

**HTL Anichstraße, Abteilung Wirtschaftsingenieurswesen**

# **Pflichtenheft SyncPipeExplorer**

**Lukas Dejakum & Alexander Ortner**



**2013**

## Inhalt

Zielbestimmungen .....	2
Musskriterien .....	2
Kannkriterien .....	2
Abgrenzungskriterien .....	2
Produkteinsatz.....	2
Anwendungsbereiche.....	2
Zielgruppen.....	2
Betriebsbedingungen .....	2
Produktumgebung.....	3
Software .....	3
Hardware .....	3
Produktschnittstellen .....	3
Produktfunktionen .....	3
Ablaufdiagramm .....	4
Anwendungsdiagramm .....	5
Produktdaten.....	5
Produktleistungen .....	5
Benutzeroberfläche .....	5
Qualitäts-Zielbestimmungen .....	6
Testszenarien und Testfälle.....	6
Entwicklungsumgebung .....	6
Ergänzungen .....	6
Zeitplan.....	6
Netzplan .....	6
Gant Plan .....	8
Arbeitsaufteilung.....	11

## **Zielbestimmungen**

### **Musskriterien**

Der Zugriff auf möglichst alle Daten des Gerätes soll gegeben sein. Dies soll über eine flüssige voll-funktionsfähige GUI passieren. Dazu sollen alle Dateien aus dem Android Gerät ausgelesen und mit grafischer Aufarbeitung in einem Fenster dargestellt werden. Dieser Dateixplorer auf dem PC soll erscheinen sobald ein Android Gerät eingesteckt und erkannt wurde.

Zusätzlich zu den Dateien sollen auch smartphoneinterne Werte wie Prozessorkerntemperaturen, RAM Auslastung, Empfangsstärke und Art, Wifi Signalstärke. Diese Daten werden mittels einer App auf dem Android Gerät gesammelt und dann vom PC ausgelesen.

### **Kannkriterien**

Möglicherweise können auch Werte wie die Sensoren ausgelesen werden. Des Weiteren könnte eine Funktion zur unterschiedlichen Sortierung der Dateien eingebaut werden (nach Datum, Format, etc.). Ob sich das Device mit dem PC auch über WLAN verbinden lässt muss noch überprüft werden. Um anderssprachigen Nutzern die Handhabung zu erleichtern könnte man auch Mehrsprachigkeit implementieren. Zusätzlich wäre eventuell eine Nutzung auf OS X und Linux Systemen, Dank Javas Plattformunabhängigkeit ein mögliches Kannkriterium.

### **Abgrenzungskriterien**

Es geht nicht darum ein Spiel zu programmieren, nicht runnable auf Mac und Linux Systemen. Es ist keine Optimierungssoftware sondern dient rein als Tool zum Anzeigen Dateien und Daten. Diese Software soll nur für Android Geräte, nicht aber für Windows und IOs Geräte ausgelegt werden.

## **Produkteinsatz**

### **Anwendungsbereiche**

Das Produkt soll zum Einsatz kommen, wenn es darum geht Dateien des Android Geräts auf einem PC zu betrachten. Neben den Dateien soll der Nutzer auch andere wichtige Daten über sein Android Gerät auf dem Computer einsehen können.

### **Zielgruppen**

Zielgruppen, sind alle Personen, welche ein Android Gerät besitzen und dessen Dateien und Daten auch einfach auf dem PC betrachten und sortieren wollen. Auch wird durch dieses Tool eine bessere Übersicht über die Dateien gegeben sein. Eine Aussage über das Alter des Nutzers lässt sich an dieser Stelle schwer treffen, da heutzutage sowohl ältere als auch jüngere Personen mit PCs und Smartphones umgehen können.

### **Betriebsbedingungen**

Um das Tool verwenden zu können werden nur ein PC mit Windows7, oder aufwärts, jre1.6 und ein Android Gerät benötigt. Auf dem Android Gerät muss mindestens Android 4.0 laufen um die App verwenden zu können.

## Produktumgebung

### Software

Benötigt werden:

- Windows 7, oder aufwärts
- Jre1.6
- Android 4.0, oder aufwärts

### Hardware

Benötigt werden:

- Ein PC
- Ein Android Gerät
- Ein micro-USB Kabel

Auf dieser Hardware soll die im Punkt „Software“ genannte Software installiert sein und einwandfrei laufen.

### Produktschnittstellen

Um eine Verbindung der beiden Geräte herstellen zu können werden Hardwareseitig ein micro-USB Kabel sowie ein funktionierender USB Port und softwareseitig die sogenannten „ADB“ von Google benötigt. Diese ermöglichen es eine Session aufzubauen und Daten zwischen den Geräten zu verschieben.

### Produktfunktionen

- Zugriff auf alle Dateien auf dem Android Gerät
- Übersicht über androidinterne Werte
- Eine GUI soll erscheinen sobald ein Android Gerät verbunden wurde
- Es soll eine Sortierung der Dateien möglich sein
- Das Tool soll schnell agieren
- Es soll zu jedem benutzungszeitpunkt volle Übersichtlichkeit gegeben sein

## Ablaufdiagramm

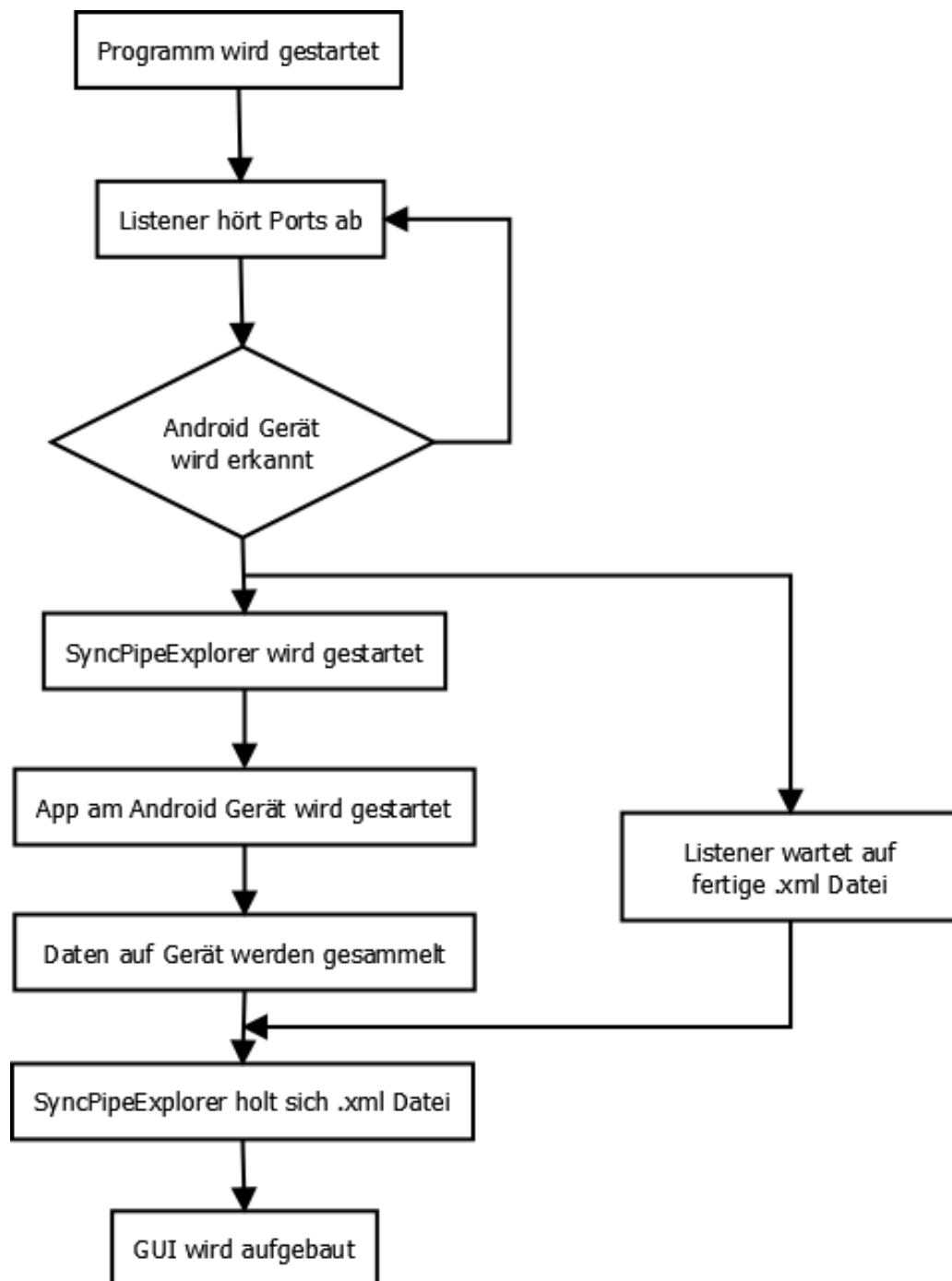


Abbildung 1: Das interne Ablaufdiagramm des SyncPipeExplorer

## Anwendungsdiagramm

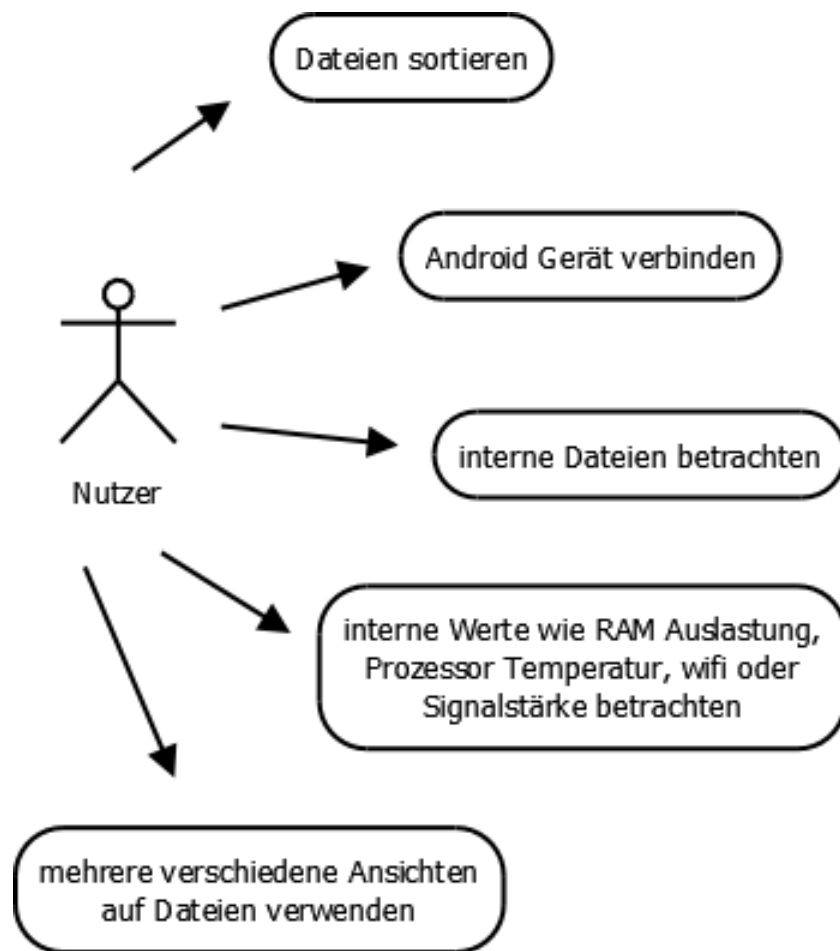


Abbildung 2: Das Anwendungsdiagramm für den Nutzer des SyncPipeExplorers

## Produktdaten

Das Urheberrecht liegt bei Lukas Dejakum und Alexander Ortner. Das Tool soll den Namen „SyncPipeExplorer“ tragen und als .jar oder .exe ausgeliefert werden. Der Fertigstellungstermin soll der Jänner 2014 sein. Die Dateigröße sollte 10 MB nicht überschreiten.

## Produktleistungen

Das Produkt soll schnell und flüssig arbeiten, Mehrsprachigkeit bieten (Deutsch und Englisch) und dem Nutzer eine übersichtliche einfache Oberfläche bieten.

## Benutzeroberfläche

Die Benutzeroberfläche soll sowohl ansprechend sein, als auch eine Übersicht über alle zur Verfügung gestellten Funktionen des Programmes geben. Durch das Nutzen eines Look&Feel Pakets (NimROD) soll die GUI auf dem PC an das Android interne Design Holo.dark angelehnt werden. (Grau, Schwarz, Hellblau).

## Qualitäts-Zielbestimmungen

Der Grad an Benutzerfreundlichkeit soll durch die einfache und übersichtliche Gestaltung sehr hoch liegen und dadurch soll dem Nutzer eine Handhabung ohne Einarbeitungszeit ermöglicht werden.

## Testszenarien und Testfälle

Um unser Projekt testen zu können und die Fehlersuche zu vereinfachen, wird unsere Software in einer Beta Version unseren freundlichen Klassenkollegen zur Verfügung gestellt, welche uns bei den Testszenarien und Testfällen unserer Software unterstützen und uns auf Fehler im System aufmerksam machen können.

## Entwicklungsumgebung

Um ein qualitativ hochwertiges Produkt herstellen zu können, wurden folgende Entwicklungsumgebungen für die Erstellung der Software verwendet:

- Eclipse
- Android SDK
- GitHub

## Ergänzungen

Bezüglich der Lizenzen und rechtlichen Richtlinien müssen noch folgende Punkte beachtet werden:

- Lizenz für die verwendete Java Look and Feel (NimROD)
- eventuelle Icons und Bilder Lizenzen

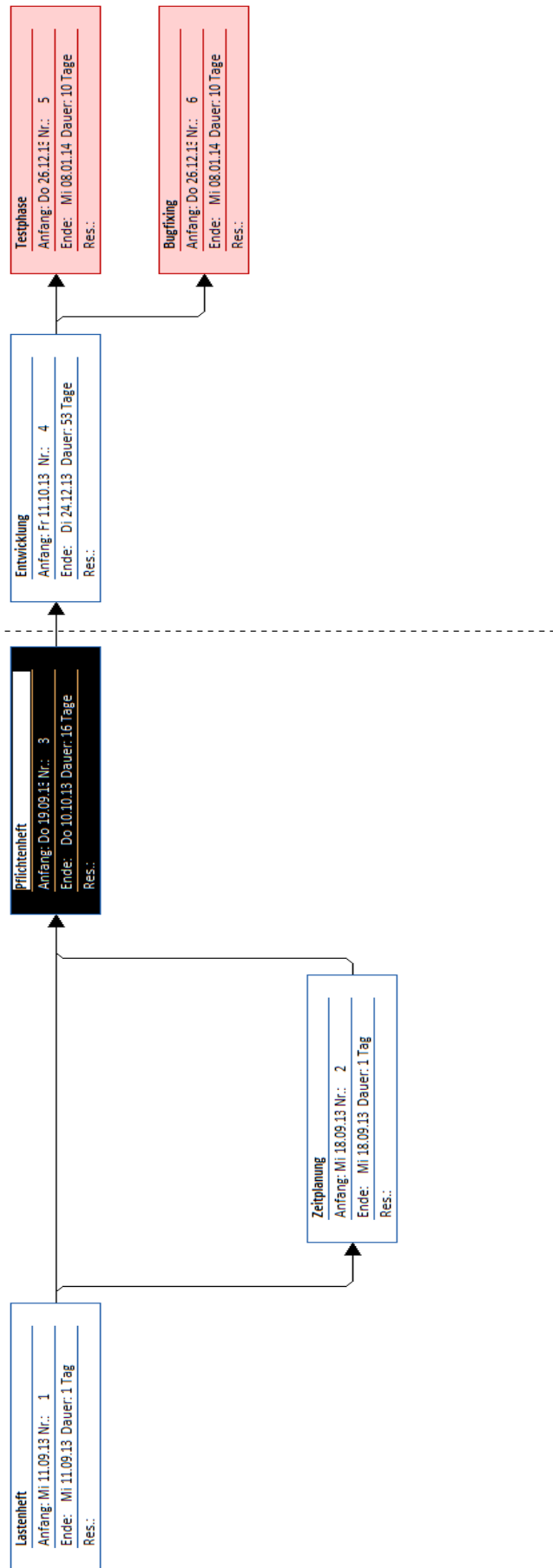
## Zeitplan

Die Zeitplanung beginnt am 11.09.13 mit dem Lastenheft/Anwendungsanalyse und endet am 08.01.14 mit Ende der Testphase und dem Bugfixing.

## Netzplan

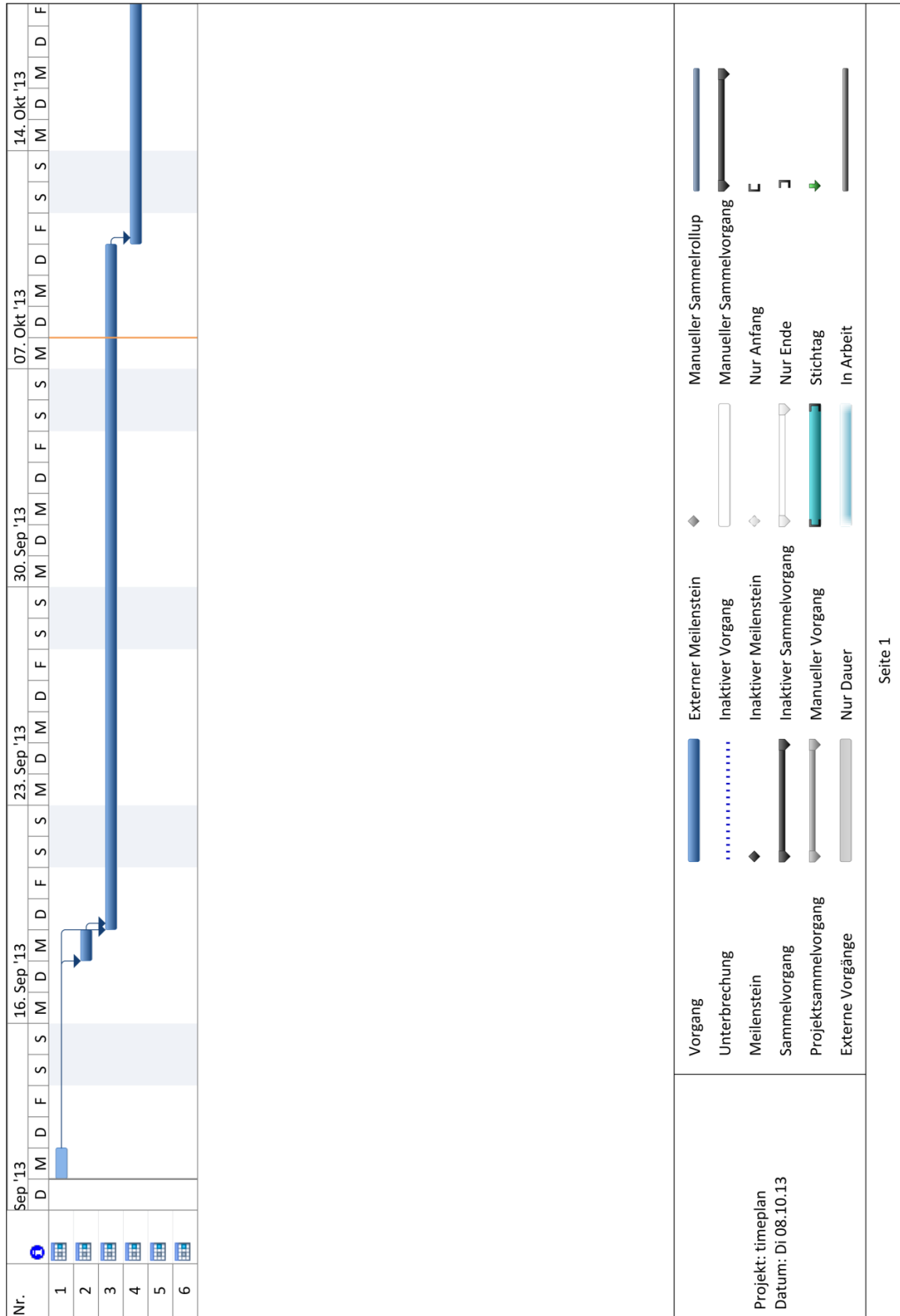
Der Netzplan stellt die einzelnen Vorgänge in Abhängigkeit voneinander dar. Dabei lässt sich sowohl der kritische Pfad, als auch der Anfangs- und Fertigstellungstermin ablesen.

Vorgangsname	Dauer	Anfang	Fertig stellen	Vorgänger
Lastenheft	1 Tag	Mi 11.09.13	Mi 11.09.13	
Zeitplanung	1 Tag	Mi 18.09.13	Mi 18.09.13	1
Pflichtenheft	16 Tage	Do 19.09.13	Do 10.10.13	1;2
Entwicklung	53 Tage	Fr 11.10.13	Di 24.12.13	3
Testphase	10 Tage	Do 26.12.13	Mi 08.01.14	4
Bugfixing	10 Tage	Do 26.12.13	Mi 08.01.14	4





## Gant Plan







## Arbeitsaufteilung

Grundsätzlich wird das Projekt in zwei große Arbeitsbereiche aufgespalten: Das Entwickeln der Android App und das Entwickeln des Datei Explorers. Dabei wird sich Lukas Dejakum um die App und Alexander Ortner um die Desktop Anwendung kümmern. Sollte einer der beiden früher als der seine Arbeit abgeschlossen haben, so soll dieser beim jeweils anderen Arbeitsbereich zum Einsatz kommen.

Die beiden Unterprojekte werden dann, mithilfe der ADB als Schnittstelle, nach Fertigstellung der Arbeit, zusammengefügt.