

## Dokumentation AirSwimmer

<b>1</b>	<b>Anforderungen.....</b>	<b>2</b>
<b>2</b>	<b>Vorbereitung .....</b>	<b>2</b>
	2.1 Versionsverwaltung .....	2
	2.2 Android-Tutorium .....	2
<b>3</b>	<b>Alternativen.....</b>	<b>2</b>
	3.1 Alternativen ohne Infrarot.....	2
	3.2 Simulation/Emulation .....	4
<b>4</b>	<b>Projektleitung .....</b>	<b>4</b>
<b>5</b>	<b>Senden.....</b>	<b>4</b>
	5.1 Irdroid-App .....	4
	5.2 Senden.....	4
<b>6</b>	<b>Empfangen .....</b>	<b>4</b>
	6.1 Signalanalyse .....	4
	6.2 Lirc-File .....	4
<b>7</b>	<b>Gehäuse .....</b>	<b>4</b>
	7.1 Recherche Gehäusemöglichkeiten .....	4
	7.2 Gehäuse bauen .....	4
	7.3 Gehäuse bemalen und verschönern.....	4
<b>8</b>	<b>Oberfläche .....</b>	<b>5</b>
	8.1 Konzept .....	5
	8.2 Startseite.....	5
	8.2.1 Zielsetzung .....	5
	8.2.2 Vorüberlegung.....	5
	8.2.3 Vorgehensweise.....	6
	8.2.4 Ergebnis.....	9
	8.3 Steuerung mittels Buttons .....	9
	8.4 Steuerung durch Wischen .....	9
	8.5 Steuerung durch Kippen .....	9
	8.6 Menü .....	9
<b>9</b>	<b>Graphische Gestaltung der Oberfläche.....</b>	<b>9</b>
	9.1 Konzept .....	9
	9.1.1 Zielsetzung .....	9
	9.1.2 Vorüberlegung.....	9
	9.1.3 Vorgehensweise.....	11
	9.1.4 Ergebnis.....	12
	9.1.5 Ausblick.....	13
	9.2 Launcher-Icon .....	13
	9.2.1 Zielsetzung .....	13
	9.2.2 Vorüberlegung.....	14
	9.2.3 Vorgehensweise.....	14
	9.2.4 Ergebnis.....	14
	9.3 Animierter Hintergrund .....	15
	9.4 Hintergrundbilder.....	15
	9.5 Graphiken für Buttons und Hai.....	15

# **1 Anforderungen**

## **2 Vorbereitung**

### **2.1 Versionsverwaltung**

### **2.2 Android-Tutorium**

## **3 Alternativen**

### **3.1 Alternativen ohne Infrarot**

*von Anja Hafner*

Für den Fall, dass die Ansteuerung des AirSwimmers mit Infrarot nicht realisiert werden kann, wird nach alternativen Geräten gesucht.

Diese Geräte sollten mittels Bluetooth ansteuerbar sein, nicht mehr als 100 € kosten und fliegen können.

#### **Warum Bluetooth?**

Die Steuerung per Bluetooth ist gewünscht, da generell alle aktuellen Tablets und Smartphones Bluetooth besitzen und keine zusätzliche Hardware (wie bei der Steuerung mit Infrarot) nötig ist. Außerdem ist die Umsetzung dieser Kommunikation mit dem Fisch deutlich einfacher als mit Infrarot, da die Kommunikation mit Bluetooth besser dokumentiert ist.

#### **Warum soll es fliegen?**

Da es sich um ein Avionik-Projekt handelt, sollte der Gedanke, etwas flugfähiges zu steuern, nicht verloren gehen.

#### **Suche nach Alternativen**

Zu Beginn wird nach einem AirSwimmer gesucht, der mit Bluetooth gesteuert werden kann. Jedoch gibt es keinen alternativen AirSwimmer, der mit Bluetooth gesteuert wird, lediglich ferngesteuerte AirSwimmer werden angeboten.

Aus diesem Grund wird die Suche auf alle flugfähigen, mit Bluetooth und Android gesteuerten Geräte, erweitert.



Auch hier ist die Auswahl nicht groß, lediglich eine flugfähige Alternative der Firma BeeWi wird angeboten. Dieser Hubschrauber ist per Bluetooth 3.0 ansteuerbar und bietet bis zu 8 Minuten Flugzeit. Die Ladezeit hingegen beläuft sich auf ca. 40 Minuten und erfolgt per USB. Ein weiterer Nachteil ist die schwere Steuerbarkeit des Hubschraubers. Die Kosten des BeeWi BB/301-A0 Bluetooth Controlled Hubschraubers in schwarz belaufen sich auf 50,37 €.

### **Fahrfähige Alternativen**

Da nur eine flugfähige Alternative mit Bluetooth gefunden wurde, wird ebenfalls nach Bluetooth-gesteuerten Fahrzeugen gesucht, auch wenn dies nicht ganz den Anforderungen entspricht. Der Vorteil von Fahrzeugen ist die einfachere Steuerbarkeit.

Das erste Modell stammt ebenfalls von der Firma BeeWi und ist ein Mini Cooper. Dieses Modellauto wird mit Bluetooth 2.0 gesteuert und hat eine Reichweite von bis zu 10 Metern. Für den Betrieb sind 3 AA Batterien erforderlich und die Kosten betragen 30,54 € bei Amazon.

Eine weitere Alternative sind die sogenannten Tankbots. Die kleinen Fahrzeuge werden per USB aufgeladen und besitzen 3 bereits vorhandene Fahrmodi: sie können Hindernisse selbständig ausweichen, in freien Bewegungen "tanzen" oder per Smartphone gesteuert werden. Die Kosten belaufen sich auf 24,99 € bei Amazon.

Für den Fall, dass die Ansteuerung per Infrarot nicht realisiert werden kann, wurde der Hubschrauber als mögliche Alternative ausgewählt.

### **3.2 Simulation/Emulation**

## **4 Projektleitung**

## **5 Senden**

### **5.1 Irdroid-App**

### **5.2 Senden**

## **6 Empfangen**

### **6.1 Signalanalyse**

### **6.2 Lirc-File**

## **7 Gehäuse**

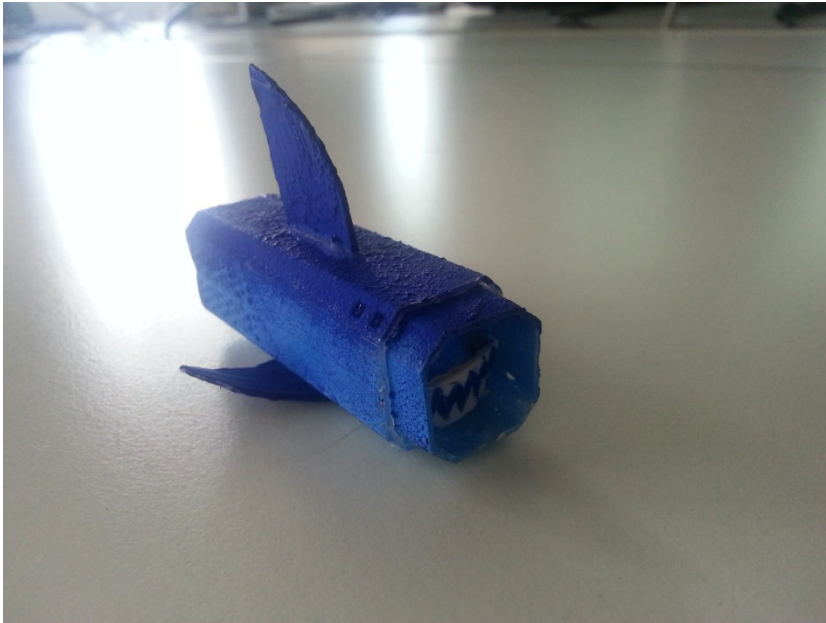
### **7.1 Recherche Gehäusemöglichkeiten**

### **7.2 Gehäuse bauen**

### **7.3 Gehäuse bemalen und verschönern**

*Von Belgüzar Kocak*

Nachdem ein passendes Gehäuse gefunden wurde und auch der Anfang zur Gestaltung von Andreas Gerken gegeben wurde, musste dieses noch farblich gestaltet werden. Als erstes wurden die Plastikflossen gefeilt anschließend mit Tesa Sekundenkleber angebracht. Die farbliche Gestaltung wurde mit Acrylfarbe und Naturschwamm erarbeitet.



## **8 Oberfläche**

### **8.1 Konzept**

### **8.2 Startseite**

*Von Caroline Pilot*

#### **8.2.1 Zielsetzung**

Jede App benötigt eine Startseite, die es dem Benutzer ermöglicht, eine Auswahl zu treffen, zwischen verschiedenen Seiten und Funktionen zu wechseln und somit den Benutzer empfängt und leitet.

Da die Startseite einen Knotenpunkt für alle Teilanwendungen (Tasten, Kippen, Wischen) darstellt, müssen diese hiermit verknüpft werden.

#### **8.2.2 Vorüberlegung**

#### **Aussehen**

Da die Startseite in erster Linie die Navigation zu den verschiedenen Benutzermodi übernehmen soll, werden drei Buttons gebraucht.

Außerdem soll die erste Seite der App natürlich optisch ansprechend wirken, deshalb wäre eine Eröffnungsanimation oder eine einladende Graphik des AirSwimmers von Vorteil.

### **Programmcode**

Mit Hilfe der Java-Klasse „Activity“ können die Buttons mit Aktionen belegt werden, die zu den einzelnen Teilfunktionen führen.

Der Programmcode soll in Englischer Sprache erstellt werden.

Außerdem soll für ein Tablet der Größe 10,1 Zoll mit einer Auflösung von 1280 x 800 Pixeln programmiert werden.

### **8.2.3 Vorgehensweise**

#### **Erstellen der Startseite**

##### ***a) Konfiguration***

Um die Einbindung der Teilfunktionen einfacher zu gestalten, wird ein neues Android Application Project aufgesetzt („AirSwimmer“) und mit folgenden Einstellungen konfiguriert:

- Minimum Required SDK → API 7: Android 2.1 (Eclair)
- Target SDK → API 7: Android 2.1 (Eclair)
- Compile with → API 7: Android 2.1 (Eclair)

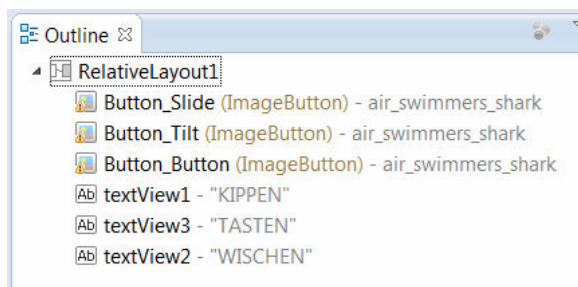
Um Probleme bei der Abwärtskompatibilität zu vermeiden, werden niedrige SDK-Versionen verwendet, damit die App auch auf Endgeräten mit älteren Android-Versionen lauffähig ist.

Die Launcher-Konfiguration wird über ein „Image“ der Form „Circle“ mit der erstellten Graphik gespeist.

Der „Activity-Name“ ist passend dazu „FrontPage“ sowie der zugehörige „Layout Name“ „front\_page“ lautet.

## **b) Layout Erstellung**

Im Unterordner „Layout“ kann nun per Drag&Drop das Erscheinungsbild der Startseite, wie in der Vorüberlegung angesprochen, erstellt werden. Diese setzt sich aus folgenden Komponenten zusammen:



**Abbildung** Fehler! Kein Text mit angegebener Formatvorlage im Dokument.1  
Komponenten des Layouts der Startseite

Die Buttons werden je nach Funktion mit einem String beschriftet.

Nach Erstellen der Buttons befindet sich für jeden Button ein Codeblock in der xml-Datei. Dieser wird jeweils um folgende Codezeile erweitert, damit der Benutzer diesen Button tatsächlich anklicken kann.

```
android:onClick="onButtonClick"
```

Außerdem wird ein passender Hintergrund gewählt, der das „AirSwimmer-Thema“ aufgreift und die Buttons werden in ihrer Größe und Ausrichtung auf die Maßen eines Tablets ausgerichtet.

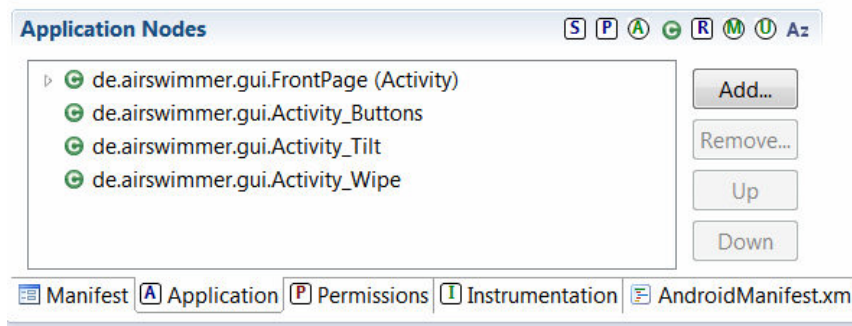
## **Einbinden der Teilfunktionen**

### **a) Layout und Source-Code einbinden**

Durch Erzeugen neuer xml-Dateien („activity\_layout\_buttons“, „activity\_layout\_wipe“ und „activity\_layout\_tilt“) im Unterordner „Layout“ und Einfügen des bereits vorhandenen Codes für die Modi, sind die Layouts der Unterfunktionen nun Bestandteil des Hauptprogramms.

Dasselbe Vorgehen findet auch für den Source-Code im Unterordner „src“ statt, mit neuen Java-Dateien („Activity\_Buttons“, „Activity\_Wipe“ und „Activity\_Tilt“).

Diese Funktionen müssen nun im AndroidManifest unter „Application“ als Knoten hinzugefügt werden:



## ***b) Funktion programmieren***

Da durch den Buttonklick zu der jeweiligen Funktion gewechselt werden soll, wird der Source-Code der Startseite um switch-case Anweisungen erweitert:

```
// What will happen when you click on a button
public void onClick(View view){
    switch(view.getId()){
        case R.id.Button_Buttons:
            startActivity(new Intent(this, Activity_Buttons.class));
            break;
        case R.id.Button_Tilt:
            startActivity(new Intent(this, Activity_Tilt.class));
            break;
        case R.id.Button_Wipe:
            startActivity(new Intent(this, Activity_Wipe.class));
        default;;
    }
}
```

Insgesamt muss beachtet werden, dass alle Namenskonflikte aufgelöst werden, alle zusätzlich genutzten Pakete ebenfalls implementiert werden und alle eingefügten Graphiken in das neue Projekt eingepflegt werden.



#### **8.2.4 Ergebnis**

Der Benutzer gelangt mit Hilfe dieser App nun über die Startseite hin zu den Teilfunktionen (Tasten, Wischen, Kippen) und kann somit den AirSwimmer steuern.

### **8.3 Steuerung mittels Buttons**

### **8.4 Steuerung durch Wischen**

### **8.5 Steuerung durch Kippen**

### **8.6 Menü**

## **9 Graphische Gestaltung der Oberfläche**

### **9.1 Konzept**

*Von Caroline Pilot*

#### **9.1.1 Zielsetzung**

Ziel eines Prototyps für das Design der Oberfläche<sup>1</sup> ist es, der Gruppe einen ersten visuellen Eindruck für die App zu geben. So bekommt das gemeinsame Ziel des Teams ein Gesicht, auf das motivierter hingearbeitet werden kann.

Außerdem bieten Modelle eine bessere Diskussionsgrundlage für Wünsche, Ideen und Verbesserungsvorschläge. So kann die Oberflächenentwicklung mit konkreten Designs ausgeführt werden.

#### **9.1.2 Vorüberlegung**

Zunächst muss ein Graphikprogramm gefunden werden, das den Anforderungen entspricht.

Hierbei ist folgendes zu beachten:

---

<sup>1</sup> Hiermit sind im Kapitel 5.1 „Konzeptentwicklung für das Oberflächendesign“ die unterschiedlichen Benutzermodi zum Steuern des AirSwimmers gemeint, nicht die Startseite der App

- Vielerlei Möglichkeiten Bilder zu bearbeiten (Kontraste, Layer, Transparenz, ...)
- Erstellen von Graphiken in gängigen Bildformaten
- Einfache Bedienung, damit keine lange Einarbeitungszeit nötig ist
- Kostengünstig, am besten ein Freeware Download Programm

Des Weiteren muss der Kreativität bei folgenden Punkten freien Lauf gelassen werden:

- Wie soll der Hintergrund aussehen?
- Wie soll das Objekt dargestellt werden?
- Welche Besonderheiten benötigen die einzelnen Benutzermodi?

### **9.1.3 Vorgehensweise**

Unter [http://www.chip.de/downloads/PhotoFiltre\\_13012070.html](http://www.chip.de/downloads/PhotoFiltre_13012070.html) ist das Freeware Graphikprogramm als Download verfügbar. Es entspricht unseren Anforderungen und enthält zudem sehr gute Nutzerrezensionen.

Aus dem Brainstorming zum Thema Oberflächendesign ergibt sich folgendes:

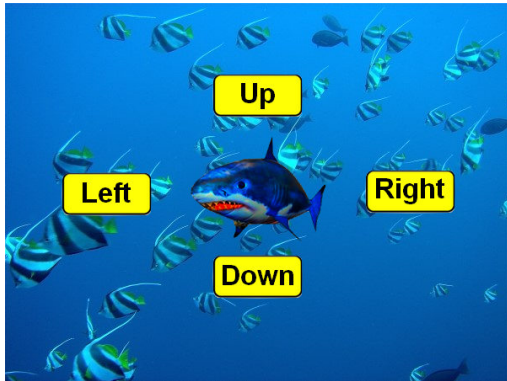
- Hintergrund:
  - Himmel
  - Unterwasserwelt
  - Stadt/Skyline
- Objekt
  - Bild des AirSwimmers
  - Bild eines Fisches
  - Bild eines Köders, dem der echte AirSwimmer „hinterher schwimmt“
- Besonderheiten einzelner Modi
  - Tastensteuerung
    - Tasten in gewöhnlicher Buttonform
    - Tasten in Form eines Fisches
    - Kontrastreiche Farben <—> Gedeckte Farben
  - Kippsteuerung
    - Angabe von Himmelsrichtungen (north, south, west, east)
    - Angabe von Bewegungsrichtung (up, down, left, right)
    - Keine Angabe
  - Wischsteuerung

- Angabe von Himmelsrichtungen (north, south, west, east)
- Angabe von Bewegungsrichtung (up, down, left, right)
- Keine Angabe

#### 9.1.4 Ergebnis

Bilder zum Verwirklichen dieser Ideen und zum Erstellen von Designprototypen lassen sich im Internet finden oder mit dem Graphikprogramm erstellen.

In verschiedenen Kombinationen, umgesetzt mit dem Photofiltre7 ergeben sich folgende Entwürfe:



**Abbildung 1**  
Tastensteuerung;  
Unterwasserwelt, Kontrastreiche Buttons



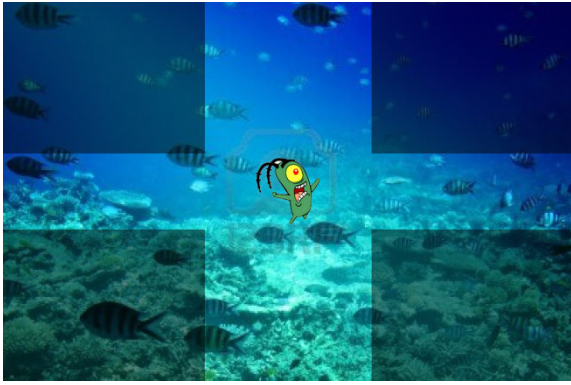
**Abbildung 2**  
Tastensteuerung;  
Himmel, Gedeckte Buttons



**Abbildung 3**  
Kippsteuerung;  
Unterwasserwelt, Angabe von  
Himmelsrichtungen



**Abbildung 4**  
Kippsteuerung;  
Skyline, ohne Angaben



**Abbildung 5**

Wischsteuerung  
Benutzergrenzen, für definierte Bewegungen, Köder  
als Objekt



**Abbildung 6**

Wischsteuerung  
Benutzergrenzen, für definierte Bewegungen

Die Gruppendiskussion bringt hervor, dass es in der App einen Punkt für einen Wechsel des Hintergrunds zwischen Himmel und Unterwasserwelt geben wird, je nach Wunsch des Benutzers.

Des Weiteren werden nicht die Himmelsrichtungen im Kippmodus angegeben, denn dies könnte zu Verwirrungen führen, stattdessen werden keine Angaben platziert, ebenso im Wischmodus.

Das Objekt in der Mitte soll ein Fisch sein.

### **9.1.5 Ausblick**

Da es sich bei den verwendeten Bildern um Eigentum Dritter handelt, könnte man sich durch dessen Verwendung rechtlich Probleme einhandeln. Aus diesem Grund ist es wünschenswert, eigene Graphiken, vielleicht auch dynamische, zu erstellen oder die Urheber schriftlich um eine Genehmigung zu bitten.

## **9.2 Launcher-Icon**

*von Caroline Pilot*

### **9.2.1 Zielsetzung**

Um einen einheitlichen Auftritt der App zu ermöglichen, wird ein Launcher-Icon benötigt. Dieser repräsentiert die App auf dem Bildschirm und ist auch bei der Ausführung am Rand zu sehen.

### **9.2.2 Vorüberlegung**

Da bereits das Graphikprogramm PhotoFiltre vorhanden ist, kann dieses verwendet werden, um den Launcher-Icon zu erstellen.

Des Weiteren muss der Kreativität bei folgenden Punkten freien Lauf gelassen werden:

- Welche Form soll der Icon bekommen?
- Welches Objekt soll darauf zu sehen sein?
- Welche Farben könnten passen?

### **9.2.3 Vorgehensweise**

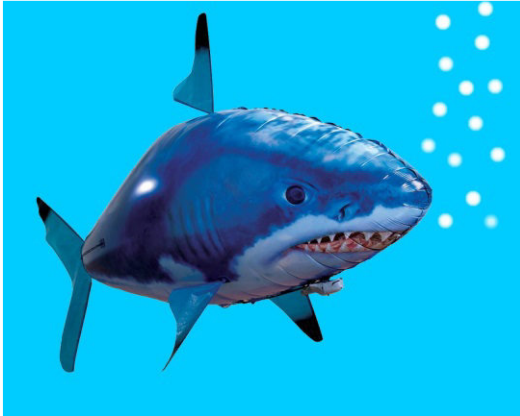
Aus dem Brainstorming ergeben sich folgende Punkte:

- Form
  - Rund, da es einer Wasserblase entspricht, in dem sich der AirSwimmer fortbewegt
- Objekt
  - Da es eine App für den AirSwimmer sein soll, liegt es nahe, dass auch auf dem Icon der Fisch abgebildet ist. Somit wäre es eindeutig auf dem Desktop zu identifizieren
- Farben
  - Die Farben müssen herausstechen, sodass die App nicht auf dem Desktop untergeht und man sie nicht sehen kann (Beispiel schwarz), dennoch müsste die Farbe zum Thema Wasser/Himmel passen

### **9.2.4 Ergebnis**

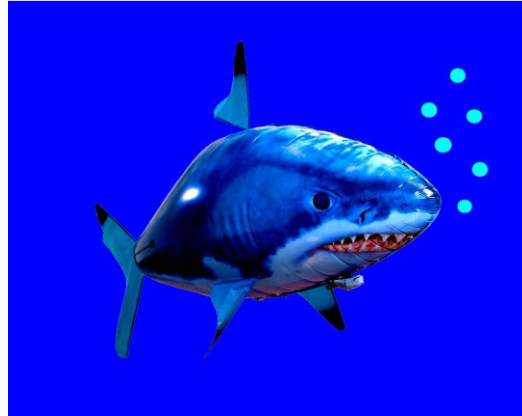
Das Bild des AirSwimmers lässt sich im Internet finden.

In verschiedenen Kombinationen mit Farben und Hintergründen, umgesetzt mit dem Photofiltre7 ergeben sich folgende Entwürfe:



**Abbildung 3**

Launcher-Icon mit hellem Hintergrund



**Abbildung 3**

Launcher-Icon mit hellem Hintergrund

Nach dem Testen auf dem Tablet wird festgestellt, dass der dunkle Hintergrund bei einem schwarzen Desktop untergeht, daher fiel die Entscheidung auf einen runden Launcher-Icon mit hellem Hintergrund (Abbildung 2).

### **9.3 Animierter Hintergrund**

### **9.4 Hintergrundbilder**

### **9.5 Graphiken für Buttons und Hai**