

# Anforderungsspezifikation CTI

Projektteam 10

Lukas Seglias  
Luca Ritz

V. 1.0 30.11.2019

# Inhaltsverzeichnis

1	Zweck des Dokuments	3
2	Vision	3
3	Projektzielsetzung	3
3.1	Ausgangslage	3
3.2	Schwachstellen der Ausgangslage	3
3.3	Stakeholder	4
3.4	Projektziele	4
4	Systemabgrenzung	5
4.1	Prozessumfeld	5
4.2	Systemumfeld	6
4.3	Nicht unterstützte Projektziele	7
5	Anforderungen	7
5.1	Quellen und Vorgehen	7
5.2	Funktionale Anforderungen	8
5.2.1	Use Cases	8
5.2.2	Detaillierte Anforderungen	10
5.3	Qualitätsanforderungen	10
5.4	Randbedingungen	11
5.5	Datenmodell	11
6	Glossar	11
7	Anhang	12
7.1	Definition of Ready – Checklist	12
8	Versionskontrolle	12

# 1 Zweck des Dokuments

Dieses Dokument beschreibt die Ziele und Anforderungen für das Projekt CTI.

## 2 Vision

In vielen Bereichen, wo mit Vouchern gearbeitet wird, ist es Teil des Prozesses, dass ein Voucher manuell von einer Person verarbeitet werden muss. So muss jedes Mal erkannt werden, um welche Art von Voucher es sich handelt und die Informationen, welche mit dem Voucher assoziiert sind, ausgelesen und in digitale Systeme überführt werden. Diese gängigen Teilschritte der Erkennung und des Auslesens von Informationen sollen automatisiert werden. Diese Teilschritte sollen in einem zu erarbeitenden Softwarebaustein gelöst werden, um diese Funktionalität für verschiedene konkrete Anwendungsbereiche bereitzustellen.

## 3 Projektzielsetzung

### 3.1 Ausgangslage

In vielen Bereichen des Lebens werden Dienstleistungen mithilfe von in Papierform vorliegenden Vouchern eingelöst. Voucher in elektronischer Form zu verkaufen und einzulösen ist unerwünscht. Es soll die Freiheit gewahrt bleiben, mit Papier zu arbeiten. Trotzdem müssen die Voucher und deren assoziierten Informationen mit digitalen Systemen abgeglichen werden.

Meist muss eine Person die repetitive Arbeit erledigen, Voucher von Kunden entgegenzunehmen, zu prüfen, abgedruckte Informationen abzugleichen und in digitale Systeme zu überführen.

Eine mögliche konkrete Ausgangslage wird anhand des Beispiels der Bärnhäckt 2019 Challenge der Jungfraubahnen veranschaulicht:

Ein Kunde kauft bei einem externen Reiseveranstalter eine Reise mitsamt Aktivitäten, wofür ihm ein Voucher ausgestellt wird. Diese Voucher werden vom Reiseveranstalter mehr oder weniger frei gestaltet, einzig der Tourcode, der Name und das Logo des Reiseveranstalters muss aufgedruckt sein. Der Tourcode hält fest, welche Dienstleistungen (z.B. Restaurant, Skilift, Übernachtung) mit dem Voucher bezogen werden können.

Der Kunde löst diesen Voucher am Schalter der Jungfraubahnen gegen gültige Tickets ein. Das Schalterpersonal prüft den Voucher auf seine Gültigkeit (bekannter Reiseveranstalter). Anschliessend werden mithilfe des aufgedruckten Tourcodes die dazugehörigen Dienstleistungen gesucht und die erwähnten Tickets ausgestellt. Zu jedem eingelösten Voucher wird ein Ausgabebeleg ausgedruckt, auf welchem die ausgestellten Tickets aufgelistet sind.

Jeden Abend stellt das Schalterpersonal die eingelösten Voucher und Ausgabebelege den Produktmanagern in der Direktion zu. Jeden Morgen kontrollieren die Produktmanager die Voucher mithilfe der Ausgabebelege auf deren Richtigkeit und legen diese ab. Dabei muss manuell digitalisiert werden, welcher Reiseveranstalter welche Touren (sprich Tourcodes) vermittelt hat. Die getätigten Transaktionen (Ticketausgaben) werden monatlich dem Reiseveranstalter in Rechnung gestellt und dessen Provision abgerechnet. Am Ende des Monats werden alle in Papierform vorliegenden Voucher zusammengepackt und an den Reiseveranstalter versendet.

### 3.2 Schwachstellen der Ausgangslage

Die manuelle Entgegennahme und Verarbeitung eines Vouchers sind weder wirtschaftlich sinnvoll noch erfüllend für die betroffene Person. Ohne diese Tätigkeit hätte die Person mehr Zeit für den persönlichen Kontakt zu den Kunden oder für andere Arbeiten, welche am besten von einem Menschen erledigt werden.

Die in Papierform vorliegenden Voucher sind meist Teil eines grösseren Prozesses einer Unternehmung. Dadurch, dass der Teilprozess der Voucher nicht automatisiert digitalisiert wird, können spätere Prozesse, beispielsweise in den Bereichen Buchhaltung, Rechnungsstellung, Auswertungen, verzögert werden.

Wenn die Voucher und deren Informationen nicht frühzeitig digitalisiert werden, kann es sein, dass diese verloren gehen können.

### 3.3 Stakeholder

- Peter Lange (Product Owner)
- Lukas Seglias (Entwickler)
- Luca Ritz (Entwickler)
- Lukas Seglias (Auftraggeber)
- Luca Ritz (Auftraggeber)

### 3.4 Projektziele

1. Das Teilproblem der Erkennung eines Vouchers und das Auslesen der darauf sichtbaren Information soll gelöst werden. Dieser Prozess soll automatisiert werden.
2. Es soll möglich sein, Voucher-Vorlagen zu definieren, welche das Design und die relevanten Informationen eines Vouchers definieren.
3. Mithilfe der Voucher-Vorlagen soll erkannt werden können, ob es sich bei einem in Papierform vorliegenden Voucher um einen der vordefinierten Voucher handelt.
4. Mithilfe der Voucher-Vorlagen sollen die relevanten Informationen eines in Papierform vorliegenden Voucher ausgelesen werden können.
5. Die in Papierform vorliegenden Voucher werden dem zu erarbeitenden System in Form eines Bildes des Vouchers übergeben. Das bedeutet, dass eine Lösung im Bereich des maschinellen Sehens und der digitalen Bildverarbeitung zu erarbeiten ist.
6. Das System unterstützt die Betriebssysteme Windows und Linux und die gängigen Programmiersprachen C++ und Java.
7. Es werden mithilfe von Testdaten aussagekräftige Metriken berechnet, um die Zuverlässigkeit des Systems zu bestätigen.

Anhand eines Beispiels soll das Prinzip veranschaulicht werden:

In Abbildung 1 sind mögliche Voucher abgebildet. Diese sind beliebig gestaltet und enthalten einige für den konkreten Anwendungsfall relevante Informationen.

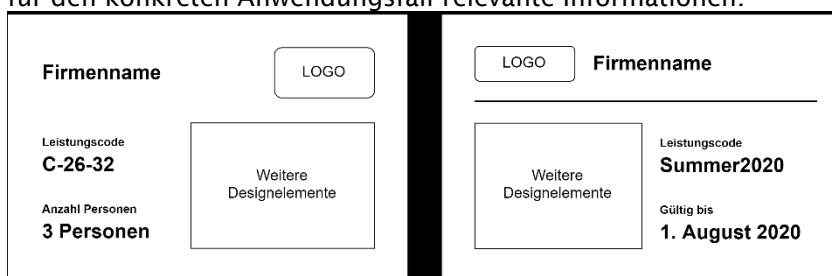


Abbildung 1: Verschiedene Voucher

Anhand des Aussehens des in Papierform vorliegenden Voucher kann erkannt werden, ob es sich um einen der beiden Voucher handelt. Nachdem bestimmt wurde, welcher Voucher vorliegt, sollen die relevanten Informationen, verdeutlicht in Abbildung 2, ausgelesen werden.

Firmenname  
**Firmenname**

Leistungscodex  
**C-26-32**

Anzahl Personen  
**3 Personen**

LOGO

Weitere Designelemente

Abbildung 2: Relevante Informationen eines Vouchers

Für die Erkennung des Vouchers und das Auslesen der Informationen muss es eine Vorlage geben, mit der ein in Papierform vorliegender Voucher verglichen werden kann. Die Vorlage enthält das Design und die Regionen mit den verschiedenen relevanten Informationen, siehe Abbildung 3.

Firmenname

Leistungscodex

Anzahl Personen

LOGO

Weitere Designelemente

Abbildung 3: Voucher-Vorlage

Bei der Entgegennahme eines in Papierform vorliegenden Voucher wird ein Bild davon erstellt, die dazugehörige Vorlage erkannt und die damit assoziierten Informationen ausgelesen, siehe Abbildung 4.

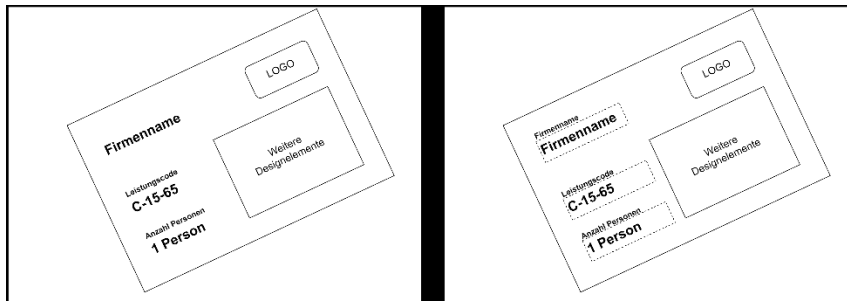


Abbildung 4: Voucher und auszulesende Informationen

## 4 Systemabgrenzung

### 4.1 Prozessumfeld

Der genaue Prozess, in welchem das zu erarbeitende System eingesetzt wird, ist unbekannt. Bevor ein Voucher erkannt und ausgelesen werden kann, muss eine passende Vorlage an das System übergeben werden. Dies wird wie in Abbildung 5 von der Anwendung durchgeführt.

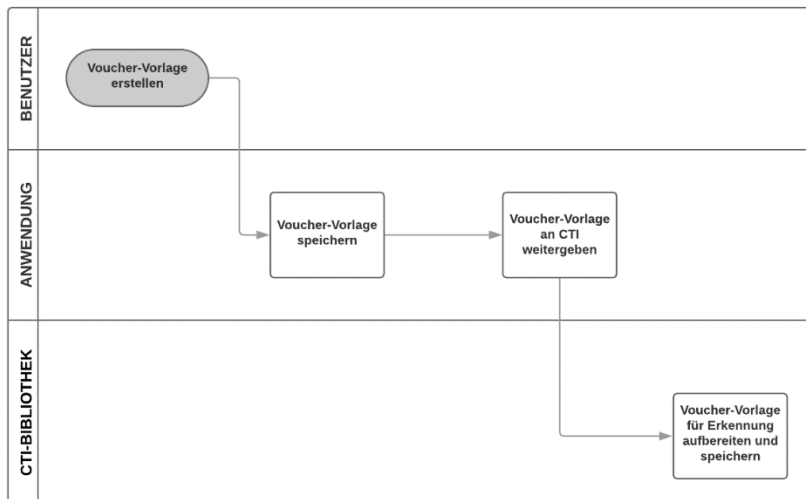


Abbildung 5: Prozess Voucher-Vorlagen

In Abbildung 6 wird verdeutlicht, wie die Verarbeitung eines in Papierform vorliegenden Voucher abläuft. So ist die Anwendung frei, welche Schritte sie vor und nach der Verwendung der Library tätigt. Ebenfalls zu sehen sind die beiden Kernkompetenzen der Library, namentlich die Erkennung und das Auslesen der Informationen eines Vouchers.

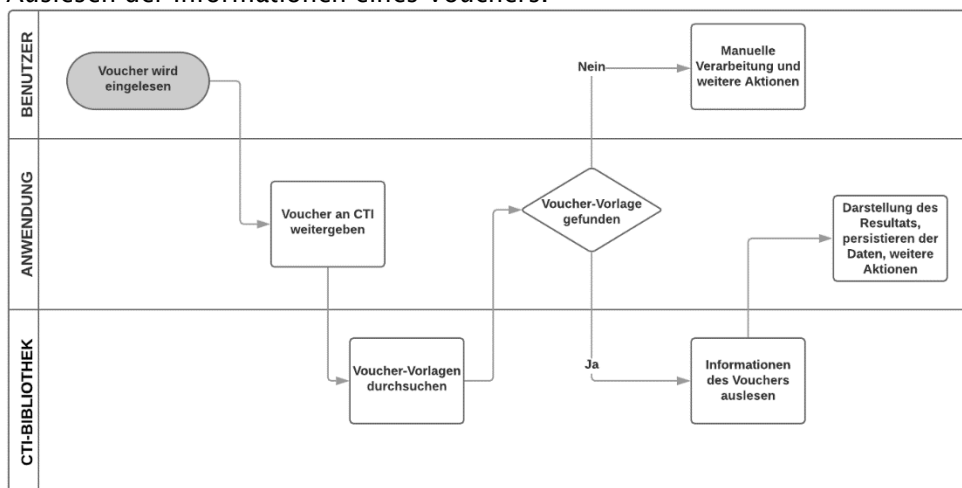


Abbildung 6: Prozess Voucher erkennen und auslesen

## 4.2 Systemumfeld

Der Softwarebaustein CTI, im folgenden Library genannt, welcher im Rahmen dieses Projekts entwickelt wird, kann in konkreten Geschäftsanwendungen integriert werden. Die Library wird mit der konkreten Anwendungssoftware ausgeliefert und diese kommuniziert mit Umsystemen, welche evtl. am weiteren Prozess beteiligt sind. Die Anwendungssoftware kommuniziert mit der Library und verwendet ihre Funktionalität. Inwiefern die Anwendungssoftware und sonstige beteiligte Umsysteme strukturiert sind oder miteinander kommunizieren, ist für CTI nicht von Bedeutung.

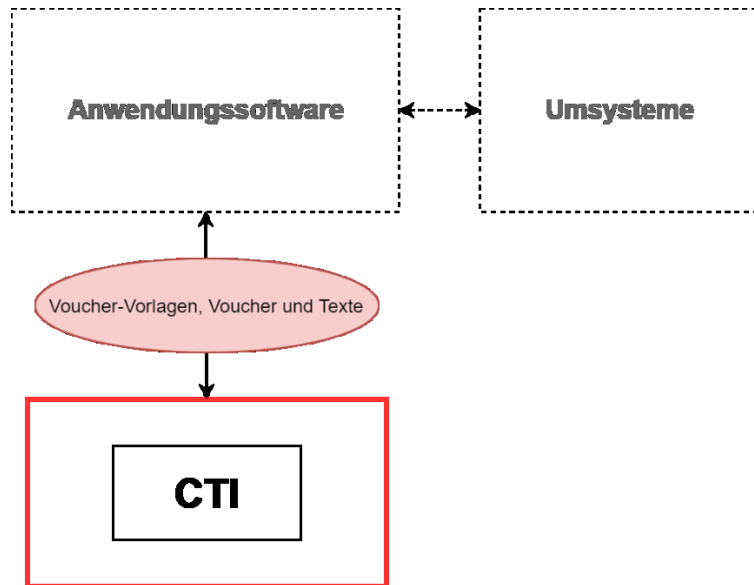


Abbildung 7: Systemumfeld

### 4.3 Nicht unterstützte Projektziele

Es geht um die Erarbeitung einer Library, auf der weitere Anwendungen aufsetzen können, welche diese in konkrete Geschäftsanwendungen integrieren. So ist die Implementation einer betriebsbereiten Komplettlösung für einen spezifischen Typ von Vouchern nicht das Ziel, sondern die Erarbeitung einer Software-Library. Durch die Lösung dieses Teilproblems werden erst weitere Anwendungen ermöglicht, welche sich mit dem gesamten Prozess einer Unternehmung, beispielsweise inkl. des Marketings, der Rechnungsstellung und der Buchhaltung etc., beschäftigen oder auch die Anbindung an Umsysteme lösen.

Im Rahmen des Projekts wird keine grafische Benutzeroberfläche implementiert.

## 5 Anforderungen

### 5.1 Quellen und Vorgehen

Die funktionalen und nichtfunktionalen Anforderungen an das zu erarbeitende System leiten sich aus der Projektvision und den Projektzielen ab. Sie werden vom Projektteam erarbeitet.

## 5.2 Funktionale Anforderungen

### 5.2.1 Use Cases

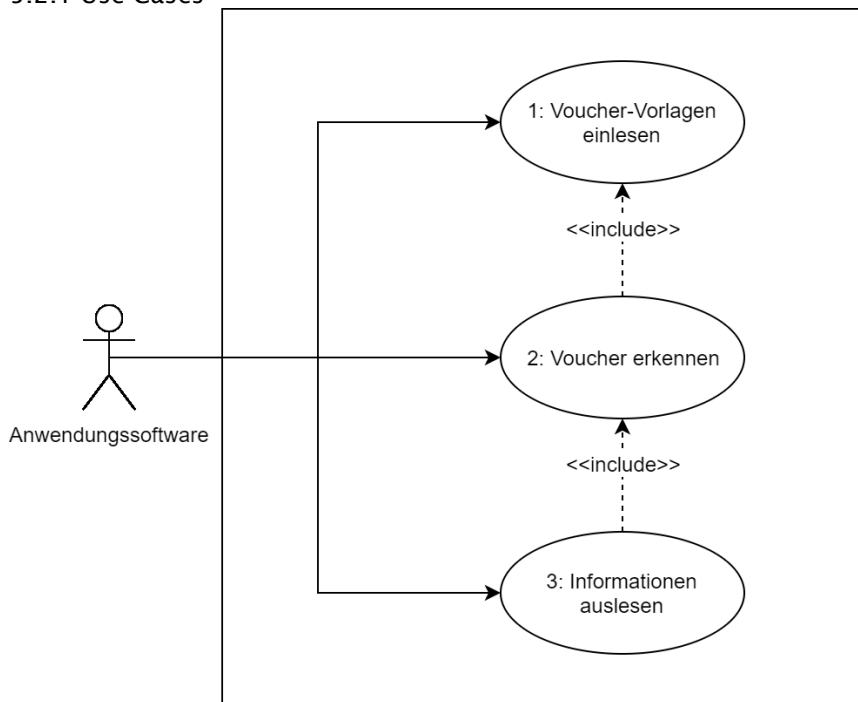


Abbildung 8: Use Case Diagramm

#### Voucher-Vorlagen einlesen

Nr., Name	1, Voucher-Vorlagen einlesen
Szenario	Mögliche Voucher-Vorlagen erfassen
Kurzbeschreibung	Voucher-Vorlagen sollen initial erfasst oder geändert werden.
Beteiligte Akteure	Anwendungssoftware
Auslöser / Vorbedingung	Voucher-Vorlagen vorhanden
Ergebnisse / Nachbedingungen	Voucher-Vorlagen sind bereit für die Erkennung.

#### Ablauf

Nr	Wer	Was
1.1	Anwendungssoftware	Bereitet Voucher-Vorlagen auf, indem die Identifikation, das Bild und einige Textbereiche geladen werden.
1.2	Anwendungssoftware	Übergibt vorbereitete Voucher-Vorlagen an CTI.
1.3	CTI	Verarbeitet und speichert die Voucher-Vorlagen für die Erkennung ab.

#### Ausnahmen, Varianten

Nr	Wer	Was
1.3.1	CTI	Wenn Voucher-Vorlage mit der gegebenen Identifikation bereits vorhanden ist, wird sie ersetzt.
1.3.2	CTI	Wenn das Bild der Voucher-Vorlage nicht geladen werden konnte, wird ein Fehler erzeugt.
1.3.3	CTI	Wenn die Identifikation der Voucher-Vorlage nicht übergeben wurde, wird ein Fehler erzeugt.
1.3.4	CTI	Wenn die Identifikation eines Textbereichs nicht übergeben wurde, wird ein Fehler erzeugt.
1.3.5	CTI	Wenn ein Textbereich ausserhalb des Bildes liegt, wird ein Fehler erzeugt.
1.3.6	CTI	Wenn ein Textbereich ungültig ist (Breite=0 / Höhe=0), wird ein Fehler erzeugt.



### Voucher erkennen

Nr., Name	2, Voucher erkennen
Szenario	Ein in Papierform vorliegender Voucher soll erkannt werden.
Kurzbeschreibung	Anhand der eingelesenen Voucher-Vorlagen soll ein konkreter Voucher erkannt werden.
Beteiligte Akteure	Anwendungssoftware
Auslöser / Vorbedingung	Voucher-Vorlagen wurden eingelesen (UC 1).
Ergebnisse / Nachbedingungen	Der Voucher wurde korrekt erkannt. Dazu zählt auch eine korrekte Nicht-Erkennung, falls keine passende Vorlage vorliegt.

### Ablauf

Nr	Wer	Was
1.1	Anwendungssoftware	Bereitet Bild eines Vouchers auf.
1.2	Anwendungssoftware	Übergibt Bild des Vouchers an CTI.
1.3	CTI	Verarbeitet das Bild für die weitere Verwendung.
1.4	CTI	Gleicht das Bild mit den eingelesenen Voucher-Vorlagen ab.
1.5	CTI	Liefert die zum Bild des Vouchers passende Vorlage.

### Ausnahmen, Varianten

Nr	Wer	Was
1.5.1	CTI	Wenn keine der Vorlagen passt, wird keine Vorlage geliefert.

### Informationen auslesen

Nr., Name	3, Informationen auslesen
Szenario	Textuelle Informationen sollen ausgelesen werden.
Kurzbeschreibung	Die Informationen eines Vouchers sollen anhand der in der Voucher-Vorlage hinterlegten Textbereiche ausgelesen werden.
Beteiligte Akteure	Anwendungssoftware
Auslöser / Vorbedingung	Voucher wurde erkannt (UC 2). Der auf dem Voucher sichtbare Text ist nicht handschriftlich und ist nicht verdeckt.
Ergebnisse / Nachbedingungen	Textuelle Informationen sind der Anwendungssoftware bekannt.

### Ablauf

Nr	Wer	Was
1.1	Anwendungssoftware	Bereitet Voucher-Vorlage und Bild eines Vouchers auf.
1.2	Anwendungssoftware	Übergibt vorbereitete Voucher-Vorlage und Bild an CTI.
1.3	CTI	Verarbeitet das Bild für die weitere Verwendung, indem es das Bild in die Position (Drehung, Skalierung) des Vorlage-Bildes überführt (normalisiert).
1.4	CTI	Legt die Textbereiche der Voucher-Vorlage über das normalisierte Bild des Vouchers.
1.5	CTI	Liest die textuellen Informationen jedes Textbereiches aus und liefert sie an die Anwendungssoftware.

### Ausnahmen, Varianten

Nr	Wer	Was
1.5.1	CTI	Wenn gewisse textuelle Informationen nicht ausgelesen werden konnten, werden die Restlichen trotzdem zurückgeliefert.

### 5.2.2 Detaillierte Anforderungen

ID	Status	Prio	Beschreibung
F1.1	Freigegeben	Muss	Verwaltung: Voucher-Vorlagen können eingespielt werden.
F1.2	Freigegeben	Muss	Verwaltung: Auf einer Voucher-Vorlage kann ein Bild eines Voucher-Designs hinterlegt werden.
F1.3	Freigegeben	Muss	Verwaltung: Auf einer Voucher-Vorlage können die Positionen relevanter textueller Informationen hinterlegt werden.
F1.4	Freigegeben	Optional P1	Verwaltung: Voucher-Vorlagen können geändert werden.
F2.1	Freigegeben	Muss	Erkennung: Ein in Papierform vorliegender Voucher kann erkannt werden, indem ein Bild des Vouchers gegen die hinterlegten Voucher-Vorlagen abgeglichen wird. Massgeblich für die Erkennung, dass es sich um denselben Voucher handelt, ist das hinterlegte Voucher-Design.
F2.2	Freigegeben	Muss	Erkennung: Ein Voucher muss auch dann erkannt werden, wenn dieser innerhalb des Bilds maximal 60 Grad gedreht ist.
F2.3	Freigegeben	Optional P1	Erkennung: Ein Voucher muss auch dann erkannt werden, wenn kleine Teile des Vouchers verdeckt sind.
F2.4	Freigegeben	Muss	Erkennung: Ein Voucher muss auch dann erkannt werden, wenn der Voucher maximal ca. 30cm entfernt von der Kamera ist (Skalierung).
F3.1	Freigegeben	Muss	Auslesen: Die textuellen Informationen auf einem in Papierform vorliegenden Voucher können ausgelesen werden. Dafür werden die in den Vorlagen hinterlegten Textpositionen ausgelesen. Voraussetzung dafür ist, dass der Voucher erkannt wurde, also eine passende Vorlage gefunden wurde.
F3.2	Freigegeben	Muss	Auslesen: Die Informationen eines Vouchers müssen auch dann ausgelesen werden, wenn dieser innerhalb des Bilds maximal 60 Grad gedreht ist.
F3.3	Freigegeben	Optional P1	Auslesen: Die Informationen eines Vouchers müssen auch dann ausgelesen werden, wenn Teile des Vouchers, welche nicht die gesuchte Information enthalten, verdeckt sind.
F3.4	Freigegeben	Muss	Auslesen: Die Informationen eines Vouchers müssen auch dann ausgelesen werden, wenn der Voucher maximal ca. 30cm entfernt von der Kamera ist (Skalierung).
F3.5	Freigegeben	Muss	Auslesen: Wenn nur Teile der Informationen eines Vouchers ausgelesen werden konnten, sollen diese der Anwendungssoftware trotzdem zurückgegeben werden.

### 5.3 Qualitätsanforderungen

ID	Status	Prio	Beschreibung
Q1.1	Freigegeben	Muss	Verständlichkeit: Die öffentliche Schnittstelle des Systems ist mit einer verständlichen Dokumentation ausgestattet. Ein C++-Entwickler mit einem Jahr Erfahrung kann die Dokumentation innerhalb von 30 Minuten lesen und verstehen.
Q2.1	Freigegeben	Optional P1	Einfachheit: Eine typische Verwendung der öffentlichen Schnittstelle des Systems ist maximal 15 Zeilen lang.
Q3.1	Freigegeben	Optional P2	Performance: Die Erkennung eines Vouchers bei 10 Voucher-Vorlagen muss durchschnittlich innerhalb von 4 Sekunden abgeschlossen sein.
Q3.2	Freigegeben	Optional P2	Performance: Das Auslesen der Informationen eines Vouchers muss durchschnittlich innerhalb von 3 Sekunden abgeschlossen sein.

Q4.1	Freigegeben	Wunsch	Skalierbarkeit: Das System muss horizontale Skalierung ermöglichen.
Q5.1	Freigegeben	Muss	Korrektheit: Die Treffergenauigkeit (engl. Accuracy) der Erkennung eines Vouchers soll bei 10 Voucher-Vorlagen mind. 85% betragen.
Q5.2	Freigegeben	Muss	Korrektheit: Die Treffergenauigkeit (engl. Accuracy) der ausgelesenen Texte eines Vouchers soll mind. 85% betragen.

## 5.4 Randbedingungen

ID	Status	Prio	Beschreibung
R1.1	Freigegeben	Muss	Technologie: Die öffentliche Schnittstelle ist für C++ 17 und Java 8 verwendbar.
R1.2	Freigegeben	Muss	Technologie: Das System ist auf Windows und Linux lauffähig.

## 5.5 Datenmodell

Nachfolgend ist das Datenmodell aufgeführt, welches für das Verständnis der Anforderungen notwendig ist.

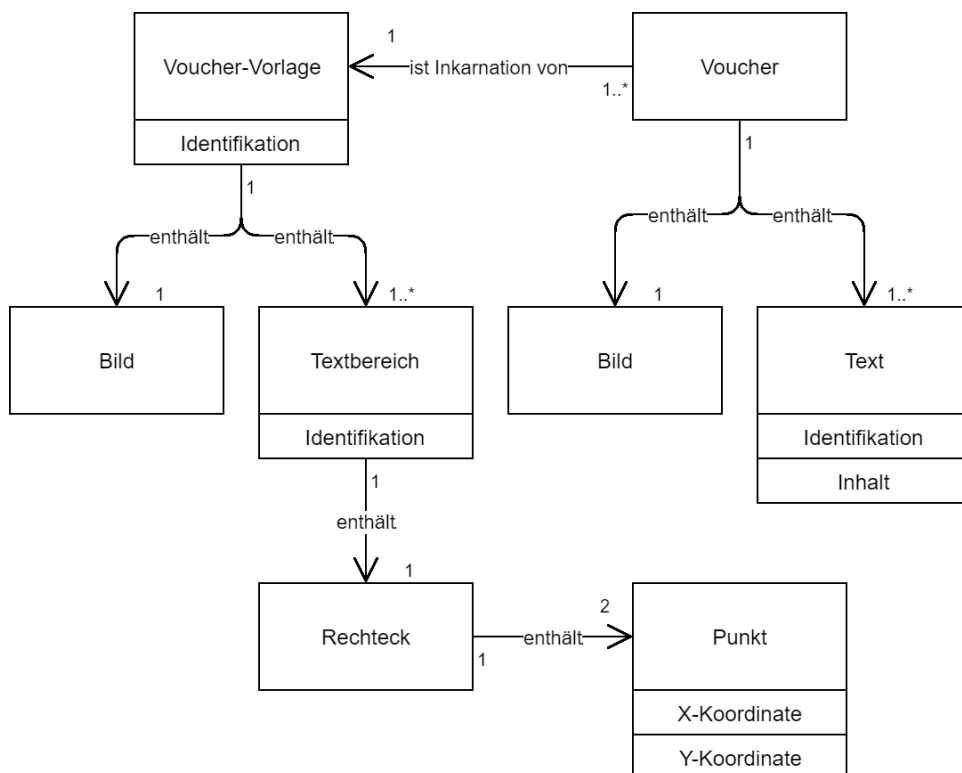


Abbildung 9: Datenmodell

## 6 Glossar

Begriff	Beschreibung
CTI	Computerunterstützte Ticketerkennung und Informationsextraktion

## 7 Anhang

### 7.1 Definition of Ready – Checklist

- Backlog Items sind klein genug, um im Sprint abgeschlossen zu werden
- Backlog Items sind für jeden beteiligten Entwickler klar verständlich
- Jedes Backlog Item hat Akzeptanzkriterien (Min. 1)
- Jedes Backlog Item hat einen Wert im Sinne der Projektziele

## 8 Versionskontrolle

Version	Datum	Beschreibung	Autor
X0.1	27.09.2019	Vision und Projektziele	Lukas Seglias & Luca Ritz
X0.2	02.10.2019	Überarbeitung Vision und Projektziele	Lukas Seglias & Luca Ritz
X0.3	07.10.2019	Systemabgrenzung, Anforderungen	Lukas Seglias & Luca Ritz
X0.4	15.10.2019	Überarbeitung Systemabgrenzung, Ergänzung Anforderungen, Use Cases, Datenmodell	Lukas Seglias & Luca Ritz
X0.5	18.10.2019	Überarbeitung Use Cases, Datenmodell	Lukas Seglias & Luca Ritz
X0.6	23.10.2019	Überarbeitung Anforderungen, Definition of Ready	Lukas Seglias & Luca Ritz
X0.7	22.11.2019	Überarbeitung Peer Review Team	Lukas Seglias & Luca Ritz
X1.0	30.11.2019	Finale Version	Lukas Seglias & Luca Ritz