|  |  |
| --- | --- |
| **Projektbezeichnung:** | Photo Profis |
| **Erstelldatum:** | Mittwoch, 21. Jänner 2016 |
| **Zustand:** | In Bearbeitung |
| **Projektleiter:** | Eric Buchinger |
| **Beteiligte:** | * Eric Buchinger * Andreas Mayrhofer * Josip Kajic * Anton Thomas * Oscar Yim |



Photo Profis

Änderungsverzeichnis

Version Änderungen

1.0 Ursprüngliches Dokument

2.0 Neu aufgebaut

3.0 Einführung des Änderungsverzeichnisses

4.0 Von Grund auf neu aufgebaut

Inhalt

[Änderungsverzeichnis 1](#_Toc441050055)

[Beschreibung der Ausgangslage 2](#_Toc441050056)

[Ist-Zustand 2](#_Toc441050057)

[Zielsetzung 2](#_Toc441050058)

[Funktionale Anforderungen 2](#_Toc441050059)

[Mengengerüst 3](#_Toc441050060)

[Nicht funktionale Anforderungen 3](#_Toc441050061)

[Risikoakzeptanz 3](#_Toc441050062)

[Graphical User Interface 4](#_Toc441050063)

# Beschreibung der Ausgangslage

Im Laufe der Zeit sammeln sich viele Fotos auf Geräten wie Kameras, Smartphones usw. an. Die Fotos werden meistens auf den Computer importiert und in einem Verzeichnis gespeichert. Da diese Fotos unsortiert sind, sind sie schwer aufzufinden.

Diesem Problem werden wir nachgehen.

# Ist-Zustand

Die Fotos können natürlich vom Benutzer sortiert werden. Da dies aber eine enorme Zeit in Anspruch nimmt, machen das nicht viele. Wir entwickeln ein Programm mit diesen Prozessen:

Kategorisieren

In bestehende Fotoarchive einordnen

**Die Fotos werden**

**Vom Benutzer**

**eingelesen**

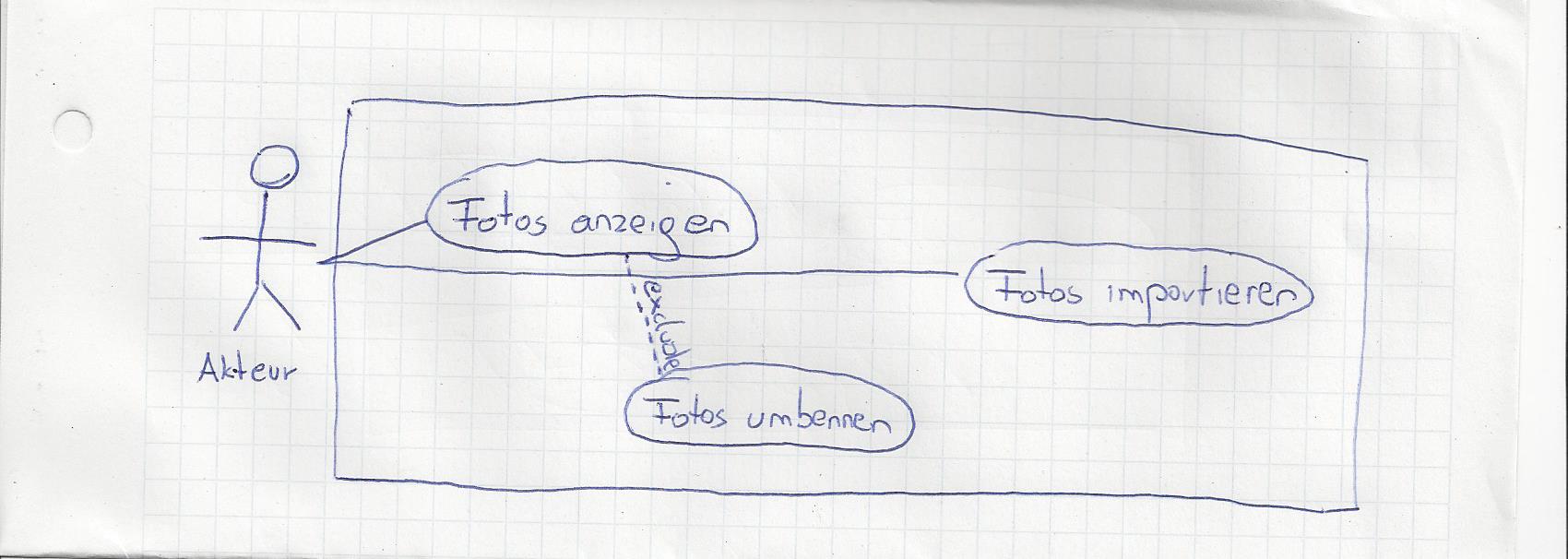
# Zielsetzung

Die Zielsetzung ist ein geordnetes und übersichtliches Fotoverzeichnis, in dem man die Fotos leicht wiederfindet.

Das Ergebnis ist eine Software zur Umbenennung und Sortierung mehrerer Bilder bzw. das Bearbeiten der Meta-Daten.

# Funktionale Anforderungen

Die Funktionalen Anforderungen sind das reibungslose Kategorisieren und Einordnen der zuvor eingelesen Bilder.



*Abbildung 1.1 Use-Case Diagramm*

# Mengengerüst

Datenbestände 🡪Das Programm benötigt Festplattenspeicher zum speichern der Fotos.

Datenbewegungen 🡪Je nach Anzahl der Bilder werden größere Datenmengen pro Tag transportiert.

Z.B 1000 Bilder mit der angenommenen Bildauflösung von 3840 × 2160. Der Speicherbedarf liegt bei ca. 2 Megabyte. So werden 2000 MB ~ 2GB bewegt.

# Nicht funktionale Anforderungen

* Usability -> Das Programm ist einfach zu bedienen und wird ohne Benutzerhandbuch ausgeliefert, da alles sehr übersichtlich und selbsterklärend ist.
* Performance -> Die Fotos werden mit der maximalen Festplattengeschwindigkeit verschoben und das Sortieren, Umbenennen etc. erfolgt in unter 1 Sekunde
* Robustheit -> Es gibt keine Abstürze da ein umfangreiches Exceptionhandling (Behandlung eines auftretenden Problems innerhalb der Laufzeit) implementiert ist.
* Sicherheit -> Hier ist keine Authentifizierung oder Autorisierung erforderlich da der Benutzer keine Daten löschen kann.
* Portabilität -> Es läuft auf Windows und Linux basierten Systemen
* Korrektheit -> Es läuft ohne Probleme.
* Skalierbarkeit -> Man kann neue Funktionen einfach implementieren da Objektorientiert programmiert wird.
* Wartbarkeit -> Der Code ist einfach zu lesen und kann ohne Probleme bearbeitet werden.

# Risikoakzeptanz

Ein Risiko, wodurch das Projekt scheitern könnte, ist die Zeit. Ein weiteres Risiko ist fehlende Fähigkeit das Programm zu implementieren.

# Graphical User Interface

