

# **Realisierungsbericht**

Niklaus Hofer, Lukas Knöpfel, Kaleb Tschabold

May 9, 2011

<b>Status</b>	In Arbeit/ <b>In Prüfung</b> / Abgeschlossen
<b>Projektname</b>	Projektexplorer
<b>Projektleiter</b>	Lukas Knöpfel
<b>Auftraggeber</b>	M. Frieden, GIBB
<b>Autoren</b>	Kaleb Tschabold, Lukas Knöpfel, Niklaus Hofer
<b>Verteiler</b>	Lukas Knöpfel, Kaleb Tschabold, Niklaus Hofer

### Änderungskontrolle, Prüfung, Genehmigung

Version	Datum	Beschreibung, Bemerkung	Name oder Rolle
0.1	08.02.2011	Gesammelten Text einfügen	Kaleb Tschabold
0.9	08.02.2011	Abgabebereit	Kaleb Tschabold
0.99	May 9, 2011	Transfer nach $\text{\LaTeX}$	Niklaus Hofer
1.0	May 9, 2011	Korrekturen	Niklaus Hofer

### Definitionen und Abkürzungen

Begriff/ Abkürzung	Bedeutung
CLI	Command Line Interface

## Contents

<b>1</b>	<b>Zweck des Dokuments</b>	<b>5</b>
<b>2</b>	<b>Technische Detailspezifikation</b>	<b>5</b>
2.1	Innere Struktur	5
2.1.1	Lösungsvorschläge für die Struktur des Systemdesigns	5
2.1.1.1	GUI	5
2.1.1.2	Datenstruktur	5
2.1.2	Struktur des Systemdesigns	5
2.1.3	Beschreibung der Elemente	5
2.2	Schnittstellendefinition	6
2.3	Datenmodell	6
2.3.1	Datenbank	6
2.3.2	File-object	6
2.4	Sicherheit	6
2.5	Anforderungszuordnung	6
<b>3</b>	<b>Systemdokumentation</b>	<b>6</b>
3.1	Inline-Dokumentation	6
3.2	Benutzerhandbuch	6
3.2.1	Systemübersicht	6
3.2.1.1	Aufgabengebiet des Programms	6
3.2.1.2	Programmoberfläche	6
3.2.1.3	Anmerkungen zur korrekten Verwendung unter dem Aspekt der Sicherheit	6
3.2.2	Anwenderfunktionalität	6
3.2.2.1	Ansicht wechseln	6
3.2.2.2	Verzeichnis wechseln	7
3.2.2.3	Hisotry	7
3.2.2.4	Dateien Öffnen	7
3.2.2.5	Tags hinzufügen	7
3.2.2.6	Tags entfernen	7
3.2.2.7	Dateien anhand der Tags durchsuchen	7
3.2.2.8	Fehlermeldungen	7
3.3	Supporthandbuch	7
3.3.1	Massnahmen bei Benutzerproblemen	7
3.3.2	Massnahmen bei technischen Problemen	7
3.3.3	Anhang zum Supporthandbuch	7
<b>4</b>	<b>Systemtest</b>	<b>7</b>
4.1	Testspezifikation	7
4.1.1	Kritikalität der Funktionseinheit	7
4.1.2	Testanforderungen	7
4.1.3	Testverfahren	7
4.1.4	Testkriterine	7
4.1.5	Testfälle	7
4.2	Testprozedur	7
4.2.1	Vorbereitung	7
4.2.2	Durchführung	7
4.2.3	Nachbearbeitung	7
4.3	Testprotokoll	7
4.3.1	Testobjekt	7
4.3.2	Testresultate	7
4.3.3	Testauswertung	7
<b>5</b>	<b>Mittelbedarf</b>	<b>7</b>
<b>6</b>	<b>Planung und Organisation</b>	<b>7</b>
<b>7</b>	<b>Wirtschaftlichkeit</b>	<b>7</b>

<b>8</b>	<b>Konsequenzen</b>	<b>7</b>
<b>9</b>	<b>Antrag auf Freigabe der nächsten Projektphase</b>	<b>7</b>

## 1 Zweck des Dokuments

Wir hatten jetzt einige Wochen Zeit um an der Realisierung zu arbeiten. Wir konnten jetzt unsere Programm Spezifikationen noch genauer ausarbeiten, weil wir während dem programmieren gesehen haben was noch verbessert oder ergänzt werden sollte. In diesem Dokument sind jetzt die genauen Informationen zum Programm.

## 2 Technische Detailspezifikation

### 2.1 Innere Struktur

#### 2.1.1 Lösungsvorschläge für die Struktur des Systemdesigns

##### 2.1.1.1 GUI

##### 2.1.1.2 Datenstruktur

#### 2.1.2 Struktur des Systemdesigns

#### 2.1.3 Beschreibung der Elemente

**Main** Wird zum Starten des Programmes aufgerufen. Main.py instanziert alle weiteren Elemente die für das Funktionieren des Programms nötig sind und koordiniert die Kommunikation zwischen den einzelnen Elementen.

**DB** DB.py ist ein interface zur Datenbank. Es nimmt allen anderen Klassen die Aufgabe ab selbst SQL statements ab zu setzen und bietet stattdessen nach aussen hin verschiedene Funktionen an um Daten zu lesen oder zu schreiben.

**Utility** Enthält verschiedene nützliche Methoden die immer mal wieder von einzelnen Teilen des Programmes benötigt werden.

**CLI** Dient als Command-line-interface für das Program. Es nimmt über die Kommandozeile beim Aufruf verschiedene Befehle entgegen die es dann ausführt.

**TagManager** Der Tag Manager ist für kleine Tag Verwaltungsaufgaben zuständig.

**FileManager** Über den FileManager wird auf das Dateisystem zugegriffen. Hier wird aus jeder Datei aus dem File System ein File Objekt erstellt.

**FileSystemListener** Registriert beim Kernel Listener für zu überwachende Ordner. Wird in diesen Ordnern eine Operation ausgeführt (wie das Verschieben, Löschen, Umbenennen oder Erstellen einer Datei), so wird der FileSystemListener vom Kernel darüber in Kenntnis gesetzt, woraufhin er wiederum die nötigen Aktionen auslöst um die Datenbank auf dem aktuellen Stand zu halten.  
Dies soll dazu beitragen, dass möglichst selten Dateien angezeigt werden, die auf Dateisystem-Ebene nicht existieren.

**GUI** Ist für die grafische Darstellung des Programms mittels GTK zuständig. Das GUI ist in der Lage je nach Bedarf eine andere 'View' darzustellen. Direkt nach dem Programmstart wird HierarchicalView dargestellt. Im Betrieb kann jederzeit zwischen 'TagView' und 'HierarchicalView' umgeschaltet werden.  
Dazu kann GUI, per Polymorphismus, eine Klasse aufnehmen die von 'View' erbt.

**TagView** Enthält die Darstellung der Tag-Ansicht und wird von 'GUI' bei Bedarf geladen.

**HierarchicalView** Enthält die Darstellung der hierarchischen Ansicht und wird von 'GUI' bei Bedarf geladen.

**View** Mutterklasse von GUI.

**File** Repräsentiert eine Datei und wird benutzt um Informationen über Dateien zwischen den Elementen des Programms auszutauschen. Für mehr Informationen siehe 2.3.2.

## 2.2 Schnittstellendefinition

## 2.3 Datenmodell

### 2.3.1 Datenbank

### 2.3.2 File-object

## 2.4 Sicherheit

## 2.5 Anforderungszuordnung

1. Main
2. DB
3. Utility
4. CLI
5. TagManager
6. FileManager
7. FileSystemListener
8. GUI
9. TagView
10. HierarchicalView
11. View
12. File

Nr.	Anforderungen	1	2	3	4	5	6	7	8	9	10	11	12
1	Das Programm starten												
2	GUI vorhanden												
3	CLI vorhanden												
4	Tag hinzufügen												
5	Datei auswählen												
6	Versionierung												
7	GUI: Versionierung für Tags												
8	Dateiänderungen werden erkannt												
9	Löschen wird erkannt												
10	Neue Dateien werden erkannt												
11	Hierarchische Anzeige												
12	Tag Anzeige												

## 3 Systemdokumentation

### 3.1 Inline-Dokumentation

### 3.2 Benutzerhandbuch

#### 3.2.1 Systemübersicht

##### 3.2.1.1 Aufgabengebiet des Programms

##### 3.2.1.2 Programmoberfläche

##### 3.2.1.3 Anmerkungen zur korrekten Verwendung unter dem Aspekt der Sicherheit

#### 3.2.2 Anwenderfunktionalität

##### 3.2.2.1 Ansicht wechseln

**3.2.2.2 Verzeichnis wechseln****3.2.2.3 Hisotry****3.2.2.4 Dateien Öffnen****3.2.2.5 Tags hinzufügen****3.2.2.6 Tags entfernen****3.2.2.7 Dateien anhand der Tags durchsuchen****3.2.2.8 Fehlermeldungen****3.3 Supporthandbuch****3.3.1 Massnahmen bei Benutzerproblemen****3.3.2 Massnahmen bei technischen Problemen****3.3.3 Anhang zum Supporthandbuch****4 Systemtest****4.1 Testspezifikation****4.1.1 Kritikalität der Funktionseinheit****4.1.2 Testanforderungen****4.1.3 Testverfahren****4.1.4 Testkriterine****4.1.5 Testfälle****4.2 Testprozedur****4.2.1 Vorbereitung****4.2.2 Durchführung****4.2.3 Nachbearbeitung****4.3 Testprotokoll****4.3.1 Testobjekt****4.3.2 Testresultate****4.3.3 Testauswertung****5 Mittelbedarf****6 Planung und Organisation****7 Wirtschaftlichkeit****8 Konsequenzen****9 Antrag auf Freigabe der nächsten Projektphase**