

Machbarkeitsstudie:

Sensitives Kuscheltier

TEAM: LUKAS STRÄSSLER, JAKOB GRIESHOFFER,
LUKAS MAYER, PATRICK KOMON

PROJEKT: SENSITIVES KUSCHELTIER 0.8 | LETZTE ÄNDERUNG: 18.04.2015

	Name	Datum	Unterschrift
Erstellt	Patrick Komon	02.03.2015	
Geprüft	-		
Abgenommen	Prof. Walter Rafeiner-Magor	11.03.2015	

Version	Autor	QS	Datum	Status	Kommentar
0.1	Patrick Komon		02.03.2015	Draft	Struktur
0.2	Lukas Mayer		08.03.2015	Draft	Projektorganisation, Meilensteine
0.3	Patrick Komon		10.03.2015	Draft	Soll-Zustand, Produktfunktionen (tw)
0.4	Lukas Sträßler		11.03.2015	Draft	Einleitung, Ist-Zustand, Produktauswahl, PSP
0.5	Patrick Komon		11.03.2015	Draft	Produktfunktionen, Machbarkeit
0.6	Jakob Grieshofer		11.03.2015	Draft	Management Summary, Fehlerkorrektur
0.7	Patrick Komon		18.04.2015	Draft	Fehlerkorrekturen, Nutzwertanalyse
0.8	Patrick Komon		18.04.2015	Draft	Funktion: Ein-/Ausschalten, Gassensor, Technische Machbarkeit: Stromversorgung

INHALT

Inhalt.....	2
1 Einleitung.....	3
1.1 Projektbeschreibung	3
2 Ist-Zustand.....	3
3 Produktauswahl.....	3
3.1 Trendanalyse	3
4 Soll-Zustand	3
4.1 Muss-Ziele (must have)	4
4.2 Soll-Ziele (should have)	4
5 Produktfunktionen	5
5.1 Hauptfunktionen	5
6 Machbarkeit	9
6.1 Technische Machbarkeit	9
6.2 Wirtschaftliche Machbarkeit.....	12
6.3 Persönliche Machbarkeit.....	13
6.4 Risiken und Chancen	13
7 Projektorganisation	14
8 Projektplanung	15
8.1 Projektstrukturplan	15
8.2 Meilensteinplanung.....	15
9 Management Summary.....	16

1 EINLEITUNG

1.1 PROJEKTBESCHREIBUNG

Es soll ein Kuscheltier entwickelt werden, dass auf seine einzigartige Weise Altbekanntes mit Neuem verbindet. Es dient vordergründlich als Kuscheltier für Kinder mit einigen Extras. Doch gleichzeitig kann es auch als Babyphone von den Eltern verwendet werden.

Bei der Erstellung der Software und dem Kuscheltier selbst wird besonders auf Benutzerfreundlichkeit und Performance geachtet. Die Kinder und Eltern sollen auf keine Schwierigkeiten im Punkt Bedienung treffen, je einfacher und simpler die Bedienung, desto besser.

2 IST-ZUSTAND

Momentan sind Babyphones eine rein für die Eltern interessante Sache. Kein Kind kann damit etwas anfangen und wundert sich nur warum so ein blinkendes Ding in seinem Zimmer steht und dass manchmal die Stimme der Mutter herauskommt. Es kann durchaus sein, dass die eigentliche Schutzfunktion des Babyphones dadurch umgekehrt wird, dass das Gerät für das Kind so weltfremd aussieht. Diese Tatsache kann dazu führen, dass sich der Schützling vor dem Babyphone versteckt und es so seine Funktion, die Aufnahme und Übertragung der Stimme des Kindes, nicht mehr richtig ausführen kann. Darum wäre es angenehm ein Babyphone zu haben, mit dem sich das Kind untertags anfreundet und damit spielt und somit abends beim Schlafen nahe bei sich hat, da es dem Spielzeug vertraut.

3 PRODUKTAUSWAHL

3.1 TRENDANALYSE

Der momentane Trend, fast völlig egal in welcher Hinsicht, geht Richtung Digitalisierung. Vor allem der App-Markt ist schnell am Wachsen, da so ziemlich jeder Haushalt mehrere Endgeräte besitzt mit denen sie sich aus einem App-Store die neuesten Anwendungen herunterladen können. Somit ist es klar das früher oder später auch Apps für das Schützen von Kindern erscheinen. Diese Anwendungen soll aber auch gleichzeitig Funktionen für das Kind bereitstellt, damit es früh und einfach lernt sich in der Welt der Smartphones und ähnlichen zurechtzufinden und natürlich geht alles leichter wenn man einen Freund und Wegbegleiter hat, der einem beim Lernen neuer Dinge hilft.

Diese leicht zu bedienende App und Freund wird das Projekt „sensitives Kuscheltier“ vorbringen. Mit optionalen Funktionen auf der App für das Kind, einer sicheren und stabilen Babyphonefunktion und weitere Extras zum Schützen der Kleinen. Natürlich kann das Kind jederzeit ohne Handy mit dem Teddybären interagieren und spielen, ist es bereit das erste Mal eine Smartphone in die Hand zu nehmen und in die große digitale Welt einzutauchen, ist sein Freund der Teddybär da und hilft ihm dabei.

4 SOLL-ZUSTAND

Das Ziel des Projektes ist es, ein altmodisches Kuscheltier mit der neuen, modernen Technik von heute zu verbinden. Das Ergebnis soll ein kinderfreundlicher Stoff-Teddybär sein, welcher mit einer Applikation von einem Smartphone aus gesteuert werden kann. Die Zielgruppe dieses Produkts sind

also vorwiegend Kinder, welche einerseits direkt mit dem Stofftier spielen können und andererseits die Kontrolle über es mithilfe der App zu übernehmen. Sie sollen damit spielerisch den Umgang mit einem Smartphone erlernen. Zusätzlich könnte dieser Aufbau auch bei höheren Altersgruppen Interesse wecken und da es eine Babyfon-Funktion besitzen wird, wird es auch für Eltern interessant sein.

4.1 MUSS-ZIELE (MUST HAVE)

1. Benutzerfreundlichkeit

Das wichtigste an einem Produkt, welches sich vorwiegend an Kinder richtet, ist die Benutzerfreundlichkeit. Je einfach und intuitiver die App bzw. das komplette System (denn es sind ebenfalls auch noch Sensoren am Teddy vorgesehen, mit denen man direkt interagieren kann, wie zum Beispiel ein oder mehrere Drucksensor) zu bedienen ist, umso besser. Kinder wollen sich nicht langen Bedienungsanleitungen oder gar Dokumentationen ansehen und durchlesen. Sie wollen ihr Spielzeug und sofort damit spielen. Ebenfalls kann es sein, dass wenn zum Beispiel die grafische Oberfläche der App zu kompliziert ist, erst gar nicht angefangen wird, damit zu spielen oder es nach kurzer Zeit (z.B. wenn manche gewünschten Funktionen nicht gefunden werden) schnell wieder langweilig und uninteressant wird. Deshalb sollte die GUI der App und alle Eingaben die erfolgen werden generell so unkompliziert und für den Benutzer einfach sowie leicht verständlich wie möglich sein.

2. Performanz

Ein ebenfalls sehr wichtiges Erfolgskriterium ist die Performanz bzw. Übertragungs- und Verarbeitungsrate. Das Stofftier soll über eine eingebaute Kamera und ein Mikrofon verfügen. Es soll möglich sein, das von Kamera und Mikrofon aufgenommene live (mit so wenig Verzögerung wie möglich) auf das Smartphone zu übertragen und gleichzeitig abzuspielen. (Live-Streaming). Die Aufnahmen sollen in einer guten Qualität übertragen werden können. Zusätzlich sollte die Verlustrate der übertragenen Daten so gering wie möglich gehalten werden. Dieses Ziel muss erfüllt werden, damit die Hauptfunktionen (Audio- und Videoübertragung) im Produkt schließlich korrekt und gut funktionieren können.

4.2 SOLL-ZIELE (SHOULD HAVE)

1. Stabilität

Ein plötzlicher Ausfall des Teddybären oder Absturz der App wäre sehr schlecht und würde ihre Verwendung unattraktiver machen. Deshalb sollte das komplette System so stabil und ausfallsicher wie möglich sein. Nach einem Absturz sollte das komplette System aber nach dem erneuten Hochfahren wieder wie zuvor funktionieren (kein Datenverlust bei Fehlerfall).

5 PRODUKTFUNKTIONEN

5.1 HAUPTFUNKTIONEN

/LF0005/ Verbindung herstellen

Die Verbindung des Smartphones zur App wird hergestellt wenn die App erkannt hat, dass noch keine Verbindung besteht. Für das manuelle Herstellen einer Verbindung wird es einen eigenen Button in der GUI der App geben.

Funktion		Nutzen	Aufwand	Kategorie
Name	Verbindung herstellen (/LF0005/)	Hoch	Mittel	Must Have
Art	Anwendungsfall			
Beschreibung	Der Benutzer kann sich mit der App zu Teddy drahtlos (über WLAN, siehe technische Machbarkeit) verbinden. Danach kann er alle Funktionen nutzen, die der Teddy zur Verfügung stellt.			
Auslöser	Benutzer möchte Zugriff auf Funktionen des Teddys (z.B. Video-Stream)			
Ergebnis	Benutzer hat Zugriff auf Funktionen des Teddy über die App (z.B. Video-Stream)			
Akteure	App, Teddy, (Benutzer)			
Eingehende Informationen	(Benutzername und Passwort, implizit)			
Vorbedingung	Smartphone und Teddy haben keine Verbindung			
Nachbedingung	Smartphone und Teddy sind verbunden			

/LF0010/ Drehung des Kopfes

Sobald die App und der Teddy eine Verbindung zueinander aufgebaut haben, kann man mithilfe der App den Kopf des Teddys nach links oder rechts drehen.

Funktion		Nutzen	Aufwand	Kategorie
Name	Drehung des Kopfes (/LF0010/)	Hoch	Hoch	Must Have
Art	Anwendungsfall			
Beschreibung	Der Kopf des Teddybären soll von Servomotoren gedreht werden können. Die Steuerung dieser Servomotoren soll über die App erfolgen.			
Auslöser	Benutzer will Kopf drehen			
Ergebnis	Kopf dreht sich			
Akteure	Benutzer, Teddy			
Eingehende Informationen	Grad um die der Kopf zu drehen ist			
Vorbedingung	Kopf ist in Ausgangsposition			
Nachbedingung	Kopf ist gedreht			

/LF0020/ Fotoaufnahme und Übertragung

Die Aufnahme von Fotos durch eine im Teddy angebracht Kamera soll über die App steuerbar sein. So soll man einen Button innerhalb der App drücken können und es wird direkt das, von der Kamera im Stofftier, aufgenommene Bild angezeigt werden. Eine Option zum lokalen Speichern auf dem Smartphone soll ebenfalls verfügbar sein.

Funktion		Nutzen	Aufwand	Kategorie
Name	Fotoaufnahme u. Übertragung (/LF0020/)	Hoch	Hoch	Must Have
Art	Anwendungsfall			
Beschreibung	Auf die Kamera soll über die App zugegriffen werden können. Durch das Betätigen eines Buttons in der App wird ein Foto mit der Kamera aufgenommen, an das Smartphone übermittelt und dort angezeigt.			
Auslöser	Benutzer will Foto machen, Button in der App			
Ergebnis	Foto wird in App angezeigt			
Akteure	Benutzer, Teddy			
Eingehende Informationen	-			
Vorbedingung	-			
Nachbedingung	Foto wird angezeigt			

/LF0030/ Videoaufnahme und Übertragung (live)

Die Aufnahme von Videos durch die eben genannte Kamera im Teddy soll ebenfalls unterstützt werden. Diese Daten werden dann an das Smartphone geschickt („gestreamt“) und dort (live, bzw. mit möglichst kleiner Verzögerung) wiedergegeben. Zusätzlich soll während dieser Übertragung auch das Drehen des Kopfes (/LF0010/) möglich sein.

Funktion		Nutzen	Aufwand	Kategorie
Name	Video-Live-Stream (/LF0030/)	Hoch	Hoch	Must Have
Art	Anwendungsfall			
Beschreibung	Aufnahme von Video durch Kamera in Teddy, live-Übertragung der Videodaten an Smartphone, Anzeige in App			
Auslöser	Benutzer möchte Video sehen, Druck auf Button in App			
Ergebnis	Live-Video wird angezeigt			
Akteure	App, Teddy			
Eingehende Informationen	-			
Vorbedingung	Video wird nicht angezeigt			
Nachbedingung	Video wird angezeigt			

/LF0040/ Sound-Wiedergabe (statisch)

Das Kuscheltier soll einen Lautsprecher eingebaut haben, durch den einige Sounddateien, welche direkt im Speicher des Teddybären liegen, wiedergegeben werden können. Die Wiedergabe einer zufälligen Sounddatei im Speicher des Teddys soll durch das Betätigen eines Drucksensors im Stofftier ausgelöst werden.

Funktion		Nutzen	Aufwand	Kategorie
Name	Sound-Wiedergabe von Datei (/LF0040/)	Hoch	Mittel	Must Have
Art	Anwendungsfall			
Beschreibung	Benutzer drückt auf Drucksensor im Teddy, Teddy gibt zufällige Sounddatei wieder			
Auslöser	Druck auf Drucksensor			
Ergebnis	Ausgabe von Sound			
Akteure	Benutzer, Teddy			
Eingehende Informationen	-			
Vorbedingung	Nichts wird im Moment wiedergegeben			
Nachbedingung	Sound wird wiedergegeben			

/LF0045/ Sound-Wiedergabe (live)

Eine weitere Option wär es, die Aufnahmen des Mikrofons des Smartphones direkt an den Teddybären zu schicken und über seinen Lautsprecher wiederzugeben.

Funktion		Nutzen	Aufwand	Kategorie
Name	Drehung des Kopfes (/LF0010/)	Hoch	Mittel	Nice To Have
Art	Anwendungsfall			
Beschreibung	Live-Stream von Audiodaten des Mikrofons des Smartphone zum Teddy, Wiedergabe durch Lautsprecher im Teddy			
Auslöser	Benutzer möchte Audio-Livestream verwenden, Button in App			
Ergebnis	Audio-Livestream			
Akteure	Benutzer, App, Teddy			
Eingehende Informationen	Sprache von Benutzer in Mikrofon des Smartphone			
Vorbedingung	kein Livestream von Audio			
Nachbedingung	Livestream von Audio			

/LF0050/ Sound-Aufnahme (mittels App) und Übertragung (nicht live)

Über die, am Smartphone installierte, App soll die Aufnahme von Sounds möglich sein. Anschließend können diese Sounddateien auf den Teddy übertragen werden. Wie /LF0040/ beschreibt, gibt das Kuscheltier einen bestimmten, zufälligen Sound durch das Betätigen des Drucksensors wieder. Der durch diese Funktion (/LF0050/) aufgenommene und auf den Teddy übertragene Sound wird dann dieser Auswahl (dem Pool) an Sounds, die der Teddy wiedergeben kann, hinzugefügt.

Funktion		Nutzen	Aufwand	Kategorie
Name	App-Soundaufnahme (/LF0050/)	Hoch	Mittel	Must Have
Art	Anwendungsfall			
Beschreibung	Aufnahme von Sound über die App, Übertragung der Sounddatei auf Teddy, Speicherung der Datei auf Teddy			
Auslöser	Benutzer will neue Sounddatei dem Teddy hinzufügen, Button in App			
Ergebnis	Neue Sounddatei hinzugefügt			
Akteure	Benutzer, App, Teddy			
Eingehende Informationen	Sprache/Sound von Benutzer			

Vorbedingung	Sounddatei mit dem Namen ist in dem Speicher des Stofftiers nicht vorhanden			
Nachbedingung	Neue Sounddatei in Speicher des Teddys			

/LF0060/ Babyfon

Das Kuscheltier soll ebenfalls über ein Mikrofon verfügen. Das macht sich diese Funktion zunutze, dieses Mikrofon im Teddybär nimmt alles Mögliche auf, und sendet es direkt an die App, welche es live (mit so kurzer Verzögerung wie möglich) wiedergibt. Dies entspricht der Funktion eines herkömmlichen Babyfons.

Funktion		Nutzen	Aufwand	Kategorie
Name	Babyfon (/LF0060/)	Hoch	Mittel	Must Have
Art	Anwendungsfall			
Beschreibung	Audio-Livestream vom Teddy zum Smartphone, automatische Wiedergabe am Smartphone			
Auslöser	Benutzer will Mikrofon des Teddys verwenden, Button von App			
Ergebnis	Audio-Livestream vom Teddy zur App			
Akteure	Teddy, App			
Eingehende Informationen	Audio ins Mikrofon des Teddys			
Vorbedingung	-			
Nachbedingung	Audio-Livestream und Wiedergabe			

/LF0070/ Ein-/Ausschalten des Raspberrys

Das Ein- bzw. Ausschalten des Teddys soll durch einen normalen Kippschalter mögliche sein. Dieser Kippschalter soll sich irgendwo, mögliche unauffällig oder versteckt am/im Teddy befinden. Ein möglicher Ort wäre zum Beispiel ein, mit Klettverschluss verschlossenes, Fach am Rücken des Stofftieres.

Funktion		Nutzen	Aufwand	Kategorie
Name	Ein-/Ausschalten des Raspberrys (/LF0070/)	Hoch	Hoch	Must Have
Art	Anwendungsfall			
Beschreibung	Ein-/Ausschalten des Raspberrys im Teddy			
Auslöser	Benutzer will (nicht) auf Funktionen des Teddys zugreifen			
Ergebnis	Rasperry ist ein bzw. ausgeschaltet			
Akteure	Benutzer			
Eingehende Informationen	-			
Vorbedingung	-			
Nachbedingung	Rasperry bootet / Rasperry fährt herunter			

/LF0080/ Gassensor

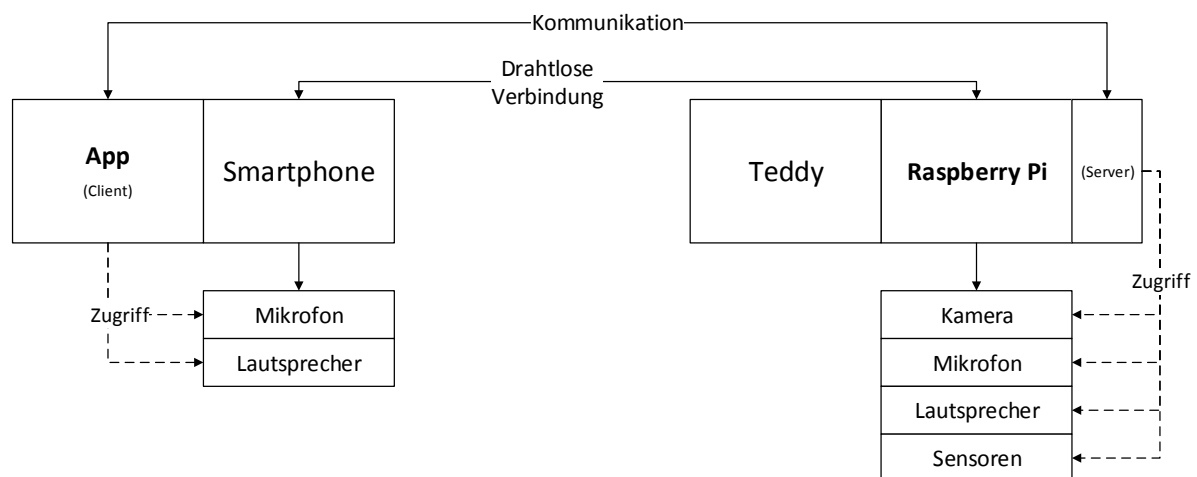
Ein Gassensor ist ebenfalls geplant. Er wird am Raspberry angeschlossen sein, um im Falle eines Brandes oder zumindest einer erhöhten Konzentration von Gas in der Luft des Kinderzimmers eine Sprachausgabe auszulösen und eine entsprechende Meldung auf dem Smartphone anzuzeigen (vorausgesetzt, es ist mit dem Raspberry verbunden).

Funktion		Nutzen	Aufwand	Kategorie
Name	Gassensor (/LF0080/)	Hoch	Hoch	Must Have
Art	Anwendungsfall			
Beschreibung	Erkennung von erhöhter Konzentration von Gas in der Luft und entsprechende Warnung			
Auslöser	Erhöhter Gasanteil in der Luft			
Ergebnis	Warnung wird ausgegeben			
Akteure	Raspberry			
Eingehende Informationen	Output des Gassensors			
Vorbedingung	Raspberry ist eingeschaltet			
Nachbedingung	-			

6 MACHBARKEIT

6.1 TECHNISCHE MACHBARKEIT

6.1.1 Schema des Systems



Wie man sehen kann besteht unser System aus zwei miteinander kommunizierenden Programmen welche auf zwei miteinander verbundenen Computern (einmal das Smartphone auf der einen Seite und auf der anderen Seite der Raspberry Pi, dessen Verwendung uns vom Auftraggeber vorgegeben wurde). Auf diesen Computern müssen zwei Programme laufen, eines greift nur auf Funktionen des anderen zu, der sogenannte Client, in Form der App und eines stellt die Funktionen zur Verfügung, dies wird ein Programm sein, das am Raspberry selbst als Dienst läuft, der Server. Diese beiden Programme kommunizieren miteinander um Daten (z.B. Audio oder Video) zwischen einander zu

übertragen und müssen beide ebenfalls jeweils auf die angeschlossene Hardware (Mikrofon, Kamera, usw.) zugreifen können. Dieser Aufbau ist im Prinzip einfach technisch realisierbar jedoch müssen einige Entscheidungen getroffen werden, da es sehr viele verschiedene Wege der Umsetzung gibt.

6.1.2 Probleme

1. Verbindung

Wie kann das Smartphone drahtlos mit dem Raspberry verbunden werden? Auf diese Frage gibt es ein paar Antworten, welche jeweils ihre Vor- und Nachteile haben.

- **Bluetooth**
Die meisten Smartphones verfügen über eine Bluetooth-Schnittstelle. Der Raspberry müsste mit einem Empfänger (z.B. in Form eines Bluetooth-Sticks über die USB-Schnittstelle). Dann müsste jedoch eben dieser Empfänger extra gekauft und konfiguriert werden. Die Übertragungsgeschwindigkeit ist aber für Video- und Audio-Streaming vermutlich ausreichend.
- **Infrarot**
Viele Smartphones besitzen auch einen Infrarot-Sender. Der Unterschied zu Bluetooth besteht, darin, dass die Infrarot-Wellen gerichtet sind, sprich: Man muss den Sender in die Richtung des Empfängers ausrichten. Dazu kommt das ebenfalls ein eigener Empfänger gekauft werden muss. Die Übertragungsgeschwindigkeit ist keinesfalls für Live-Datenstreaming ausreichend.
- **NFC (Near Field Communication)**
Moderne Smartphones sind darüber hinaus auch mit einer neuen Technologie namens NFC. Diese Technik funktioniert nur über eine sehr kleine Distanz von wenigen Zentimetern, deshalb kommt sie für unser Projekt erst gar nicht in Frage.
- **WLAN**
Eine weitere Möglichkeit wäre über das klassische WLAN. Dabei müsste ein WLAN-Adapter für den Raspberry gekauft werden. Ein Pluspunkt ist, dass man dann keinen Extra-Bildschirm oder Tastatur für die Konfiguration mit dem Raspberry benötigt, da dies über das Netzwerk geschehen kann. Damit das Produkt auch funktionieren kann ohne das ein eigener WLAN-Zugangspunkt benötigt wird, kann der WLAN-Adapter selbst einen aufbauen. Alle Smartphones können sich heutzutage ganz einfach mit einem WLAN verbinden. Somit wäre die Möglichkeit geklärt. Die Übertragungsrate hängt vom Adapter des Raspberry ab und ist bei einem qualitativ-durchschnittlich Adapter ausreichend. Auch die Reichweite ist zufriedenstellen, ca. im Bereich von 10m.

So wie es aussieht, werden die beiden Technologien, zwischen denen nun noch eine Entscheidung getroffen werden muss Bluetooth und WLAN sein. Um die Entscheidung zu treffen, wird eine Nutzwertanalyse gemacht.

Hinweis: Gewichtung und Bewertung sind im Intervall [0, 10]

Kriterien	Unterkriterien	Gewichtung	Bluetooth		WLAN	
			Bewertung	gewichtet	Bewertung	gewichtet
Kosten	<i>Empfänger</i>	5	0	0	5	25
Eigenschaften	<i>Reichweite</i>	10	5	50	6	60
	<i>Übertragungs- geschwindigkeit</i>	10	8	80	9	90
Entwicklung	<i>Einarbeitungszeit</i>	7	4	28	6	42
	<i>Entwurfszeit</i>	7	5	35	5	35
	<i>Implementierung</i>	6	5	30	5	30
Gesamt			27	223	36	282

Der Analyse kann man entnehmen, dass WLAN die bessere Variante für unser Projekt darstellt, und deshalb auch verwendet wird.

2. App-Kompatibilität

Welche Smartphones sollen die App installieren können? Nicht nur die grundsätzliche Entscheidung zwischen iOS und Android ist nötig, sondern auch eine zwischen den Versionen jener beiden. Da für die anderen Betriebssysteme für Smartphones in dem letzten Jahr einen Marktanteil von insgesamt ca. 5% festgestellt wurde, werden diese bei der Entwicklung der App vollkommen außer Acht gelassen. Der Marktanteil von Android im Jahr 2014 betrug ca. 75%, der von iOS ca. 20%. Ein weiterer Gesichtspunkt ist, dass die Entwicklung einer App für Android in Java und XML stattfindet, wobei Apps für iOS in Objective C/C/C++ geschrieben werden. Die größte Erfahrung hat das Projektteam beim Entwickeln mit Java, der große Anteil an Geräten, die es verwenden, spricht ebenfalls stark für Android.

Die Verbreitung der Android-Versionen sieht wie folgt aus: „Jelly Bean“ (4.1.x - 4.3) besitzt mit 44,5% die größte Verbreitung unter allen Android-Geräten, danach kommt „KitKat“ (4.4) mit 39,7%, „Gingerbread“ (2.3.3 - 2.3.7) mit 7,4% und „Ice Cream Sandwich“ (4.0.3 - 4.0.4) mit 6,4%. „Lollipop“ (5.0) und „Froyo“ (2.2) werden aufgrund ihrer geringen Popularität nicht beachtet.

Die Unterstützung von Versionen ab „Jelly Bean“ scheint am sinnvollsten, da dadurch etwa 84,5% aller Android-Geräte gedeckt sind, und zusätzlich nicht mit veralteten Bibliotheken gearbeitet werden muss.

3. Server-Implementierung

Die Frage nach der Sprache, in der die Server-Software implementiert werden soll, stellt sich ebenfalls. Es wäre von Vorteil wenn es einfach wäre, die Schnittstellen anzusprechen und gleichfalls eine Netzwerk-Kommunikation mit der App aufzubauen. Anbieten würde sich also dieselbe Sprache, in der auch die App geschrieben ist, da nicht auf mögliche Unterschiede zwischen den Paketen der einzelnen Programmiersprachen bestehen könnte, Rücksicht genommen werden muss. Wenn man sich für Android als Ziel-Smartphone-Betriebssystem entscheidet, wäre das Java, bei iOS Objective C, C oder C++. Dazu kommt auch die Tatsache, dass es für die Sprache eine Bibliothek gibt, mit welcher man auf die GPIO („general purpose input/output“)-Pins des Rasperrys zugegriffen werden kann. Recherchen haben gezeigt dass dies mit der Sprache Python (RPi.GPIO-Modul) besonders unkompliziert und einfach geht. In Java stünde die, in der Verwendung etwas unhandlichere Bibliothek Pi4J zur Verfügung. Eine weitere Möglichkeit wäre ein Kompromiss: Trennung der Netzwerk-Funktionalitäten von dem Zugriff und der Steuerung der Hardware. Man könnte also ein

Programm für die Kommunikation in einer Sprache, die Steuerung und Kontrolle der Hardware in einer anderen implementieren und von dem einen Programm aus, das andere aufrufen und ausführen.

4. Raspberry-Stromversorgung

Eine letzte Problemstellung, die es zu lösen gilt, gibt es noch beim Thema Stromversorgung. Wie wird der Raspberry mit Strom versorgt? Da der Teddy natürlich transportabel sein soll, und ein Kabel zu einer stationären Stromquelle wie einer Steckdose natürlich auch störend wäre, ist die einzige Möglichkeit einen Akku oder Batterien zu verwenden. Der Raspberry selbst benötigt eine Versorgungsspannung von 5V (wie der normale Output von herkömmlichen Micro-USB-Netzteilen). Alle normalen Smartphones (Ausnahme: Apple-Produkte) verwenden solche Anschlüsse zum Aufladen ihrer eingebauten Akkus. Auf dem Markt gibt es externe Akkus, welche man einfach nur an die Micro-USB-Schnittstelle seines Smartphones anstecken muss, um den internen Akku wiederaufzuladen. Genau einen solchen Akku können wir auch zur Stromversorgung des Raspberrys verwenden. Diese externen Akkus muss man selbstverständlich ebenfalls aufladen, meistens indem man sie selbst an eine Steckdose anschließt. Man könnte also den Akku in den Teddy mit dem Raspberry einbauen, und um ihn wiederaufzuladen, macht man ein kleines Fach (z.B. mittels Klettverschluss verschlossen) auf, und steckt ein Kabel zum Aufladen des Akkus an. Das wäre die einfachste Lösung. Zusätzlich könnte es von Nöten sein, dass für die Schaltungen (Sensorik) des Teddys eine zusätzliche Stromversorgung nötig ist (da der vom Raspberry zur Verfügung gestellte Strom/Spannung nicht ausreichend ist). Dafür könnte man noch einen zusätzlichen Akku verwenden.

6.2 WIRTSCHAFTLICHE MACHBARKEIT

6.2.1 Personalaufwand

Zu entwickeln sind also die App und die Server-Software. Ebenfalls muss sich das Projektteam in das Thema umfassend einlesen und einarbeiten. Dazu muss der Raspberry aufgesetzt und konfiguriert werden, elektronische Schaltungen für Sensoren entwickelt und schließlich all das noch in den Teddybären eingebaut werden. Umfassende und gewissenhafte Tests sind ebenfalls Voraussetzung für ein fehlerfreies Produkt.

Tätigkeit	Geschätzter Aufwand in Arbeitsstunden
App entwickeln	30
Server entwickeln	60
Raspberry konfigurieren	12
Entwickeln der Schaltungen	50
Einbau in Teddy	8
Testen	15

6.2.2 Investitionsaufwand

Wie man schon im Diagramm (6.1.1) erkennen kann, werden folgende Komponenten unbedingt benötigt:

- Stofftier:
Das soll ein herkömmlicher Teddy in der Größenordnung von etwa 30cm bis 50cm sein. In ihn werden später die anderen Komponenten eingebaut. Ein Preis von etwa 25€ wird erwartet.
- Raspberry Pi A+:
Das ist ein kleiner Rechner, verfügt über 256MB RAM und einen 700MHz ARM-Prozessor. Er verfügt auch über ansteuerbare GPIO-Pins. Er wird vom Auftraggeber zur Verfügung gestellt.
- Kamera:

Eine spezielle Kamera für den Raspberry Pi wird bestellt. Erwartet sind Kosten im Bereich von 25€ + Versand.

- **Mikrofon:**
Entweder es wird ein kleines USB-Mikrofon in der Preisklasse von 15€ verwendet, oder es wird ein Mikrofon-Sensor (über die GPIO-Pins angeschlossen) im Bereich von <2€ verwendet.
- **Sensoren & andere elektronische Bauteile:**
Es werden ebenfalls einige andere Sensoren an den Raspberry Pi angeschlossen, sowie andere elektronische Bauteile für deren Beschaltung. Zu erwarten sind Ausgaben im Bereich <5€

6.2.3 Materialaufwand

- **Computer und IDE(s):**
Die Entwicklungssysteme sind bereits vorhanden in Form der Laptops der Teammitglieder. IDEs zum Implementieren der Diagramme sind ebenfalls schon installiert

6.3 PERSÖNLICHE MACHBARKEIT

Das Projektteam selbst hat noch wenig Erfahrung und sein Wissen reicht im Moment nicht um das Projekt umzusetzen. Das Team besteht jedoch aus vier eifrigen Schülern, viele Inhalte des Projektes wurden im Unterricht selbst besprochen und bei den Mitgliedern ist die Bereitschaft vorhanden, sich jegliches, für die Umsetzung nötiges, Wissen, dass über die, im Unterricht vermittelten, Inhalte hinausgeht, anzueignen. Dazu sind Tutorien, Dokumentationen und Hilfestellungen im Internet zur Genüge vorhanden.

6.4 RISIKEN UND CHANCEN

1. Datenverlust	
Typ	Risiko
Möglicher Auslöser	Virus oder Computerabsturz
Eintrittswahrscheinlichkeit	Niedrig
Auswirkung	Hoch
Maßnahme	Speicherung aller relevanten Daten auf Server (z.B. GitHub); Regelmäßige Backups

2. Ausfall von Teammitglied	
Typ	Risiko
Möglicher Auslöser	Krankheit oder Verletzung eines Teammitglieds
Eintrittswahrscheinlichkeit	Niedrig
Auswirkung	Mittel
Maßnahme	Temporäre Übernahme der Tätigkeiten des ausgefallenem Teammitglied

3. Zerstörung/Beschädigung der Hardware	
Typ	Risiko
Möglicher Auslöser	Zu hohe Ströme, Gewalteinwirkung auf empfindliche Bauteile
Eintrittswahrscheinlichkeit	Mittel
Auswirkung	Hoch
Maßnahme	Vorsichtiger Umgang mit empfindlichen Teilen; Prüfung der Berechnungen der Widerstände, Ströme und Spannungen; ansonsten: Neubeschaffung der beschädigten oder zerstörten Teilen

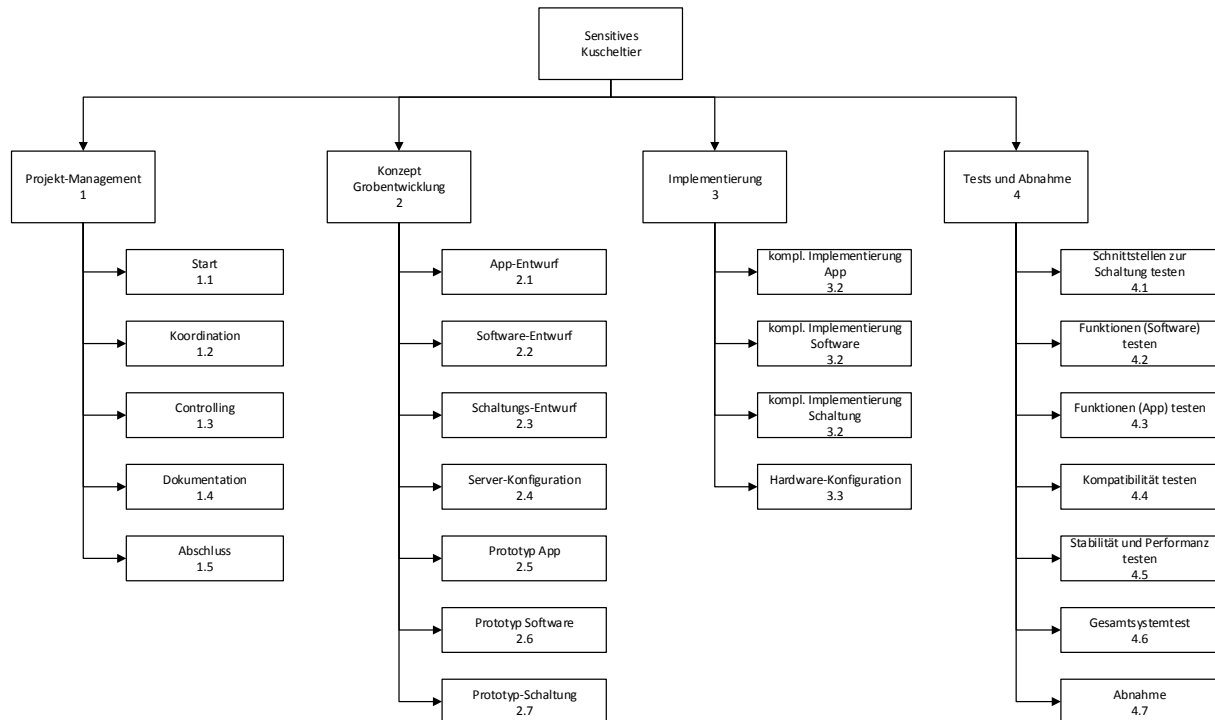
7 PROJEKTORGANISATION

Das Projektteam besteht aus dem Projektleiter Herr Sträßler und drei Entwicklern. Für den Netzwerk- sowie den Datenstreamingbereich ist Herr Komon zuständig. Die hardwarenahe Programmierung am Raspberry Pi übernimmt Herr Mayer. Soundbearbeitung sowie Implementierungs- und Testungsarbeiten übernimmt Herr Grieshofer.

Auftraggeber des Projektes ist Herr Prof. Brein. Der erweiterte Expertenkreis besteht aus Herr Prof. Rafeiner-Magor, Herr Prof. Paulitsch und Herr Prof. Radatz.

8 PROJEKTPLANUNG

8.1 PROJEKTSTRUKTURPLAN



8.2 MEILENSTEINPLANUNG

Meilenstein / Ecktermin	Termin:	Meilenstein / Ecktermin	Termin:
Ist-Analyse abgeschlossen	26.01.2015	Implementierung der Software beendet	30.04.2015
Detailplanung abgeschlossen	13.02.2015	Kuscheltier fertiggestellt	10.05.2015
Soll-Konzept ausgearbeitet	13.03.2015	Testung unter Realbedingungen abgeschlossen	24.05.2015
Entwicklung der Software abgeschlossen	13.04.2015	Projektabschluss	25.05.2015

9 MANAGEMENT SUMMARY

Kinder sind der Grundstein zukünftiger Gesellschaften und weil man wir ihren Spaß, wie auch ihre Sicherheit bewahren möchte, haben wir unser Projekt „Sensitives Kuscheltier“ gestartet. Da wir in einer digitalisierten Welt leben haben wir das erste Spielzeug eines Kindes, das Kuscheltier, mit Technik in Form eines Raspberry Pis und Sensoren, und einer für das Smartphone entwickelten Applikation zur Steuerung dieser, erweitert.

Zu den meistverwendeten Betriebssystemen im Smartphone-Bereich zählen Windows Phone, Android und iOS. Im Endergebnis ist ersichtlich, dass Android am besten für das Projekt geeignet ist.

Für die Programmierung der Software wird Python und Java verwendet.

Die Umsetzung des Projektes ist sowohl technisch, wirtschaftlich als auch persönlich machbar. Das gesamte Team sind ausreichend für die bevorstehende Aufgabe gerüstet und weitreichend einsatzbereit, die Arbeitsmittel vorhanden. Das Know-How im Feld von Java ist gut, das von Python wird sich das Projektteam noch erweitern müssen. Es wird für das Projekt ausreichend sein um die festgelegten Ziele zu erreichen.

Die Sensoren und das Kuscheltier müssen gekauft werden, dadurch fallen entsprechende Kosten an, welche voraussichtlich im Bereich unter 100€ bleiben.

Das komplette Projekt wird ca. fünf Monate in Anspruch nehmen. Die genauen Termine sind in der Meilensteinplanung ersichtlich.