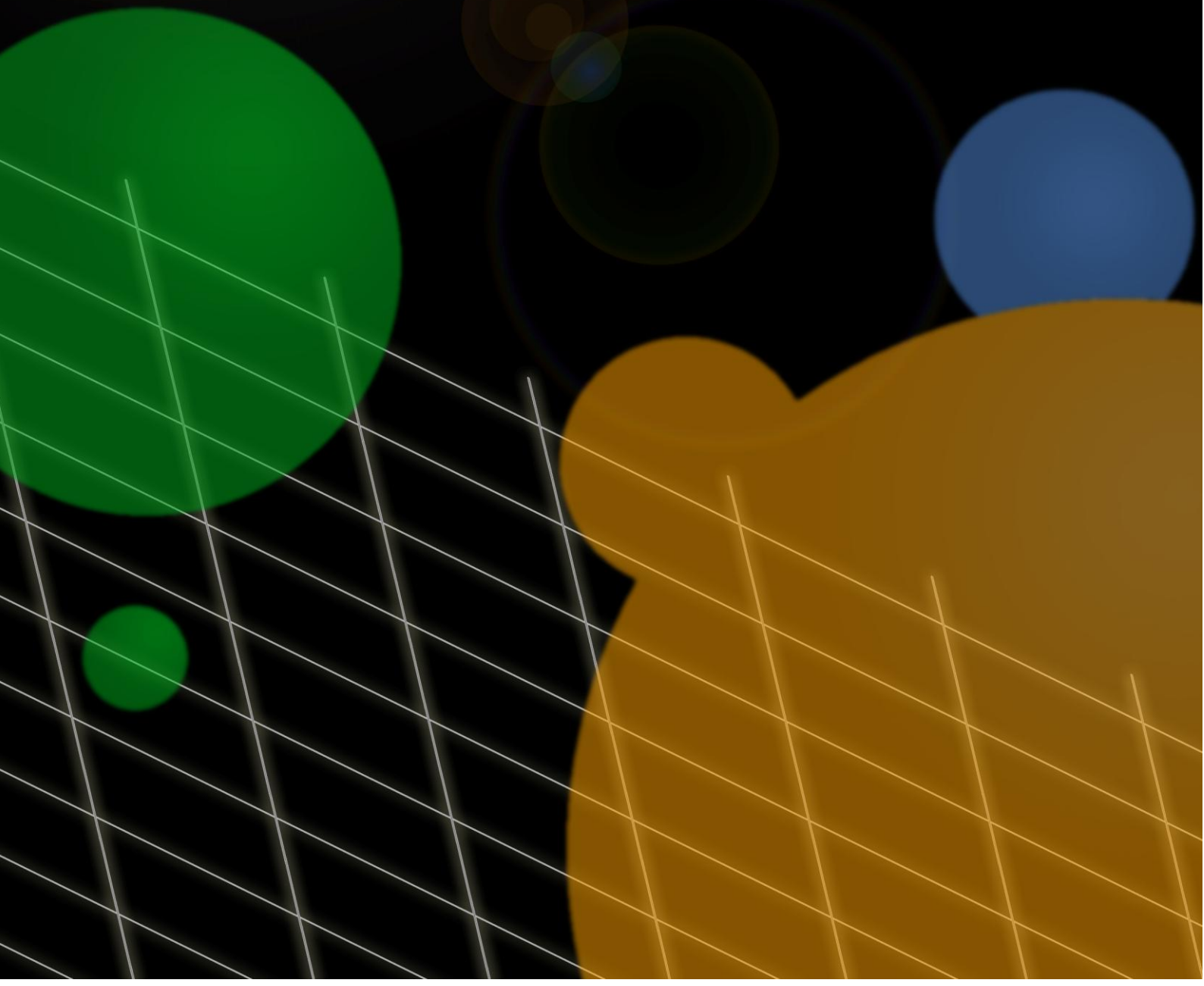


# Flash Notify



## Inhaltsverzeichnis

1.1 Projekt-Team .....	4
1.2 Projektbeschreibung .....	5
2.1 Ist-Erhebung .....	5
2.1.1 Allgemeine Ist-Erhebung .....	5
2.1.2 Ist-Zustand aktueller ähnlicher Apps .....	6
Flashlight Call .....	7
Flash Alerts .....	8
2.2 Soll-Zustand.....	9
2.2.1 Allgemeine Beschreibung .....	9
2.2.2 Muss-Ziele .....	10
2.2.3 Kann-Ziele .....	10
2.2.4 Nicht-Ziele .....	11
2.3 Festlegen der Hauptdaten .....	11
2.4 Festlegen der Hauptleistungen.....	11
2.5 Festlegen der Funktionen .....	12
2.6 Festlegen der wichtigsten Qualitätsmerkmale .....	15
3.1 Prüfen alternativer Lösungsvorschläge .....	16
3.1.1 Programmiersprachen.....	16
3.1.2 Gesamtfazit.....	18
3.2 Prüfen der technischen Durchführbarkeit.....	19
3.2.1 Technologien / Softwaretechnische Durchführbarkeit .....	19
3.2.2 Technologien / Hardwaretechnische Durchführbarkeit .....	19
3.2.3 Verfügbarkeit von Entwicklungs- und Zielmaschinen .....	19
3.3 Prüfen der personellen Durchführbarkeit .....	20
3.3.1 Qualifikation der Fachkräfte.....	20
3.3.2 Zusammenarbeit der Teammitglieder .....	20
3.4 Prüfen der Risiken.....	21
3.4.1 Personelle Risiken .....	21
3.4.2 Technische Risiken .....	22
3.5 Prüfen der ökonomischen Durchführbarkeit .....	23
3.5.1 Terminschätzung.....	23
4.1 Nutzen für den Kunden .....	26
4.2 Nutzen für das Projektteam .....	26

	Name	Datum	Unterschrift
Erstellt	Stokic Stefan	03.10.2013	
Geprüft	Mair Wolfgang	05.12.2013	

<b>Version</b>	<b>Autor</b>	<b>Datum</b>	<b>Status</b>	<b>Kommentar</b>
1.0	S. Stokic	03.10. 2013	Konzept	Grobe Anfänge
1.1	S. Stokic	24.10.2013	Progress	Fertigstellung Punkt 1
1.2	W. Mair	24.10.2013	Progress	Fertigstellung Ist-Erhebung Punkt 2
1.3	S. Stokic	07.11.2013	Progress	Durchführbarkeitsuntersuchung
1.3.1	W. Mair	07.11.2013	Progress	Fertigstellung Punkt 2
1.4	S. Stokic	14.11.2013	Progress	Fertigstellung Punkt 3, Verbesserungen offen
1.5	S. Stokic	19.11.2013	Progress	Fertigstellung Punkt 4, Punkt 5 angefangen, Verbesserungen offen
1.6	S. Stokic & W. Mair	21.11.2013	Progress	Fertigstellung der App Funktionalitäten, Terminschätzung
1.7	W. Mair	21.11.2013	Progress	Erweiterungen der Risiken
1.8	S. Stokic & W. Mair	05.12.2013	Progress	Ziele und Risiken überarbeitet

# 1. Projektdaten

---

## 1.1 Projekt-Team

**Name:** Stefan Stokic

**Rolle:** Projektleiter, Programmierer

**E-Mail:** [sstokic@student.tgm.ac.at](mailto:sstokic@student.tgm.ac.at)

**Fähigkeiten:** Java, Android-Programmierung, MySQL, C++, HTML, CSS, PHP, XML, XSLT

**Bisherige Projekte:** Space Warfare

**Name:** Wolfgang Mair

**Rolle:** Designer, Programmierer

**E-Mail:** [wmair@student.tgm.ac.at](mailto:wmair@student.tgm.ac.at)

**Fähigkeiten:** Java, Photoshop, HTML, PHP, MySQL, Flash

**Bisherige Projekte:** Space Warfare

**Name:** Daniel Djuric

**Rolle:** Designer, Projektleiter-Stellvertreter

**E-Mail:** [ddjuric@student.tgm.ac.at](mailto:ddjuric@student.tgm.ac.at)

**Fähigkeiten:** Java, Photoshop, HTML, MySQL, Javascript

**Bisherige Projekte:** Fit4U

**Name:** Fabian Freudensprung

**Rolle:** Programmierer, Designer

**E-Mail:** [ffreudensprung@student.tgm.ac.at](mailto:ffreudensprung@student.tgm.ac.at)

**Fähigkeiten:** Java, Android-Programmierung, MySQL, HTML5, CSS, Javascript, PHP

**Bisherige Projekte:** Fit4U

## 1.2 Projektbeschreibung

FlashNotify+ ist eine Android App die in Java geschrieben wird.

Bei eingehenden Anrufen, Nachrichten sogar auch bei eingehenden Nachrichten von anderen Apps wie Facebook, Whatsapp usw. soll der Flash-Blitz des eigenen Handys angehen und blitzen. Der Benutzer soll die Möglichkeit haben, die Blitz-Frequenz für die einzelnen Benachrichtigungen einstellen zu können.

# 2. Voruntersuchung des Produkts

---

## 2.1 Ist-Erhebung

### 2.1.1 Allgemeine Ist-Erhebung

Der Flash Notify ist eine Applikation welches mittels eines Smartphones ausgeführt werden kann. Das App ermöglicht dem User schnell und einfach Einstellungen auf seinem Smartphone zu verwalten.

Das Programm bietet zwar nicht einen Unterhaltungsgrad wie andere Applikationen allerdings kann es auch praktisch im Leben eingesetzt werden, was es zu einem wichtigen Werkzeug machen kann, welches durch die leicht bedienbare Oberfläche, auch von den sehr unerfahrenen Smartphone User benutzbar macht.

## 2.1.2 Ist-Zustand aktueller ähnlicher Apps

Es gibt viele verschiedene Arten von Applikationen für Smartphones die alle eine unterschiedliche Aufgabe haben. Manche haben eine Weckfunktion andere wiederum erhellen den Bildschirm um als Lampe zu blicherlesen zu dienen. Die Anzahl von Programmen die ein Smartphone benutzen kann ist schier endlos.

Um erfolgreich ein App zu produzieren muss man also auf bestimmte Punkte achten. Diese Punkte wären die Funktionalität, was bringt einem die Applikation was kann ich damit machen und wofür kann ich es einsetzen. Die Benutzerfreundlichkeit oder auch die Einfachheit der Bedienbarkeit des Programms, welches für die unerfahrenen User ein wichtiger Kritikpunkt ist. Und die Ausfallsicherheit des App's welches für den User sehr ärgerlich sein könnte wenn die Applikation an dem Moment ausfällt an dem es benötigt wird.

Die meisten App's bieten zwar eine gute Benutzerfreundlichkeit und Ausfallsicherheit, allerdings nicht besonders viele Funktionen wodurch man viele verschiedene Programme auf seinem Smartphone installieren muss um alle seine gewünschten Funktionen durchführen zu können. Manche App's bieten nicht einmal eine gute Ausfallsicherheit und werden erst mit der Zeit und Updates wirklich ausfallsicher.

## Flashlight Call



Flashlight Call ist ein sehr junges App das die ähnlichen und teilweise auch gleichen Funktionen hat wie unser Flash Notify. Es lässt ebenfalls das Licht der Kamera bei bestimmten Benachrichtigungen aufleuchten. Allerdings hat es keine weiteren Funktionen die mit dem Kamera licht zu tun hätten. Es ist ebenfalls nicht möglich die Blinkgeschwindigkeit des Lichtes bei einem Anruf zu verändern. Es hat auch nicht die Möglichkeiten die Benachrichtigungen von anderen App's wie What's app zu inkludieren.

### Funktionalität

Die Applikation „Flashlight Call“ bietet nur sehr wenige Funktionen und ist sehr eingeschränkt was Kompatibilität mit anderen Programmen betrifft. Allerdings erfüllt das App das notwendigste indem er erlaubt den Ton mit dem Blitzlicht an und auszuschalten, nebenbei kann man ebenfalls noch auswählen, falls aufgelistet, bei welchen Benachrichtigungen ein Blitzlicht erscheinen soll.

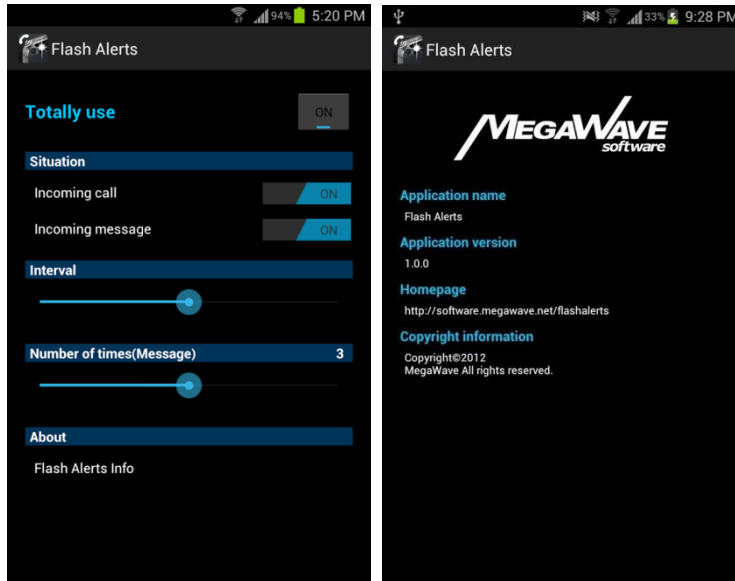
### Bedienbarkeit

Durch die wenigen Funktionen ist das App sehr übersichtlich und dadurch leicht zu bedienen. Die einzelnen Auswahlmöglichkeiten sind mittels Buttons und Checkboxen realisiert. Die einzelnen Features sind sehr leicht zu finden und zu bedienen, da die Grafik im Hintergrund nicht zu stark ablenkt und alle Features mit einen einfachen klick gestartet werden.

### Ausfallsicherheit

Das Programm ist ein Hintergrunds Programm wodurch es eine sehr lange Zeit konstant ausgeführt wird. Auch nach dem sperren des Smartphone läuft das Programm weiter und kann immer noch dessen Funktionen ausführen. Genau dieselbe Qualität an Ausfallsicherheit wollen wir mit Flashnotify ebenfalls erreichen.

## Flash Alerts



Die Applikation Flash Alerts ist ebenfalls erst ein sehr junges App das ebenfalls den Kamerablitz mit den Benachrichtigungen verknüpft. Es ist sehr simple aufgebaut hat allerdings extra Funktionen die sehr Innovativ sind, wie das einstellen der Blitzgeschwindigkeit oder wie oft es blinken soll. Die Grafik der Applikation ist Minimalistisch aufgebaut und auch die Bedienung ist so einfach wie möglich dargestellt. Flash Alert bietet keine Möglichkeit auszuwählen bei welchen Benachrichtigungen der Blitz aktiviert wird.

### Funktionalität

Flash Alerts ist sehr minimal mit der Kompatibilität von anderen App's aufgebaut hat dafür allerdings eine sehr gute Auswahl an Einstellungen für das Licht. Das App ist nur für Benachrichtigungen wie Anrufe oder SMS gedacht.

### Bedienbarkeit

Dank den einfachen Aufbau und der simplen Grafik ist es sehr einfach das App Flash Alerts zu bedienen. Ein weiterer Vorteil ist die Übersicht der Funktionen die zur Verfügung gestellt werden. Das App stellt Einfache Buttons und Slider zur Verfügung die Dank Überschrift einfach zu verstehen ist.

### Ausfallsicherheit

Das Programm ist ein Hintergrunds Programm wodurch es eine sehr lange Zeit konstant ausgeführt wird. Auch nach dem sperren des Smartphone läuft das Programm weiter und kann immer noch dessen Funktionen ausführen. Genau dieselbe Qualität an Ausfallsicherheit wollen wir mit Flashnotify ebenfalls erreichen.



## **2.2 Soll-Zustand**

### **2.2.1 Allgemeine Beschreibung**

Die Applikation Flash Notify soll größtenteils auf die Funktionalität und auf die Bedienbarkeit fokussiert sein.

Der Benutzer soll sich imstande sein Einstellungen zu tätigen die mit dem Kamerablitz zu tun haben. Das inkludiert z.B.: die Frequenz des aufleuchten des Lichtes, wann das Licht eingeschalten wird etc. Die Oberfläche sollte einfach, Übersichtlich und Informativ gestaltet werden. Es sollte die Möglichkeit bestehen eigene Benachrichtigungen zu erschaffen(Wecker Funktion) und diese auf Licht Benachrichtigung einstellbar sein. Das App soll im Hintergrund laufen und den Benutzer nicht weiter auffallen.

#### **Funktionalität**

Flash Notify ist sehr komplex was die Funktionalität betrifft, es enthält sowohl Funktionen die es ermöglichen andere App Benachrichtigungen einzubinden als auch eigene wie einen Wecker zu erstellen. Es soll dem Benutzer möglichst viele Optionen zum Konfigurieren der Blitzbenachrichtigungen geben, wie Frequenz oder Dauer des Blinkens.

#### **Bedienbarkeit**

Die Applikation soll sowohl schön anzusehen wie auch Übersichtlich und Informativ sein. Es soll Modern aussehen und dabei nicht zu ablenkend, von den Buttons und Slider die das verwenden der Funktionen ermöglicht, sein.

#### **Ausfallsicherheit**

Das Programm soll ähnlich wie andere ähnliche Apps konstant im Hintergrund laufen. Auch nach dem sperren des Smartphone läuft das Programm weiter und kann immer noch dessen Funktionen ausführen. Dabei soll Flash Notify klein genug sein das es nicht zu viele Ressourcen vom Smartphone verbraucht.

## 2.2.2 Muss-Ziele

- **Variable Einstellung des Blitzlichtes**  
Das Programm muss dem Benutzer die Möglichkeit geben, Einstellungen zu definieren, die mit dem Blitzlicht zu tun haben. z.B.: Blitzlichtfrequenz, Dauer des Blinkens und Intensität des Blitzes.
- **Variable Einstellung der Benachrichtigungen**  
Flash Notify muss Benachrichtigungen vom Smartphone erkennen können, und je nach Einstellungen das Blitzlicht benutzen.
- **Variable Verknüpfung**  
Das App wird auch Benachrichtigungen von anderen Applikationen auffangen können. Je nach Programm wird dann die vom User gewünschte Reaktion (Anruf -> schnell Blinken, SMS -> kein Blinken) ausgeführt.
- **Verfügung**  
Die Applikation muss die Möglichkeit haben im Hintergrund zu laufen und nach Belieben ausgeführt und/oder aufgerufen werden können.

## 2.2.3 Kann-Ziele

- **Geo-Wecker**  
Bei Bedarf und genug Zeit, wird die App auch einen Wecker mitliefern. Der Wecker nimmt eine Koordinate und wenn die Zeit Kritisch wird, den markierten Punkt zu erreichen, läutet der Wecker und das Blitzlicht wird aktiviert.
- **Hintergründe**  
Das Programm liefert die Möglichkeit, das Hintergrundbild von FlashNotify zu verändern. Dieses Hintergrundbild wird in den Einstellungen gespeichert.
- **Ereignis-Vielfältigkeit**  
Das Programm wird je nach App, unterschiedliche Einstellungen durchführen die der User zuvor festgelegt hat (Benachrichtigungen von What's app -> schnell blinken, Bluetooth Chat -> kein blinken).

## 2.2.4 Nicht-Ziele

- **Benachrichtigungsarten**  
Flash Notify wird mehrere Benachrichtigungsarten zur Verfügung stellen.
- **Automatisches Starten**  
Flash Notify wird die Möglichkeit besitzen mit dem einschalten des Smartphones automatisch zu starten.
- **Premium Benutzung**  
Das App bietet mit dem Kauf eines Premium Zuganges, Möglichkeit Werbung auszublenden und verschiedene Benachrichtigungsarten zur Verfügung stellen.

## 2.3 Festlegen der Hauptdaten

- **/LD10 / Grafische Benutzeroberfläche:**

Die Einstellungen sollen schnell und leicht erreichbar sein. Dafür verwenden wir ein leicht verständliches, grafisches Optionsmenü.

- **/LD20 / Benutzereinstellungen:**

Einstellungen die vom Benutzer festgelegt wurden, werden gespeichert. Diese Einstellungen werden in einer Datei dauerhaft gespeichert.

## 2.4 Festlegen der Hauptleistungen

- **/LL10 / Android Version:**

Kompatibilität ab der Android Version 2.3.3

## 2.5 Festlegen der Funktionen

- **/LF10 / Speichern der Benutzereinstellungen**

- **Eingangsdaten:**  
Frequenz, Intensität und Dauer des Blitzlichtes wird vom Benutzer bestimmt und gespeichert.
- **Ausgangsdaten:**  
Die Benutzereinstellungen werden in eine Datei gespeichert.
- **Beschreibung:**  
Der Benutzer kann in den Benutzereinstellungen die Frequenz, Intensität und die Dauer des Blitzlichtes einstellen. Diese werden dann in einer Datei gespeichert um die Einstellungen permanent gespeichert zu haben. Diese Einstellungen werden beim Programm Aufruf als Parameter ausgelesen.
- **Auslöser:**  
Das betätigen des Speicher Buttons.

- **/LF20 / Starten des Benachrichtigungsdienstes**

- **Eingangsdaten:**  
Ändern des Benachrichtigungszustandes. (Aktiviert / Deaktiviert)
- **Ausgangsdaten:**  
Der Benachrichtigungszustand. (Aktiviert / Deaktiviert)
- **Beschreibung:**  
Der Benutzer kann im Hauptmenü den Benachrichtigungsdienst entweder starten oder deaktivieren. Dieser Dienst wird dann im Hintergrund ausgeführt und hängt somit nicht direkt vom Programm ab.
- **Auslöser:**  
Das betätigen des Start Buttons für den Benachrichtigungsdienst.

- **/LF30 / Abfangen der Benachrichtigungen**

- **Eingangsdaten:**  
Je nach Benachrichtigungs-Ereignis wird die Benachrichtigung abgefangen verarbeitet.
- **Ausgangsdaten:**  
Aufrufen und Statusänderung der Funktion **/LF40 /**.

- **Beschreibung:**  
Eine Benachrichtigung wird abgefangen und eine Statusänderung in **/LF40 /** ausgeführt.
- **Auslöser:**  
Der Benachrichtigungsdienst löst dieses Ereignis aus.
- **/LF40 / Blitzlicht der Kamera aktivieren**
  - **Eingangsdaten:**  
Je nach Benutzereinstellungen wird das Blitzlicht nach der Statusänderung des Blitzlichtes (aktiviert/deaktiviert) beeinflusst.
  - **Ausgangsdaten:**  
Das Blitzlicht des Handys wird aktiviert.
  - **Beschreibung:**  
Durch die jeweilige Statusänderung werden die derzeitigen Benutzereinstellungen ausgelesen und benutzt um das Blitzlicht zu aktivieren.
  - **Auslöser:**  
**/LF30 /**.
- **/LF50 / Verbindungsaufbau mit unterstützen Apps**
  - **Eingangsdaten:**  
Die jeweiligen Apps die unterstützt werden sollen.
  - **Ausgangsdaten:**  
Erweiterungen der Benutzereinstellungen.
  - **Beschreibung:**  
Der Benutzer hat die Möglichkeit auszuwählen, welche Apps vom Benachrichtigungssystem unterstützt werden sollen.
  - **Auslöser:**  
Das betätigen des Speicher Buttons und neustarten des Benachrichtigungsdienstes.

- **/LF60W / Geo-Wecker**

- **Eingangsdaten:**  
Eine vom Benutzer eingegebene Koordinate.
- **Ausgangsdaten:**  
Aufrufen und Statusänderung der Funktion **/LF40 /** und das läuten des Weckers.
- **Beschreibung:**  
Der Benutzer hat die Möglichkeit eine Zeit anzugeben die nach Überschreitung dieser Zeit die Blitzbenachrichtigung aktiviert.
- **Auslöser:**  
Wenn die Zeit kritisch wird, die eingegebene Koordinate zu erreichen.

- **/LF70W / Benutzerdefinierte Bilder im App**

- **Eingangsdaten:**  
Eine bzw. mehrere Bilddateien.
- **Ausgangsdaten:**  
Erweiterungen der Benutzereinstellungen.
- **Beschreibung:**  
Dem Benutzer wird ermöglicht, Benutzerdefinierte Bilder als Hintergrundbilder des Apps zu definieren, die nach dem Start des Apps geladen werden.

- **/LF80W / Benutzereinstellungen für Benachrichtigungsarten**

- **Eingangsdaten:**  
Blitzlicheinstellungen (**/LF10 /**) und Benachrichtigungsart.
- **Ausgangsdaten:**  
Erweiterungen der Benutzereinstellungen.
- **Beschreibung:**  
Dem Benutzer wird ermöglicht, benutzerdefinierte Einstellungen für spezifische Benachrichtigungsarten in den Benutzereinstellungen zu speichern.

## 2.6 Festlegen der wichtigsten Qualitätsmerkmale

Produktqualität	Sehr Gut	Gut	Normal	Irrelevant
Funktionalität		X		
Zuverlässigkeit	X			
Benutzbarkeit	X			
Effizienz		X		
Änderbarkeit			X	
Übertragbarkeit				X

- **Funktionalität**

Es bietet eine Vielzahl an Features und Einstellungen für das Blitzlicht der Kamera.

- **Zuverlässigkeit**

Die Android App sollte Absturzsicher und sollte gegen Datenverlust gesichert sein.

- **Benutzbarkeit**

Das GUI soll leicht verständlich und leicht zu bedienen sein.

- **Übertragbarkeit**

Die App soll nur auf Android Geräten lauffähig sein und wird im Nachhinein definitiv nicht auf andere Systeme portiert.

## 3. Durchführbarkeitsuntersuchung

---

### 3.1 Prüfen alternativer Lösungsvorschläge

#### 3.1.1 Programmiersprachen

##### ***Variante 1: Java***

Java ist eine OOP (Objekt Orientierte Programmiersprache) welche für Webapplikationen, Programme und Android Apps verwendet wird.

Mittels Java ist die Umsetzung von Android Apps sehr einfach, da so gut wie alle Apps auf Java basieren.

##### ***Variante 2: C/C++***

Eine weitere Möglichkeit ist, die Android App in der Programmiersprache C/C++ zu programmieren. Diese Möglichkeit gewährleistet sich durch die NDK (Native Development Kit).

Es ist aber zu beachten, dass man nicht jede App so programmieren kann aber viele. Da kommen vor allem die Vor- und Nachteile ins Spiel.



## Erläuterung der Kriterien:

- **Kosten**

- Lizenzen für die Sprache:  
Muss für den Einsatz der Sprache eine Gebühr bezahlt werden?
- Lizenzen für die Entwicklungsumgebung:  
Muss für den Erwerb bzw. Einsatz einer Entwicklungsumgebung für die jeweilige Sprache eine Gebühr bezahlt werden?
- Lizenzen für kommerzielle Zwecke:  
Muss für den kommerziellen Einsatz eines in einer bestimmte Programmiersprache entwickelten Softwareprodukts eine Gebühr bezahlt werden?

- **Verbreitung**

- Kenntnisse in der Sprache  
Beschreibt inwiefern unsere Entwickler bereits mit der jeweiligen Programmiersprache vertraut sind.
- Akzeptanz der Sprache  
Beschreibt inwiefern unsere Entwickler die jeweilige Sprache gutheißen.

- **Objektorientierung**

- Umsetzung der OOP-Konzepte.  
Wurden alle Konzepte der OOP (Abstraktion, Datenkapselung, Vererbung, Polymorphie, etc.) vollständig umgesetzt?
- Durchgängigkeit  
Ist es möglich bzw. wie weit ist es möglich vom Konzept der OOP abzuweichen?

- **Einfachheit**

- Syntax  
Ist die Syntax einfach, intuitiv und logisch aufgebaut?
- Strukturierung  
Bietet die Programmiersprache die Möglichkeit einer einfachen und übersichtlichen Strukturierung?

- **API**

- Umfang  
Wie viele vorgefertigte Klassen bzw. Programmteile stehen zur Verfügung?
- Qualität  
Ist die API von einer für uns brauchbaren Qualität (d.h. fehlerfrei, effizient, etc.)?
- Dokumentation  
Sind alle Inhalte der API vollständig dokumentiert?

## **Fazit:**

Bei der zweiten Variante besteht die Möglichkeit, dass die NDK an einer Stelle nicht die benötigten Funktionen bietet und wir deshalb scheitern.

Die erste Variante(Java) schneidet am besten ab. Es besteht eine ausführliche API sowohl für Java als auch Android. Außerdem ist Java eine Programmiersprache für die Android Programmierung.

Das Ergebnis der Nutzwertanalyse deckt sich zusätzlich weitgehend mit den Wunsch-Sprachen unserer Entwickler, da diese bereits langjährige Erfahrung mit diesen sammeln konnten und somit umfangreiche Kenntnisse mitbringen.

### **3.1.2 Gesamtfazit**

Folgendes Ergebnis steht fest:

Die Applikation wird in Java, unter der Entwicklungsumgebung Eclipse, entwickelt.

## **3.2 Prüfen der technischen Durchführbarkeit**

### **3.2.1 Technologien / Softwaretechnische Durchführbarkeit**

Die Android App wird in Java entwickelt.  
Der Grund hierfür ist, dass die Variante:

1. Objektorientiert
2. Dynamisch
3. Umfangreiche API
4. Gut in der Android Programmierung anwendbar ist.

Die Objektorientierte Programmierung erlaubt es Änderungen und Weiterentwicklungen der Software in naher Zukunft effizient und fehlerfrei abzuwickeln.

Die Entscheidung für die Programmiersprache wird dadurch gestützt, dass bereits jahrelange Erfahrung besteht.

Durch die Umfangreiche API und der Erfahrung, kann die Android App Dynamisch programmiert werden.

### **3.2.2 Technologien / Hardwaretechnische Durchführbarkeit**

Dieses Projekt benötigt nur eine spezielle Hardware. Und zwar, soll das Gerät auf dem die Android App läuft eine Kamera bzw. einen Kamera Blitz haben.

### **3.2.3 Verfügbarkeit von Entwicklungs- und Zielmaschinen**

Das Produkt wird auf handelsüblichen System mit dem Betriebssystem Microsoft Windows 7 und 8 entwickelt.

Die Entwicklungsumgebung die verwendet wird, ist Eclipse und Adobe Photoshop um die gewünschten Grafiken fertigstellen zu können.

Die Zielmaschine ist ein Android Handy.

## **3.3 Prüfen der personellen Durchführbarkeit**

### **3.3.1 Qualifikation der Fachkräfte**

Da die Projektmitglieder über das nötige technische Know-How verfügen und sehr professionell und motiviert an die Sache herangehen, stellt es kein Problem dar, das Produkt zu entwickeln. Dieses Know-How wurde in der bisherigen Ausbildung sowohl theoretisch als auch praktisch oder durch freiwillige Wissensbegierde erworben.

Allerdings bestehen auch einige Risiken wie das Auftreten von unerwarteten technischen Problemen (z.B. Ausfall von Entwicklungsumgebungen), welche die personelle Durchführbarkeit beeinflussen kann.

### **3.3.2 Zusammenarbeit der Teammitglieder**

Die teaminterne Arbeit wird in diesem Projekt eine große Rolle spielen. Da jeder seine eigene Vorstellung hat wie er das Projekt am besten angehen wird und neue Ideen werden immer wieder dazukommen. Gerade diese Tatsache könnte Teamintern einerseits Verständigungsprobleme hervorrufen und andererseits zu einer nicht gerechten und gezielten Verteilung der Arbeitspakete führen und in Folge dessen einen Verzug innerhalb des Projektes verursachen.

Teamintern wurden erst wenige Erfahrungen bezüglich der Zusammenarbeit an einem gemeinsamen Projekt gesammelt. Jedoch wurden technische Probleme größtenteils durch die gute Kommunikation bereits am Ansatz behoben. Und falls es doch einmal zu einem größeren Problem gekommen ist, half das spezielle Expertenwissen jedes einzelnen Mitarbeiters zu einer schnellen, effizienten und qualitativ hochwertigen Lösung. Alle Projektmitarbeiter waren sich von Anfang an streng bewusst, dass nur durch eine starke Zusammenarbeit große Ziele erreicht werden können.

Diese bisherig gut funktionierende Zusammenarbeit wird sich in dem neuen Projekt als sehr vorteilhaft und fördernd erweisen.

## 3.4 Prüfen der Risiken

### 3.4.3 Risiken Übersicht

Risiko	Erklärung	Chance	Auswirkung	Reaktive/Präventive Maßnahme	Priorität
<b>Standard-Integration</b>	Implementierung in der nächsten Android-Version	Gering (5 %)	Groß (Abbruch)	R: Einbauen von unüblichen Funktionen.	3
<b>Änderung der API</b>	Durch neue Version, werden Funktionalitäten verhindert	Mittel (25 %)	Groß (Erhöhung der Dauer)	R: Benutzen von Teilen der API die nicht verändert werden.	1
<b>Konkurriert des Produkt</b>	Konkurrenz-Unternehmen Bringt Produkt mit ähnlichen Funktionen auf den Markt	Mittel (20 %)	Mittel (Änderungen)	R: Erweitern der Funktionen.	2
<b>Ausfall eines Mitarbeiters</b>	Ein Mitarbeiter kann seine Arbeitszeiten nicht erfüllen	Gering (10 %)	Groß (Zeitverschiebung)	R: Erhöhung der Arbeitszeiten der verbliebenen Mitarbeiter P: Mehr Ruhepausen	4
<b>Defekte Ressourcen für Testfälle</b>	Alle Test-Smartphones werden defekt.	Gering (5 %)	Mittel (Erhöhung der Kosten)	R: Kauf neuer Smartphones für Testfälle. P: Ersatzgeräte besorgen	5

### 3.4.2 Personelle Risiken

#### Ausfall eines Teammitglieds:

Der Ausfall eines Teammitglieds, sei es durch Krankheiten oder andere soziale Faktoren, kann das Gelingen des gesamten Projekts gefährden. Die Risiken liegen hierbei vor allem bei der zeitgerechten Abwicklung des Projekts und auch beim möglichen Verlust von fachspezifischen Wissensquellen.

#### Streit im Team:

Es gibt eine Menge Faktoren, die einen Streit innerhalb eines Teams auslösen können wie etwa Meinungsverschiedenheiten, soziale Spannungen, Unzufriedenheit, etc., die im schlimmsten Fall auch zur Auflösung des Teams führen können. Wichtig hierbei sind die Fähigkeiten des Projektleiters, der derartige Probleme frühzeitig erkennen und unter Einsatz seiner Soft-Skills lösen muss.

### 3.4.3 Technische Risiken

#### **Defekte Ressourcen:**

Während oder bevor die Testphase auftritt besteht die Möglichkeit das die Testgeräte defekt werden und somit nicht mehr für die Tests verwendet werden können. Um dieses Problem zu lösen muss Geld bzw. Ersatzgeräte bereitgehalten werden.

#### **Zu hoch gesteckte Ziele / Zu geringes technisches Know-How:**

Aufgrund der hohen technischen Komplexität besteht die Gefahr, dass technische Probleme auftreten, die über das technische Know-How des Projektteams hinausgehen.

Die Risiken werden hierbei durch fundierte Grundkenntnisse in den einzelnen Teilbereichen sowie bereits gesammelte Erfahrungen im Umgang mit den gewählten Technologien minimiert.

#### **Die benötigte Hardware fehlt beim Benutzer:**

Es kann sein, dass ein Benutzer der diese App benutzen wird, nicht die benötigte Hardware besitzt (der Kamera-Blitz).

Dieser Benutzer wird jedoch bei starten der App den Hinweis bekommen das die nötige Hardware fehlt.

### 3.4.4 Äußere Risiken

#### **Standard Integration:**

Es könnte passieren das während unser App in dem Alpharelease kommt die Funktionalitäten, standardmäßig im nächsten Update von Android vorkommt, dadurch wird das Projekt nutzlos und ein Misserfolg.

#### **Änderung der API:**

Es könnte bei einem API Update die nötigen Methoden bzw. benötigten Klassen verändert werden, was eine neue Änderung der Logik bewirkt und somit eine Verlängerung der Dauer.

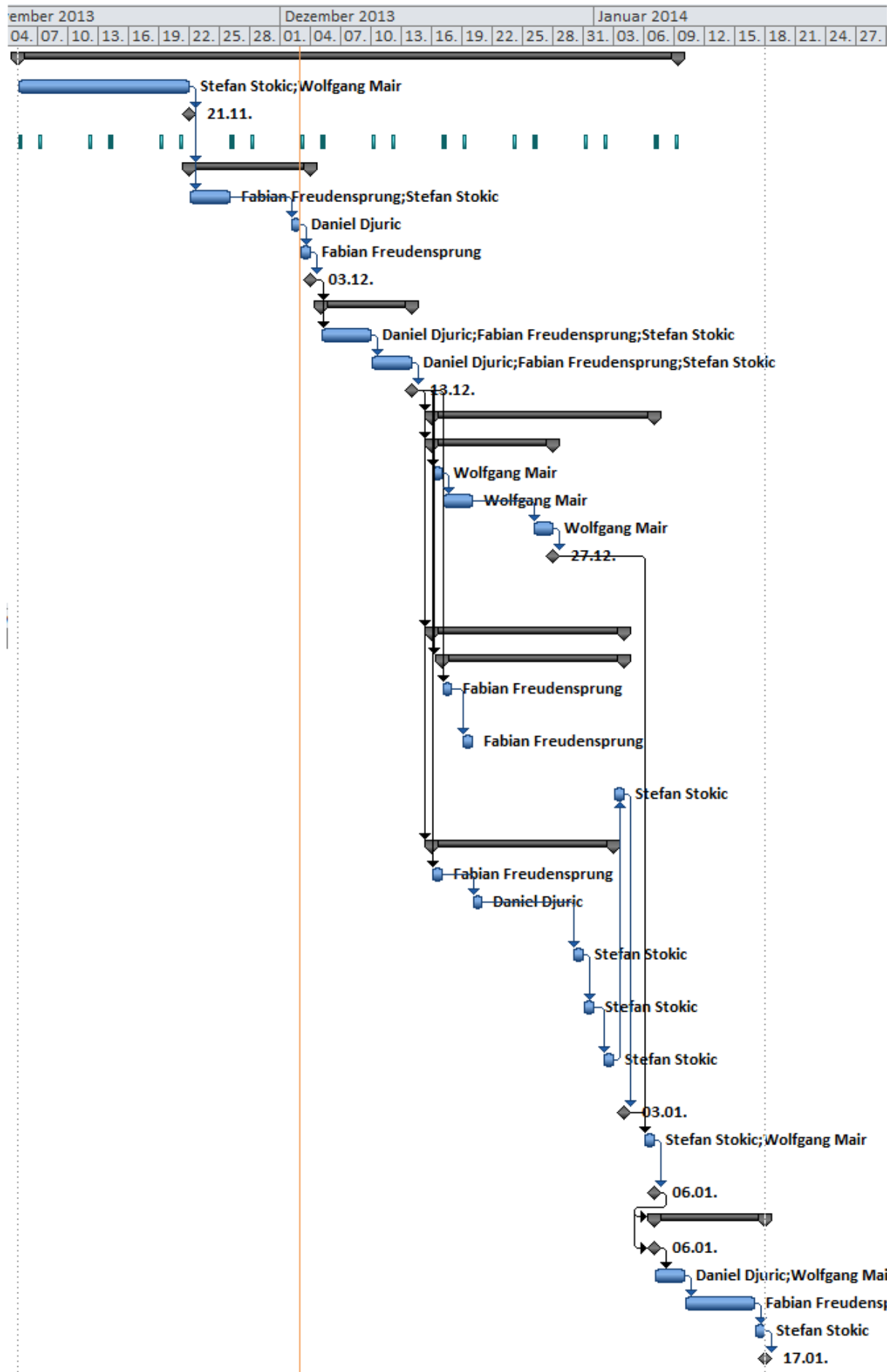
#### **Konkurrierendes Produkt:**

Ein Risiko ist das Erscheinen eines Konkurrenz Produktes das ähnliche oder gleiche Funktionen wie unser Projekt hat. Gelöst wird dieses Problem durch Informieren des Kunden bzw. erweitert die Funktionalitäten.

## 3.5 Prüfen der ökonomischen Durchführbarkeit

### 3.5.1 Terminschätzung

		Vorgan	Vorgangsname	Dauer	Anfang	Fertig stellen	Vor	Ressourcennamen	Verantwortlicher
1			<b>Projektmanagement</b>	<b>47,13 Tage</b>	<b>Di 05.11.13</b>	<b>Do 09.01.14</b>			
2			Dokumentation definieren	10 Tage	Di 05.11.13	Do 21.11.13		Stefan Stokic;Wolfgang Mair	Stefan Stokic
3			Kickoff-meeting	0 Tage	Do 21.11.13	Do 21.11.13	2	Stefan Stokic	Stefan Stokic
4			<b>Projektcontrolling</b>	<b>47,13 Tage</b>	<b>Di 05.11.13</b>	<b>Do 09.01.14</b>	<b>3</b>		<b>Stefan Stokic</b>
25			<b>Analyse u. Spezifikation</b>	<b>8 Tage</b>	<b>Fr 22.11.13</b>	<b>Di 03.12.13</b>	<b>2</b>		
26			Analyse	2 Tage	Fr 22.11.13	Mo 25.11.13	2	Fabian Freudensprung	Fabian Freudensprung
27			Spezifikationen	1 Tag	Mo 02.12.13	Mo 02.12.13	26	Daniel Djuric	Daniel Djuric
28			Überprüfung Specs	1 Tag	Di 03.12.13	Di 03.12.13	27	Fabian Freudensprung	Fabian Freudensprung
29			Specs Release	0 Tage	Di 03.12.13	Di 03.12.13	28	Daniel Djuric;Fabian Freudensprung	Daniel Djuric
30			<b>Designphase</b>	<b>7 Tage</b>	<b>Do 05.12.13</b>	<b>Fr 13.12.13</b>	<b>29</b>		
31			SystemDesign	3 Tage	Do 05.12.13	Mo 09.12.13	29	Daniel Djuric;Fabian Freudensprung	Stefan Stokic
32			SoftwareDesign	3 Tage	Di 10.12.13	Fr 13.12.13	31	Daniel Djuric;Fabian Freudensprung	Stefan Stokic
33			Design Release	0 Tage	Fr 13.12.13	Fr 13.12.13	32	Daniel Djuric;Fabian Freudensprung	Stefan Stokic
34			<b>Entwicklungsphase</b>	<b>16 Tage</b>	<b>Mo 16.12.13</b>	<b>Mo 06.01.14</b>	<b>33</b>		
35			<b>Design erstellen</b>	<b>10 Tage</b>	<b>Mo 16.12.13</b>	<b>Fr 27.12.13</b>	<b>33</b>		
36			Logo erstellen	1 Tag	Mo 16.12.13	Mo 16.12.13	33	Wolfgang Mair	Wolfgang Mair
37			Hintergrund erstellen	2 Tage	Di 17.12.13	Do 19.12.13	36	Wolfgang Mair	Wolfgang Mair
38			Buttons designen	2 Tage	Do 26.12.13	Fr 27.12.13	37	Wolfgang Mair	Wolfgang Mair
39			Bilder und Benutzeroberfläche fertig gestaltet	0 Tage	Fr 27.12.13	Fr 27.12.13	38	Wolfgang Mair	Wolfgang Mair
40			<b>Logik erstellen</b>	<b>15 Tage</b>	<b>Mo 16.12.13</b>	<b>Fr 03.01.14</b>	<b>33</b>		
41			<b>Buttons funktionen</b>	<b>14 Tage</b>	<b>Di 17.12.13</b>	<b>Fr 03.01.14</b>	<b>33</b>		
42			StartButton funktion erstellen	1 Tag	Di 17.12.13	Di 17.12.13	33	Fabian Freudensprung	Stefan Stokic
43			BeendenButton funktionen erstellen	1 Tag	Do 19.12.13	Do 19.12.13	42	Fabian Freudensprung	Stefan Stokic
44			SpeichernButton erstellen	1 Tag	Fr 03.01.14	Fr 03.01.14	50	Stefan Stokic	Stefan Stokic
45			<b>Flash funktionen</b>	<b>14 Tage</b>	<b>Mo 16.12.13</b>	<b>Do 02.01.14</b>	<b>33</b>		
46			Blitzlicht aktivieren	1 Tag	Mo 16.12.13	Mo 16.12.13	33	Fabian Freudensprung	Stefan Stokic
47			Einstellung der Blinkdauer erzeugen	1 Tag	Fr 20.12.13	Fr 20.12.13	46	Daniel Djuric	Fabian Freudensprung
48			Einstellung der Frequenz erzeugen	1 Tag	Mo 30.12.13	Mo 30.12.13	47	Stefan Stokic	Stefan Stokic
49			Abfangen der Benachrichtigungen	1 Tag	Di 31.12.13	Di 31.12.13	48	Stefan Stokic	Stefan Stokic
50			Verbindungsaufbau mit Apps	1 Tag	Do 02.01.14	Do 02.01.14	49	Stefan Stokic	Stefan Stokic
51			Fertigstellung der Logik	0 Tage	Fr 03.01.14	Fr 03.01.14	44	Stefan Stokic	Stefan Stokic
52			Verbindung des Designs und Logik	1 Tag	Mo 06.01.14	Mo 06.01.14	51;39	Stefan Stokic;Wolfgang Mair	Stefan Stokic
53			Entwicklungsabschluss	0 Tage	Mo 06.01.14	Mo 06.01.14	52	Stefan Stokic	Stefan Stokic
54			<b>Abschlussphase</b>	<b>9 Tage</b>	<b>Mo 06.01.14</b>	<b>Fr 17.01.14</b>	<b>53</b>		
55			Alpha Release	0 Tage	Mo 06.01.14	Mo 06.01.14	53	Stefan Stokic	Stefan Stokic
56			Testing	2 Tage	Di 07.01.14	Do 09.01.14	55	Daniel Djuric;Wolfgang Mair	Daniel Djuric
57			Fehler Korrektur	4 Tage	Fr 10.01.14	Do 16.01.14	56	Fabian Freudensprung	Fabian Freudensprung
58			Abnahme	1 Tag	Fr 17.01.14	Fr 17.01.14	57	Stefan Stokic	Stefan Stokic
59			Projektabschluss	0 Tage	Fr 17.01.14	Fr 17.01.14	58	Stefan Stokic	Stefan Stokic





### **Meilensteine:**

<b>Meilenstein</b>	<b>Datum</b>
<b>Kickoff-Meeting</b>	21.11.13
<b>Specs Release</b>	03.12.13
<b>Design Release</b>	13.12.13
<b>Grafiken fertiggestellt</b>	27.12.13
<b>Fertigstellung der Logik</b>	03.01.14
<b>Entwicklungsabschluss</b>	06.01.14
<b>Alpha Release</b>	06.01.14
<b>Projektabschluss</b>	17.01.14

### **Terminverschiebungen:**

Die Gründe für eine mögliche Terminverschiebung in unserem Projekt werden durch folgende Faktoren bestimmt:

- Technische Probleme
- Personalmangel
  - Krankheit von Mitgliedern
  - Ausfall von Mitgliedern
- Geringer Zeit Puffer
- Change Request

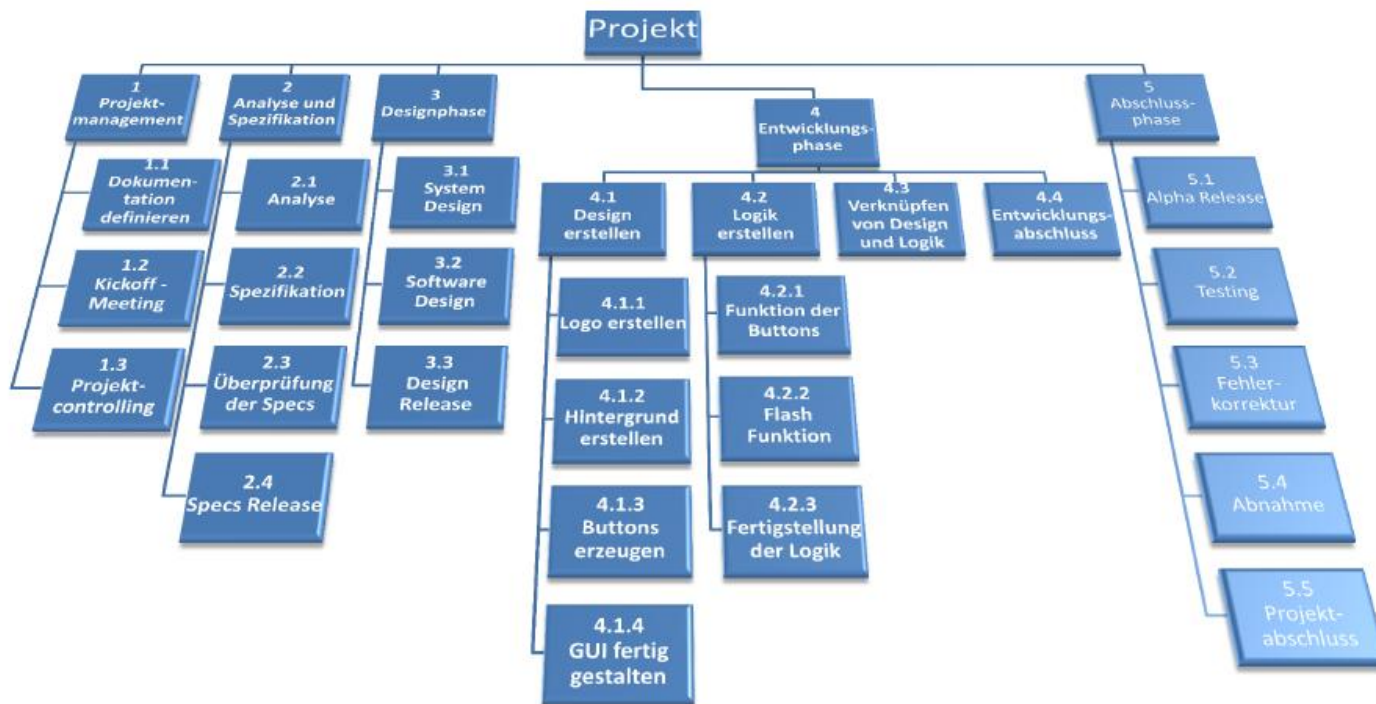
Sollte es zu einer Terminverschiebung kommen, erfolgt eine sofortige Rückmeldung an den Projektleiter, um dementsprechende Maßnahmen zu ergreifen.

### **Maßnahmen bei Terminproblemen:**

Mögliche Maßnahmen für den Fall, dass ein Termin nicht eingehalten werden kann sind:

- Erhöhung der Arbeitszeit
- Zusätzliche Personalressourcen
- Dynamisches verteilen der Aufgaben

## 3.5.2 Projekt Strukturplan



## 4. Nutzenanalyse

### 4.1 Nutzen für den Kunden

Dem Kunden wird die Möglichkeit gegeben, Personen die eine Gehörbehinderung haben, ein Produkt anzubieten. Das Produkt hilft dem Kunden lautlos, Benachrichtigungen bemerkbar zu machen.

### 4.2 Nutzen für das Projektteam

Wir zeigen, dass wir uns auch auf Bedürfnisse von Personen mit Einschränkungen abdecken können. Außerdem beweisen wir auch, dass wir uns mit untrainierten Technologien auseinander setzen können. Durch diese Punkte wollen wir Aufmerksamkeit gewinnen.