzepte durchaus denkbar sind.

Trotz der anfänglichen Planungsversuche ist das Projekt zeitlich stark unter Druck geraten. Beim Umgang mit sensibler Elektronik kann es schonmal vorkommen, dass diese verschleißt und sich das wiederum negativ auf den Zeitplan auswirkt. Daher mussten einige Punkte aus dem Strukturplan (z. B. Dokumentationsfilm) aus zeitlichen Gründen aus dieser Projektphase ausgeschlossen werden. Allerdings sind das Erfahrungswerte, die wir gerne mit in zukünftige Projekte nehmen und auch einige der liegegebliebenen Punkte werden noch umgesetzt.

Abschließend können wir sagen, dass es uns Spaß und Freude bereitet hat, diese Hausarbeit im Rahmen des Semesters zu erarbeiten und konnten eine Menge neue Erfahrungen sammeln und neue Erkenntnisse gewinnen, die man bei weiteren Projekten anwenden kann.

Credits

Konzeption & Entwicklung

Carlos Niermeier
Jannik Bussmann
Sebastian Galetzka

Verwendete Werkzeuge

Adobe Illustrator
Adobe Indesign
Adobe Photoshop
Cinema4D
Cordova
GitHub
Google Drive
IONIC
PhaserJS
Slack

Externes Bildmaterial

- S. 7: http://cdn2.coachmag.co.uk/sites/coachmag/files/images/dir_21/mens_fitness_10981.jpg
- S. 8: **Couch-Potatoe**
<https://www.fiercedad.com/wp-content/uploads/2014/08/couch-potato-dad-tv-remote.jpg>
- S. 9: Retusche
<http://www.adweek.com/files/imagecache/node-blog/blogs/usta-hed-2014.jpg>