

Hochschule Worms

Fachbereich Informatik

Studiengang Angewandte Informatik - dual (B.Sc.)

Thesis zur Erlangung des akademischen Grades

# Bachelor of Science

Wie kann die Anmeldung und Zustellung von Weinen für  
Weinproben des Regionalverbunds für Weine in der  
Weinregion Mosel effizient und profitabel durch eine  
TYPO3-Erweiterung realisiert werden?

Vorgelegt von

Leon Etienne, 676838

inf3642@hs-worms.de

mit Medienagenten Stange & Ziegler oHG

31. März 2023

Bad Dürkheim

Erstbetreuer: Prof. Dr. Volker Schwarzer

Zweitbetreuerin: Prof. Dr. Elisabeth Heinemann

# Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist noch nicht in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung vorgelegt worden.

Bad Dürkheim, 31. März 2023

Leon Etienne \_\_\_\_\_

---

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>I</b>
<b>Tabellenverzeichnis</b>	<b>II</b>
<b>Abkürzungsverzeichnis</b>	<b>III</b>
<b>Glossar</b>	<b>IV</b>
<b>1. Einleitung</b>	<b>1</b>
1.1. Problemstellung . . . . .	1
1.2. Zielsetzung . . . . .	2
<b>2. Stand der Forschung</b>	<b>4</b>
2.1. Modell nach Parviainen et al. . . . .	4
2.2. Modell nach Verhoef et al. . . . .	7
2.3. Abwägung in Bezug auf die Problemstellung . . . . .	7
2.3.1. Parviainen et al. . . . .	8
2.3.2. Verhoef et al. . . . .	8
2.3.3. Fazit . . . . .	8
<b>3. Stand der Technik</b>	<b>10</b>
3.1. Die bestehende Webseite . . . . .	10
3.2. QR-Code-Bibliotheken . . . . .	10
3.2.1. JavaScript-Implementationen . . . . .	11
3.2.2. PHP-Implementationen . . . . .	12
3.2.3. Subjektiver Vergleich im Bezug auf die Problemstellung . . . . .	14
3.2.4. Fazit . . . . .	18
3.3. PDF-Generator . . . . .	19
<b>4. Anforderungserfassung</b>	<b>20</b>
4.1. Interview mit Product Owner . . . . .	21

---

4.2. Online-Fragebögen für Stakeholder . . . . .	21
4.3. Erkenntnisse . . . . .	22
<b>5. Konzeption und Umsetzung</b>	<b>26</b>
5.1. Setup der TYPO3-Erweiterung . . . . .	26
5.2. Digitization . . . . .	27
5.2.1. Teilnehmerregistrierung . . . . .	29
5.2.2. Weinregistrierung . . . . .	34
5.2.3. PDF- und QR-Code-Generierung . . . . .	40
5.2.4. Jahresauswahlproben- und Wein-Detailansichten .	42
5.3. Digitalization . . . . .	43
5.3.1. Schnittstelle QR-Code-Scanner . . . . .	43
5.3.2. CSV-Export . . . . .	44
<b>6. Ergebnisse</b>	<b>46</b>
6.1. Praktische Ergebnisse . . . . .	46
6.2. Forschungsergebnisse . . . . .	46
6.3. Diskussion . . . . .	49
<b>7. Fazit</b>	<b>53</b>
7.1. Ausblick . . . . .	54
<b>Literaturverzeichnis</b>	<b>56</b>
<b>Anhang</b>	<b>63</b>
<b>A. Anmeldeforumular Jahresauswahlprobe</b>	<b>64</b>
<b>B. Interview-Fragebogen</b>	<b>67</b>
<b>C. Online-Fragebogen Weingüter</b>	<b>69</b>
<b>D. Interview-Protokoll</b>	<b>72</b>
<b>E. Pflichtenheft</b>	<b>75</b>
<b>F. Klassendiagramm</b>	<b>77</b>

---

<b>G. Notizen: Digitization des Weinanmelde-Formulares</b>	<b>79</b>
--	-----------

# Abbildungsverzeichnis

1.	Geschäftsprozess Jahresauswahlprobe: vor der Digitalisierung, Quelle: Eigene Darstellung . . . . .	2
2.	Modell um digitale Transformation in Angriff zu nehmen, Quelle: [Parviainen et al., 2022] . . . . .	5
3.	Das Plan-Do-Check-Act-Modell, Quelle: [Abraham, 2005]	6
4.	Umsetzungsplanung, Quelle: Eigene Darstellung . . . . .	9
5.	Geschäftsprozess Jahresauswahlprobe: nach der Digitalisierung, Quelle: Eigene Darstellung . . . . .	25
6.	Objektrelationen: Weinland Mosel Jahresauswahlprobenwerkzeug, Quelle: Eigene Darstellung . . . . .	28
7.	UX-Flow: Registrierung, Quelle: Eigene Darstellung . . .	31
8.	UX-Flow: Weinanmeldung, Quelle: Eigene Darstellung .	35
9.	Stichprobenartige Laufzeitanalyse des Kategorie-Renderers, gegenüber einer quadratischen Kurve, Quelle: Eigene Darstellung . . . . .	38
10.	Geschäftsprozess Jahresauswahlprobe: Ausblick, Quelle: Eigene Darstellung . . . . .	54

# Tabellenverzeichnis

1.	Subjektive Evaluation zur Eignung als QR-Code-Bibliothek: kjua . . . . .	15
2.	Subjektive Evaluation zur Eignung als QR-Code-Bibliothek: soldair/node-qrcode . . . . .	16
3.	Subjektive Evaluation zur Eignung als QR-Code-Bibliothek: chillerlan/php-qrcode . . . . .	16
4.	Subjektive Evaluation zur Eignung als QR-Code-Bibliothek: Kreativekorp/barcode . . . . .	17
5.	Subjektive Evaluation zur Eignung als QR-Code-Bibliothek: Bacon/BaconQrCode . . . . .	18
6.	Gesamtübersicht: Subjektive Evaluationen zur Eignung von QR-Code Bibliotheken . . . . .	18
7.	Initial bekannte Anforderungen . . . . .	20

# Abkürzungsverzeichnis

**WM** “Weinland Mosel” e.V.

**PDCA** Plan-Do-Check-Act

**SWOT** Strength-Weakness-Opportunity-Threat

**FA** funktionale Anforderung

**NFA** nichtfunktionale Anforderung

**ELN** Einlieferungsnummer

**JAP** Jahresauswahlprobe

**TCA** Table configuration array

**PO** Product Owner

**UID** Unique Identifier

**PID** Parent Identifier



# Glossar

## Frontend

*“Front-end website is a combination of hypertext markup language (HTML), cascading style sheets (CSS) and JavaScript that generates pages that can be viewed and interacted directly with the user.”* [Udjaja, 2018, S. 294].

## TYPO3-Backend

*“The backend’s main role is to enable users to create and publish content for their site. The backend is also used to configure a TYPO3 installation. Domains, languages and other information that determine how a site behaves are managed via the backend. Tasks such as adding backend users and managing third-party extensions also take place in the backend. ”* [TYPO3 Contributors, 2023l].

## TYPO3-Extension

*“Developed by the community, extensions provide a range of solutions that help extend TYPO3”* [TYPO3 Contributors, 2023l].

## TYPO3-Plugin

*“Plugins are a specific type of content elements [...] Plugins often use additional database tables which contain records which are dynamically displayed via the plugin - often in a list view, a single view, optionally with pagination and search functionality. An extension may provide several plugins, each with a dedicated function, such as the list view.”* [TYPO3 Contributors, 2023f]

## Weinland-Mosel-Mitglied

Ein Weinland-Mosel-Mitglied ist ein Mitglied des Moselwein e.V. Diese Mitglieder sind i.d.R. Weingüter und weinproduzierende Unternehmen. Mitglieder profitieren von einer Reihe an Vorteilen, wie beispielsweise einer Auflistung auf Weinland Mosels Internetauftritt. Durch diesen Auftritt sind bereits Stammdatensätze vorhan-

den. Durch eine Teilnahme an einer Jahresauswahlprobe ist ein Unternehmen kein Mitglied.

# 1. Einleitung

Der Regionalverband für Weine “Weinland Mosel” e.V. (WM) lässt Weine in organisierten Weinproben, sog. Jahresauswahlproben, von Juroren bewerten. Teilnehmende Weingüter registrieren ihre Weine in verschiedenen Kategorien und schicken diese auf dem Postweg ein. Dieser Prozess bildet sich in Form ausgedruckter Formulare ab, die von Hand ausgefüllt und von Hand in eine Excel-Tabelle übertragen werden.

## 1.1. Problemstellung

Die teilnehmenden Weingüter schicken ihre Weine zusammen mit Formularen über den Postweg zu WM. Es ist der Normalfall, dass ein teilnehmendes Weingut *mehrere* Weine zur Bewertung anbringt. In diesem Fall ist für jeden anzumeldenden Wein ein solches Formular erneut auszufüllen. Hierbei werden sämtliche auf das Weingut bezogene Daten redundant ausgefüllt. Das ist so, da diese Daten zwischen den Weinen gleichbleibend sind. Abgesehen davon, dass solche Redundanzen auf Weinguts- und Verbandsseite die hedonische Qualität schädigen, bietet ein solcher Workflow Freiraum für Fehler und Inkonsistenzen. Dieser Workflow, mit den zuvor genannten Nachteilen, wird auf Verbandsseite, nach Zustellung der Weine, weiter fortgeführt: WM erfährt erstmalig mit der Zustellung eines Weines von dessen Teilnahme. Das erschwert die Planung der Logistik, da im Voraus keine konkrete Anzahl der zu erwartenden Flaschen bekannt ist. Geht eine Flasche auf dem Postweg verloren, könnte das unbemerkt bleiben, da der Prozess für das Weingut mit dem Versand endet und der Prozess für WM erst mit dem Erhalt des der Flasche beiliegendem Formulars beginnt. Der Postweg stellt somit eine Diskontinuität zwischen diesen Prozessen dar. Kommt ein teilnehmender Wein bei WM an, wird das beiliegende Formular von Hand in eine Excel-Tabelle übertragen. Diese Schnittstelle ist besonders ressourcenaufwändig und fehleranfällig, da es oft vorkommt, dass die teils dys-

graphisch verfassten Formulare nur schwer, mehrdeutig, oder gar nicht dechiffriert werden können. In diesem Prozess wird der Teilnahme des Weines eine inkrementell aufsteigende Einlieferungsnummer (ELN) zugewiesen, die anschließend in Form eines Aufklebers auf der Flasche angebracht wird. Darauffolgend wird die Flasche im Lager verstaut.

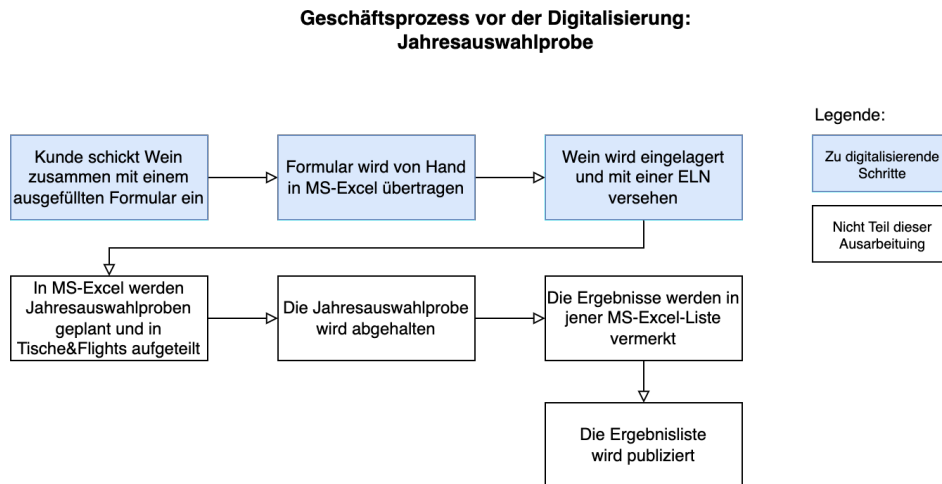


Abbildung 1.: Geschäftsprozess Jahresauswahlprobe:  
vor der Digitalisierung, Quelle: Eigene Darstellung

## 1.2. Zielsetzung

Ziel dieser Arbeit ist es, in Erfahrung zu bringen, wie der zuvor genannte Prozess bestmöglichst, im Rahmen bestimmter Randbedingungen und funktionalen- sowie nichtfunktionalen Anforderungen, digitalisiert werden kann. Während die Randbedingungen bereits bekannt sind, werden detaillierte Anforderungen im Rahmen der Anforderungstechnik ausgearbeitet [Heinemann, 2021]. Die Randbedingungen besagen, dass der Anmeldeprozess in die existierende Internetpräsenz des Weinverbandes integriert werden muss. Bei dieser Internetpräsenz handelt es sich um ein TYPO3-Redaktionssystem. Sämtliche Interaktionen zwischen Akteuren, die nicht WM oder dem Softwaresystem zugehörig sind, müssen im Frontend der Webseite stattfinden, da das Backend den Redakteuren

und Administratoren vorbehalten ist. Oberflächen für Mitarbeiter von WM dürfen in der TYPO3-Backend-Oberfläche implementiert werden.

Somit lautet die **Forschungsfrage**:

*“Wie kann die Anmeldung und Zustellung von Weinen für Weinproben des Regionalverbunds für Weine in der Weinregion Mosel effizient und profitabel durch eine TYPO3-Erweiterung realisiert werden?”*

## 2. Stand der Forschung

Der Stand der Forschung beleuchtet verschiedene Entwicklungsmethodiken zur Digitalisierung.

### 2.1. Modell nach Parviainen et al.

Parviainen et al. stellten sich die Frage, wie man in der Praxis von Digitalisierung profitieren kann und entwickelten in ihrer Forschungsarbeit einen konzeptionellen Rahmen, um zu verstehen, wie die Digitalisierung in der Praxis umgesetzt werden kann und welche Vorteile sich daraus ergeben. Dieser Rahmen basiert auf dem Plan-Do-Check-Act (PDCA) -Prinzip.

Dieses Rahmenwerk sieht anhand des PDCA-Prinzips vier Schritte vor:

**Im ersten Schritt** wird definiert, wie weit die Digitalisierung für das Unternehmen gehen kann und welche Position das Unternehmen dabei anstrebt. Dieser Schritt kann in vier Teilschritte unterteilt werden: Ausmaße, Treiber, Szenarien und Ziele. Für die Bestimmung der Ausmaße ist die Analyse aktueller Trends und deren Relevanz für die Domäne des Unternehmens wichtig. Ebenfalls ist wichtig, wie weit diese Trends bereits im Fachgebiet verankert sind. Zur Einordnung eignen sich Strength-Weakness-Opportunity-Threat (SWOT) -Analysen. Diese Analysen sind die Grundlage, um ein Unternehmen in der Digitalisierung aufzustellen.

Aus den Ergebnissen der Trendanalysen sollten dann Treiber identifiziert werden. Diese Treiber sollten auf der Grundlage zukünftiger Ergebnisse skalierbar sein: Beispielsweise könnten drastische Maßnahmen erforderlich sein, um schlimme Auswirkungen zu verhindern oder große Verbesserungen zu erreichen.

Für die relevantesten Treiber sollten Zukunftsszenarien untersucht werden. Dies ist wichtig, um zu wissen, welche Auswirkungen bestimmte Trends in welcher Ausprägung haben würden. Relevant sind hier die Vorteile der Umsetzung des Szenarios, die Kosten der Umsetzung, sowie die Risiken. Auf dieser Basis kann das beste Szenario ausgewählt werden.

Aus diesem Szenario werden schließlich die Ziele der Digitalisierung abgeleitet. Diese Ziele müssen so formuliert sein, dass sie mit der Ausgangssituation verglichen werden können.

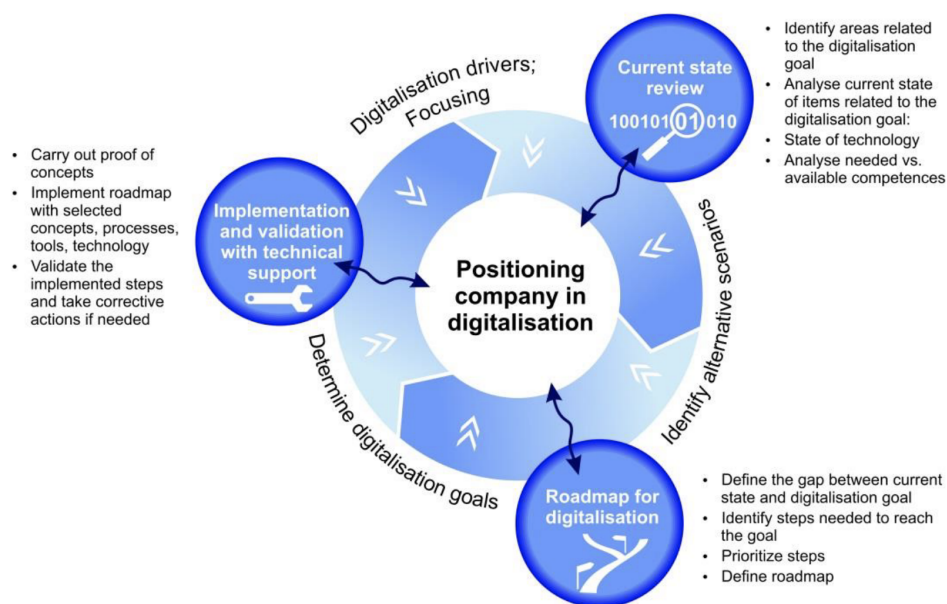


Abbildung 2.: Modell um digitale Transformation in Angriff zu nehmen,  
Quelle: [Parviainen et al., 2022]

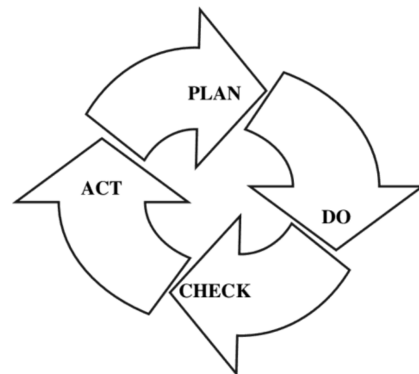


Abbildung 3.: Das Plan-Do-Check-Act-Modell, Quelle: [Abraham, 2005]

**Im zweiten Schritt** wird der Ist-Zustand des Unternehmens ermittelt. Dazu wird die aktuelle Positionierung des Unternehmens im Hinblick auf den Zielzustand mit Fokus auf die Digitalisierungsziele betrachtet. Dazu wird der Ist-Zustand im Kontext des Soll-Zustandes anhand definierter Fragen bewertet. Die Auswahl der Fragen unterscheidet sich je nach Art der Ziele. Der gesamte Fragenkatalog kann im Detail der Ausarbeitung von Parviainen et al. entnommen werden.

**Im dritten Schritt** werden konkrete Schritte festgelegt, die für den Übergang vom Ist-Zustand in den Soll-Zustand erforderlich sind. Dazu muss zunächst die Lücke zwischen dem Ist- und dem Soll-Zustand identifiziert werden. Relevant ist dabei der aktuelle Stand der Technik und welche Veränderungen notwendig sind, um den Zielzustand zu erreichen. Anschließend sollten die konkreten Schritte identifiziert werden, die erforderlich sind, um diese Lücke zu schließen. Wenn zum Beispiel ein Treiber “interne Effizienz” ist, könnten die Schritte darin bestehen, neue digitale Werkzeuge zu integrieren. Schließlich werden diese Schritte analysiert und priorisiert. Prädestiniert dafür sind Kosten-Nutzen-Analysen, Analysen der Umsetzbarkeit, des Wartungsaufwands und der Mitarbeiterschulung.



**Der vierte Schritt** befasst sich mit der Umsetzung der in Schritt drei geplanten Maßnahmen und der Bewertung der erzielten Ergebnisse. Diese Bewertung der Ergebnisse sollte z.B. soziokulturelle Barrieren berücksichtigen, die sich aus den Reaktionen bestimmter Stakeholder ergeben, die möglicherweise negativ auf Veränderungen reagieren oder Schwierigkeiten bei der Einführung neuer Technologien haben. Wenn diese Analyse zeigt, dass die Ziele der Digitalisierung nicht erreicht wurden, sollten Korrekturmaßnahmen in früheren Phasen ergriffen werden [Parviainen et al., 2022].

### 2.2. Modell nach Verhoef et al.

Nach Verhoef et al. lässt sich der hier sogenannte “Prozess der Digitalisierung” in drei Phasen unterteilen. Diese drei Phasen sind *Digitization*, *Digitalization* und *Digital Transformation* [Verhoef et al., 2021]. Die Phase *Digitization* befasst sich mit der Umwandlung analoger Datenstrukturen und Modelle in digitale Datenmodelle, sodass diese digital, in Form von Nullen und Einsen, gespeichert und elektronisch weiterverarbeitet werden können [Dougherty und Dunne, 2012] [Loebbecke und Picot, 2015]. “*Examples concern the use of digital forms in ordering processes, the use of digital surveys, or the use digital applications for internal financial declarations.*” [Verhoef et al., 2021, S. 891]. *Digitalization* beschreibt den Prozess der Veränderung bestehender Geschäftsprozesse, um mit digitalen Werkzeugen und Datenmodellen zu arbeiten [Li et al., 2016]. Beispielsweise der Verwendung von Robotern in der Produktion [Verhoef et al., 2021]. Die letzte Phase, die *Digitale Transformation*, beschreibt eine firmenweite Veränderung, die beispielsweise Ergründungen neuer Geschäftsmodelle mit sich bringen könnte [Pagani und Pardo, 2017].

## 2.3. Abwägung in Bezug auf die Problemstellung

In Bezug auf die hier betrachteten Methodiken ist es wichtig zu erwähnen, dass der betrachtete Kontext lediglich die Digitalisierung **eines** Geschäftsprozesses behandelt. Diese Ausarbeitung befasst sich nicht mit firmenweiten Veränderungen, wie sie von den nahegelegten Modellen abgedeckt ist. Daher sind geringfügige Anpassungen der Methodiken unabdinglich. Des Weiteren ist Ressourcenintensität ein relevanter Gesichtspunkt dieser Abwägung, da eine solche Digitalisierung effizient und profitabel sein soll.

### 2.3.1. Parviainen et al.

Der Autor stellt fest, dass das Modell nach Parviainen et al. überwiegend methodisch ausgerichtet ist und für den betrachteten Kontext der Problemstellung unverhältnismäßig ressourcenintensiv ist, da der Umfang der Projektes geringfügig ist. Dieses Modell legt den Fokus auf firmenweite Veränderungen anstatt auf einzelne Prozesse und könnte daher in Bezug auf diese Problemstellung als “überdimensioniert” bezeichnet werden.

### 2.3.2. Verhoef et al.

Im Modell nach Verhoef et al. stellt die hier behandelte Problemstellung lediglich Phase eins und Phase zwei der drei Phasen *Digitization*, *Digitalization* und *Digital Transformation* dar. Das ist so, da sich die ersten zwei Phasen mit der Digitalisierung bestimmter Geschäftsprozesse befassen. Phase drei befasst sich mit firmenweiten Veränderungen, die über Geschäftsprozesse hinausgehen [Verhoef et al., 2021]. Firmenweite Veränderungen befinden sich außerhalb des Rahmens dieser Ausarbeitung. Der Autor empfindet das Modell nach Verhoef et al. als agiler, unrestrictiver, weniger overheadlastig und somit effizienter.

### 2.3.3. Fazit

Aufgrund des kleinen Projektumfanges und der großen Wahrscheinlichkeit, dass sich Anforderungen ändern werden, fällt die Entscheidung auf das minimalistischere Modell nach Verhoef et al.. Grundlage dieser Entscheidung sind Forschungsarbeiten von M. Ayman Al Ahmar, die nahelegen, dass Projekte dieser Art idealerweise nach “extreme programming”-Paradigmen behandelt werden [Ahmar, 2005]. Nachdem Phase drei des Verhoef-Modells ausgeklammert wurde, sieht das zu verfolgende Modell aus wie folgt:

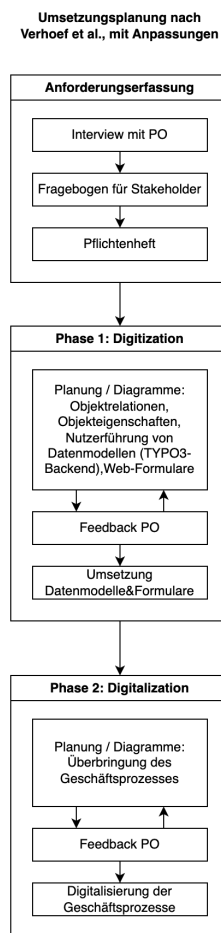


Abbildung 4.: Umsetzungsplanung, Quelle: Eigene Darstellung

## 3. Stand der Technik

Der Stand der Technik bezieht sich auf bestehende, praktische Umsetzungen der erforderlichen Technologien. Im Wesentlichen gibt es drei Arten von Technologien, die untersucht werden müssen: Die bestehende Webseite von WM, Bibliotheken zur Erzeugung von QR-Codes und Bibliotheken zur Erzeugung von PDF-Dateien.

### 3.1. Die bestehende Webseite

Als Mitentwickler des Projektes ist dem Autor bekannt, dass die bestehende Webseite ein TYPO3-Redaktionssystem ist. Das Frontend der Webseite wird mit Webpack und Sass übersetzt. Webpack ist ein Modulbundler [Zimmermann, 2017] und Sass ein CSS-Präprozessor [W3Schools, 2023b]. TYPO3 ist ein Redaktionssystem und PHP-Rahmenwerk, das Daten- und Inhaltspflege in einem geschützten Bereich ermöglicht. Außerdem steuert TYPO3 Frontend-, Backend-Nutzer und deren Berechtigungen [TYPO3 Contributors, 2023l]. Über die Systemerweiterung “Extbase” wird eine mächtige Entwicklerschnittstelle bereitgestellt, um hochindividualisierte Funktionalitäten zu ermöglichen. [TYPO3 Contributors, 2023c]

### 3.2. QR-Code-Bibliotheken

Um mit QR-Codes zu arbeiten, ist es unabdinglich, QR-Codes zu erstellen. Im Folgenden werden einige Implementationen von QR-Code-Generator-Bibliotheken im Detail betrachtet. Es wird sich auf bereits vom System verwendete Programmiersprachen begrenzt.

### 3.2.1. JavaScript-Implementationen

#### **jquery-qrcode.js**

“jquery-qrcode.js” ist ein Plugin für jQuery, um dynamisch QR-Codes auf Browserseite zu generieren. Jedoch verweist diese Bibliothek selbst auf ihren desolaten Zustand und empfiehlt stattdessen “kjua” zu verwenden [Jung, 2020a]. Somit scheidet “jquery-qrcode.js” für nähere Evaluationen aus, da die Software nicht mehr gepflegt wird. “jquery-qrcode.js” wurde bis Januar 2023 853 mal von Nutzern mit einem Stern markiert [Jung, 2020a]. Wenn ein Nutzer ein Github-Repository mit einem Stern markiert, zeigt sich der Nutzer interessiert genug, um über Neuigkeiten auf dem Laufenden gehalten werden zu wollen [Github, 2023]. “jquery-qrcode.js” ist MIT-lizenziert [Jung, 2020a].

#### **kjua**

“kjua” ist eine JavaScript-Bibliothek, die dynamisch QR-Codes auf Browsersebene generiert. Im Gegensatz zu “jquery-qrcode.js” funktioniert *kjua* auch ohne jQuery. Es werden diverse Stilattribute für gestaltete QR-Codes unterstützt [Jung, 2020b]. “kjua” kann QR-Codes über HTML-Canvas, Bilder und Vektorgrafiken umsetzen. Das ist bei näherer Betrachtung der Kjua-Tech-Demo (<https://larsjung.de/kjua/latest/demo>) ersichtlich, jedoch nicht explizit erwähnt. Nach dem aktuellen Stand (Januar 2023) wurde “kjua” von 243 Nutzern mit einem Stern markiert. “Kjuas” letzte Codeänderung liegt circa zwei Jahre zurück. Es gibt eine unbeantwortete, sieben Jahre alte Pull-Request und Issues scheinen ignoriert zu werden. “kjua” ist MIT-lizenziert [Jung, 2020b].

#### **soldair/node-qrcode**

“soldair/node-qrcode” ist eine Node.js-basierte Implementation eines QR-Code-Generators und bietet somit Funktionalität serverseitig, als Kommandozeilenwerkzeug, sowohl auch browserseitig an. Die Readme-Datei ist umfangreich, reich an Beispielen und detailreichen Erklärungen. Der letzte Commit ist zu diesem Zeitpunkt knapp älter als ein hal-

bes Jahr. Somit macht das Projekt einen moderat gepflegten Eindruck. Die Readme-Datei verweist auf Unit-Tests bei Travis, jedoch lief die letzte Pipeline Februar 2021 durch und schlug fehl. Einige Pull-Requests und Issues werden seit Jahren ignoriert [Day, 2020]. Die Bibliothek wurde 74 Millionen mal heruntergeladen und mit 6.308 Sternen markiert. “Soldair/node-qrcode” ist MIT-lizenziert [npm, inc, 2020].

#### 3.2.2. PHP-Implementationen

##### **chillerlan/php-qrcode**

Eine arrivierte Lösung für QR-Code-Generierung in PHP ist “chillerlan/php-qrcode”, umgesetzt durch Eugen Rochko et al.. “chillerlan/php-qrcode”s Etabliertheit zeigt sich durch den zu diesem Zeitpunkt (Januar 2023) auf 5 Millionen stehenden Downloadzähler, mehreren Entwicklern und dem letzten Commit, der zu diesem Zeitpunkt nicht älter als einen Monat ist. Insgesamt erfolgten bis dato 808 Commits von 6 Entwicklern. Das Projekt verfügt über Unit-Tests, die 90% der Zeilen der Codebase abdecken. Rochko übernahm Teile der Codebase aus dem Java-Projekt “ZXing” und übersetzte diese zu PHP. Es gibt zu diesem Zeitpunkt keine unbeantworteten Issues oder Pull-Requests. “chillerlan/php-qrcode” basiert auf einer angepassten Version von “kazuhikoarase/qrcode-generator”. Einzig auffällig sind die Commitnachrichten, die zuteils nur aus einem Emoji bestehen. “chillerlan/php-qrcode” wurde von 1.212 Github-Nutzern mit einem Stern markiert. Die Bibliothek ist MIT-lizenziert [Rochko, 2022b].

##### **kreativekorp/barcode**

“kreativekorp/barcode” ist eine PHP-Bibliothek zur Generierung von QR-Codes, bereitgestellt von “Kreative Software”, R.G. Bettencourt. Diese Implementation umfasst etablierte Barcode-Formate und unterstützt eine Vielzahl an Anpassungsmöglichkeiten. Das Projekt wurde bis zum heutigen Tag 189 mal mit Sternen markiert [Bettencourt, 2018]. Vier von den sechs vorhandenen Commits erfolgten innerhalb von zwei

Tagen in 2016. Ein weiterer Commit erfolgte wenige Tage später. Der aktuellste Commit wurde knapp zwei Jahre später, 2018, veröffentlicht. Das liegt zu diesem Zeitpunkt vier Jahre in der Vergangenheit. Damit ist diese Bibliothek de facto sechseinhalb Jahre alt und wurde seitdem einmal um Features erweitert. Issues und Pull-Requests werden weitestgehend ignoriert. Die Bibliothek verwendet die MIT-Lizenz.

[Bettencourt, 2018].

#### **Bacon/BaconQRCode**

“Bacon/BaconQRCode” ist eine PHP-Bibliothek zur Generierung von QR-Codes, bereitgestellt von Ben Scholzen, hinter der Github-Organisation “Bacon”, dessen einziges Mitglied Scholzen darstellt. Verlinkt ist eine Homepage, die zu einer Nginx-“Hello World”-Seite führt. Begleitet wird “BaconQRCode” von etlichen weiteren “Bacon-Projekten” wie Beispielsweise “BaconPdf”, “BaconStringUtils” und “BaconUser” um nur einige zu nennen. Insgesamt machen die stichprobenartig betrachteten Projekte einen desolaten Eindruck mit zuteils aktuellsten Commits von vor zehn Jahren. “BaconQRCode” stellt das beliebteste und gepflegteste Projekt von Scholzen mit 1.508 Sterne-Markierungen und einem aktuellsten Commit von vor zwei Monaten dar. “BaconQRCode” fällt mit siebzehn Entwicklern auf, die jeweils zumindest einen Commit beigetragen haben. Zu diesem Zeitpunkt fanden 177 Commits statt. Githubs DependencyGraph verzeichnet nahezu 80.000 Projekte, die BaconQRCode verwenden [Scholzen, 2018] und Packagist meldet 50 Millionen Downloads [Packagist, 2023]. Wie “chillerlan/php-qrcode” baut auch Scholzen auf existierende Technik von “ZXing” auf. Eine Dokumentation neben der Readme-Datei existiert nicht und diese ist sehr minimalistisch. “BaconQRCode” kann QR-Codes als Rasterbilder und Vektorgrafiken (SVG und EPS) generieren. Spezielle Styles sind nicht erwähnt. Ein Großteil der Issues und Pull-Requests wurden behandelt. “BaconQRCode” unterliegt einer BSD-2-Clause-Lizenz [Scholzen, 2018].

### 3.2.3. Subjektiver Vergleich im Bezug auf die Problemstellung

Im Folgenden werden subjektive Einschätzungen und Meinungen des Autors über die Eignung der beleuchteten Bibliotheken vorgestellt. Um eine Bibliothek als “am geeignetsten” für einen Anwendungsfall unter den betrachteten Bibliotheken zu kurieren, müssen die konkreten Anforderungen für diesen Anwendungsfall beachtet werden. Das ist so, da verschiedene Eigenschaften der Bibliotheken verschiedene Auswirkungen in Gewichtung und Richtung je nach Anwendungsfall aufweisen. Hierfür werden die zuvor vorgestellten Bibliotheken zur Erstellung von QR-Codes in den folgenden Attributen gegenübergestellt:

#### **Funktionalität**

Der Umfang, der unterstützten Funktionen, in Annahme dessen, dass die Bibliothek syntaktisch und semantisch korrekt ist.

#### **Gepflegtheit**

Das Ausmaß, in dem das Projekt aktiv gepflegt wird und ordnungsgemäß entwickelt zu sein scheint. Hierzu zählen beispielsweise: Bearbeitung von Issues, Bearbeitung von Pull-Requests, Präsenz von Tests, Präsenz einer angemessenen Dokumentation, häufige Commits, mehrere Entwickler, Anzahl der Sterne auf Github, sowie der Anzahl der Downloads bzw. Installationen, falls verfügbar.

#### **Workflow-Eignung**

Die Eignung einer Bibliothek in existierende Workflows und Randbedingungen übernommen zu werden. Maßgeblich ist, ob und mit wie viel Aufwand eine Bibliothek in das Projekt übernommen werden kann. Beispielsweise ist es deutlich aufwändiger eine JavaScript-Bibliothek in einem PHP-Projekt zu verwenden, als eine native PHP-Bibliothek. Ebenfalls ist relevant, ob die Lizenz einer Bibliothek eine Verwendung gestattet, bzw. welche Bedingungen gelten.

Hierbei werden den verschiedenen Bibliotheken Punkte  $[0, 10]$  in den drei zuvor genannten Kategorien vergeben. Nicht ermittelte Werte wer-



den in ihrer Kategorie durch “–” repräsentiert, wodurch eine weitere Verwendung ausgeschlossen wird. Die Kumulativpunktzahl  $[0, 30]$  einer Bibliothek beschreibt deren Gesamteignung, nach subjektivem Empfinden des Autors.

#### **kjua**

“kjua” ist funktional für dieses Projekt gut aufgestellt, da es optisch ansprechende QR-Codes mit Logo unterstützt. Das wird mit acht Punkten in “Funktionalität” bemessen. “kjua” macht andererseits einen eher desolaten Eindruck und lässt somit begründete Zweifel an dessen Zukunftssicherheit zu. Eine knappe Dokumentation ist verfügbar. Für die Dokumentation werden zwei Punkte in “Gepflegtheit” angerechnet. “kjua” ist als JavaScript-Bibliothek nur schwer mit den Anforderungen vereinbar, da der QR-Code in einem PDF eingebunden werden soll. Hierfür ist ein serverseitiger PDF-Generator vorgesehen. “kjuas” Lizenz erlaubt Verwendung in kommerziellen, Closed-Source-Projekten [Opensource.org, b]. Die Exklusivität für Nutzung in Webbrowsern schließt eine Einbindung in den Workflow aus.

Funktionalität	Gepflegtheit	Workflow-Eignung	$\Sigma$
8	2	0	10

Tabelle 1.: Subjektive Evaluation zur Eignung als QR-Code-Bibliothek: kjua

#### **soldair/node-qrcode**

Soldairs Lösung sieht dokumentativ und funktional vielversprechend aus. Hierdurch bilden sich acht Punkte in “Funktionalität” ab. Da diese Bibliothek eine Node.js-Bibliothek ist und Node.js ohne Browser, als Kommandozeilenapplikation, ausgeführt werden kann [OpenJS Foundation, 2023], bietet sich “soldair/node-qrcode” als serverseitige Anwendung an. Die Lizenz spricht nicht gegen eine Verwendung. Dennoch wäre es ein vermeidbarer Mehraufwand generierte QR-

Codes aus einer Node.js-Applikation in eine PHP-Umgebung zu übertragen. Das bildet sich mit vier Punkten in “Workflow-Eignung” ab. Darüberhinaus macht “soldair/node-qrcode” einen verbesserungswürdigen Eindruck der Projektpflege, wofür es lediglich drei Punkte gibt.

Funktionalität	Gepflegtheit	Workflow-Eignung	$\Sigma$
8	3	4	15

Tabelle 2.: Subjektive Evaluation zur Eignung als QR-Code-Bibliothek:  
soldair/node-qrcode

### chillerlan/php-qrcode

Rochkos Lösung macht einen aktiv gepflegten Eindruck und wird von großen Downloadzahlen gestützt. Rochko stellt eine ausführliche Dokumentation und Codebeispiele bereit. Hierfür wird die volle Punktzahl in “Gepflegtheit” vergeben. Der Funktionsumfang übersteigt weit die Anforderungen und gestattet so ausführlich gestaltete QR-Codes zu generieren [Rochko, 2023], die somit die hedonische Qualität fördern [König, 2022]. Bewertet wird das mit einer vollen Punktzahl in “Funktionalität”. “chillerlan/php-qrcode” lässt sich nahtlos in das existierende Projekt eingliedern, da es als PHP-Bibliothek über Composer eingebunden werden kann und eine eigene API bereitstellt [Rochko, 2022b]. Die von Rochko verwendete Lizenz gestattet eine unkomplizierte Verwendung. “chillerlan/php-qrcode” funktioniert sowohl mit PHP- 7.x als auch 8.x. Die Bibliothek benötigt zwei weitere Abhängigkeiten. Eine dieser Abhängigkeiten ist ebenfalls von Rochko bereitgestellt [Rochko, 2022c] und weist eine ähnlich gute Projektpflege auf [Rochko, 2022a]. Das wird mit der Maximalwertung in “Workflow-Eignung” berechnet.

Funktionalität	Gepflegtheit	Workflow-Eignung	$\Sigma$
10	10	10	30

Tabelle 3.: Subjektive Evaluation zur Eignung als QR-Code-Bibliothek:  
chillerlan/php-qrcode

### **kreativekorp/barcode**

Kreativekorp beeindruckt in der Readme-Datei durch Nutzungsbeispiele und Dokumentation, sowie einer Vielzahl unterstützter Barcode-Formate, darunter auch QR-Codes und einiger improvisierter Tests. In Anbetracht dessen, dass die Bibliothek de facto sechseinhalb Jahre alt ist und seit vier Jahren nicht mehr angepasst wurde, wird eine geringe Wertung von drei Punkten in “Gepflegtheit” vergeben. Null Punkte in “Workflow-Eignung” rechtfertigen sich durch die Abwesenheit jeglicher Unterstützung für Paketmanager, wodurch eine saubere Verwendung in dem Brown-Field-Projekt Änderungen an der Bibliothek an sich erfordern würde. Die Funktionalität wurde aufgrund der desaströsen Gepflegtheit und Eignung nicht näher untersucht, da eine Verwendung selbst mit guter Funktionalität nicht infrage käme.

<b>Funktionalität</b>	<b>Gepflegtheit</b>	<b>Workflow-Eignung</b>	$\Sigma$
-	3	0	-

Tabelle 4.: Subjektive Evaluation zur Eignung als QR-Code-Bibliothek: Kreativekorp/barcode

### **Bacon/BaconQrCode**

“BaconQrCode” nennt keine speziellen Optionen, um näheren Einfluss auf den generierten QR-Code auszuüben. Es werden Rasterbilder und verschiedene Vektorformate als Renderziel unterstützt. Aufgrund der fehlenden Optionen, die andere Bibliotheken bereitstellen, werden lediglich fünf von zehn Punkten in “Funktionalität” vergeben. Die Projektgepflegtheit ist inkonsistent. Manche Stellen, beispielsweise die Mitwirkenden, Alter des neuesten Commits, Tests und Nutzerzahlen machen einen guten Eindruck, während Punkte wie die Dokumentation, die Organisationswebseite und andere Projekte von Scholzen Sorgen begründen. Weil der Programmcode an sich gut gepflegt aussieht und große Downloadzahlen von häufiger Verwendung sprechen, werden für die “Gepflegtheit” acht Punkte vergeben. Da es sich hierbei um eine PHP-Bibliothek handelt, die über Composer in PHP- 7.x und 8.x Projekte geladen werden

kann und das Projekt eine API bereitstellt, ist die technische Workflow-Eignung gut. Die BSD-2-Clause-Lizenz verkompliziert eine Integration, da somit eine Copyright-Notiz an Nutzer gezeigt werden müsste [Opensource.org, a]. Dadurch werden drei Punkte einer vollkommenen Workflow-Eignung abgezogen, wodurch sieben Punkte vergeben werden.

Funktionalität	Gepflegtheit	Workflow-Eignung	$\Sigma$
5	8	7	20

Tabelle 5.: Subjektive Evaluation zur Eignung als QR-Code-Bibliothek:  
Bacon/BaconQRcode

### 3.2.4. Fazit

Nach Evaluation der verschiedenen QR-Code-Bibliotheken im Kontext der vorliegenden Problemstellung erweist sich aus Sicht des Autors “chillerlan/php-qrcode” mit 30 Gesamtpunkten als die am besten geeignete Bibliothek unter den Betrachteten. Somit wird “chillerlan/php-qrcode” als QR-Code Technologie in der Lösung dieser Problemstellung verwendet.

Bibliothek	Funkt.	Gepflegt.	WF.-Eignung	$\Sigma$
chillerlan/php-qrcode	10	10	10	30
Bacon/BaconQRcode	5	8	7	20
soldair/node-qrcode	8	3	4	15
kjua	8	2	0	10
kreativekorp/barcode	-	3	0	-

Tabelle 6.: Gesamtübersicht: Subjektive Evaluationen zur Eignung von  
QR-Code Bibliotheken

### 3.3. PDF-Generator

Firmenintern ist der PDF-Generator “mpdf/mpdf” etabliert und wird bereits in zahlreichen Projekten verwendet. Um die projektübergreifende Kontinuität der Technik zu wahren und um somit die Wartbarkeit und Nachhaltigkeit des hier behandelten Softwareproduktes zu fördern, wird sich für den PDF-Generator entschieden, der bereits firmeninterner Standard ist. Unabhängig dessen ist “mpdf” ein gut gepflegtes Projekt mit einem Alter von mehr als acht Jahren, Sponsoren, 72 Entwicklern, über 31 Millionen Downloads, über 3.900 Sterne-Markierungen, über 800 Commits und regelmäßigen Updates. “Mpdf” ein Composer-Paket und für verschiedene PHP-Versionen verfügbar. Mpdf verwendet eine GPL-2.0 Lizenz [Mpdf Contributors, 2023a].

## 4. Anforderungserfassung

Obwohl bereits ein grober Anriss des Zielsystems bekannt ist, ist es unabdinglich eine Anforderungsanalyse durchzuführen, um Details auszuarbeiten [Ebert, 2019]. Hierbei ist es wichtig, kein exzessives Pflichtenheft aufzubauen, denn letztendlich zählt nur, was dem Kunden geliefert wird. Nicht, wie viele gar nicht benötigte Anforderungen umgesetzt wurden. *“Zu viele oder falsche Anforderungen ruinieren Budgets, Termine und die Qualität.”* [Ebert, 2019, S. viii]. Die Anforderungen eines Produktes sind in drei Kategorien einzuteilen: funktionale Anforderungen (FAs), nichtfunktionale Anforderungen (NFAs) und Randbedingungen [Heinemann, 2021]. Wie oben erwähnt, sind bereits die Randbedingungen und einige funktionale und nichtfunktionale Anforderungen bekannt. Diese sind:

Art der Anforderung	Anforderungsbeschreibung
NFA	Angaben zum Weingut des Weines sollen aus dem Accountdatensatz anstatt aus dem Web-Formular kommen.
NFA	Aus dem Papierformular soll ein Web-Formular werden.
FA	Beim Erstellen einer Weiteilnahme soll ein QR-Code als PDF generiert werden, der den Wein identifiziert.
FA	Durch scannen des QR-Codes auf dem mit einer Weiteilnahme erstellten PDFs soll dem Wein der Status “eingegangen” zugewiesen werden.
Randbedingung	Einbau in Brown-Field TYPO3
Randbedingung	Mitarbeiter-UI in TYPO3-Backend
Randbedingung	Nutzer-UI im Frontend
NFA	Registrierte Weiteilnahmen bestehen aus einem Wein und einem Zustand (ausstehend, eingegangen, abgelehnt).

Tabelle 7.: Initial bekannte Anforderungen

Um nähere Anforderungen zu ermitteln, werden die Befragungstechniken “Interview” und “Fragebogen” verwendet [Heinemann, 2021].

### 4.1. Interview mit Product Owner

Zunächst wird ein Interview mit dem Product Owner (PO) geführt. Ziel dieses Interviews ist es, konkrete Fragen zu Anforderungen zu klären und somit klare Anforderungen zu formulieren. Aufgrund der individuellen Gesprächsführung wurde sich für ein “teil-standardisiertes Interview” entschieden. Bei einem teil-standardisiertem Interview gibt es vordefinierte Fragen, aber auch Freiraum für Improvisation und Persönlichkeit. Für ein gutes Interview ist Vorbereitung wichtig. Daher wurden bereits sämtliche wichtigen Fragen in einem Fragebogen festgehalten. Dieser Interview-Fragebogen liegt in  $\langle\langle B \text{ Interview-Fragebogen} \rangle\rangle$  anbei. Ebenso ist Vorbereitung auf Seiten des Interview-Teilnehmers wichtig, weshalb das Interview einen Tag zuvor angesprochen wurde. Um möglichst objektive und unvorbeeinflusste Antworten zu gewährleisten wird darauf geachtet, keine Suggestivfragen zu stellen [Die SOPHISTen, 2016].

### 4.2. Online-Fragebögen für Stakeholder

Um Fragebögen für Stakeholder formulieren zu können, muss zunächst bekannt sein, wer die Stakeholder sind. *“Ein Stakeholder eines Systems ist eine Person oder Organisation, welche (direkt oder indirekt) Einfluss auf die Anforderungen des betrachteten Systems hat.”* [Die SOPHISTen, 2016, S. 9]. Daraus ergeben sich die Stakeholdergruppen: “Mitarbeiter WM” und “teilnehmende Weingüter”. Jede dieser Stakeholdergruppen sieht das System aus einer anderen Perspektive [Die SOPHISTen, 2016]. Daher ist es wichtig, Einblicke und Bedürfnisse aller Stakeholdergruppen einzuholen und die individuellen Perspektiven und Bedürfnisse dieser beim Entwurf der Fragebögen zu berücksichtigen. Ebenso ist es wichtig, die relevantesten Fragen am Anfang zu stellen, da Formulare nicht immer vollständig ausgefüllt wer-

den [Die SOPHISTen, 2016]. Somit sind auch bei einem nur teilweise ausgefüllten Fragebogen die wichtigsten Fragen beantwortet. Sämtliche Fragen an die Stakeholdergruppe “Mitarbeiter WM” wurden bereits im Interview mit dem PO beantwortet und als Anforderungen festgehalten. Der PO repräsentiert in diesem Falle die Stakeholdergruppe “Mitarbeiter WM” und steht seit geraumer Zeit mit WM in persönlichem, engen Austausch. Daher gibt es keine offenen Fragen, die diese Stakeholdergruppe beantworten könnte. Somit fällt ein Onlinefragebogen für die Stakeholdergruppe “Mitarbeiter WM” weg. Der Fragebogen der Stakeholdergruppe “teilnehmende Weingüter” liegt im Anhang unter  $\langle\langle C \text{ Online-Fragebogen Weingüter} \rangle\rangle$  bei. Um den Aufwand und somit die Hemmschwelle des Ausfüllens eines solchen Online-Fragebogens zu minimieren, werden lediglich sechs quantitative Fragen zuzüglich einem optionalen Freitextfeld, um sonstige Wünsche zum Ausdruck zu bringen, gestellt. Eine Anmeldung über Google für Google-Forms ist nicht erforderlich. Der Link zu diesem Online-Fragebogen wurde WM mit der Bitte um Weitergabe an die Probandenteilnehmer übergeben.

### 4.3. Erkenntnisse

Aus dem Interview mit dem PO ergibt sich ein Pflichtenheft. Das Pflichtenheft und das Protokoll zum Interview sind im Anhang unter je  $\langle\langle E \text{ Pflichtenheft} \rangle\rangle$  und  $\langle\langle D \text{ Interview-Protokoll} \rangle\rangle$  zu finden. Ergebnisse dieses Interviews sind zahlreiche Anforderungen und Ideen. Eine der wichtigsten Erkenntnisse stellt das projektbezogene, wöchentliche Statusmeeting dar: Jeden Donnerstag soll um 9:30 Uhr der aktuelle Stand des Projektes dem PO vorgestellt und mit ihm diskutiert und Rücksprache gehalten werden. Ziel dessen ist unter anderem die pragmatische Korrektheit laufend zu überprüfen und ggf. Anforderungen zu überarbeiten. Diese Idee wurde unmittelbar nach Beendigung des Interviews vorgeschlagen und entzieht sich somit dem Transkript.



Weitere wichtige Erkenntnisse dieses Interviews sind:

##### **Endgerät für Scanning und Scanneranwendung**

Gescannt wird von Mobiltelefonen mit einer QR-Code-App wie QRBot. QRBot ermöglicht es Nutzern für jeden aufgerufenen QR-Code eine Vorlagen-URL aufzurufen, um den gescannten Wert als Teil der URL zu übergeben [TeaCapps, 2023]. Das ist prädestiniert für API-ähnliche Webseitenaufrufe, um Weine einzuchecken.

##### **Trennung von Weinen nach Jahresauswahlproben im Frontend**

Da es  $n$  Jahresauswahlproben gibt und Weine immer genau einer Jahresauswahlprobe zugeordnet sein müssen, macht es wenig Sinn alle Weine eines Nutzers auf einmal anzuzeigen. So ist es eine Anforderung, dass die Weinansicht in zwei Ebenen unterteilt ist: Die erste Ebene soll eine Auflistung aller Jahresauswahlproben sein und in der Einzelansicht der Jahresauswahlproben sollen alle Weine aufgelistet sein, die dieser Jahresauswahlprobe angehören. Diese Weine sind ebenso anklickbar und führen zu einer Einzelansicht der Weine.

##### **Nutzerführung für WM-Angestellte**

Gescannte QR-Codes von Weinen sollen den Wein als eingegangen markieren und anschließend dem Mitarbeiter zeigen, welcher Wein eingchecked wurde. Somit dient das Scannen eines Codes ebenso zur Einsicht der Details der gelagerten Flaschen. Es soll ein Backendmodul [TYPO3 Contributors, 2023m] geben, das für den Export von CSV-Daten zuständig ist. Sonstige Aktionen sind im TYPO3-Backend mit nativen Werkzeugen erreichbar.

##### **Verschiedene Web-Ansichten**

- Jahresauswahlproben-Listenansicht
- Jahresauswahlproben-Detailansicht
- Wein-Registrierungsformular
- PDF-URL für Versandbriefe

- Registrierungsseite mit mehreren Schritten
- Mitarbeiterseite für gescannte QR-Codes
- Listen- und Detailansicht der Jahresauswahlproben im CSV-Exporter im Backend

Der Online-Fragebogen für teilnehmende Weingüter wurde über einen Monat hinweg nicht beantwortet, insofern gibt es keine Bedürfnisse dieser Herkunft zu präsentieren.

Aus der Anforderungserfassung ergibt sich als Pendant zu *⟨1 Geschäftsprozess Jahresauswahlprobe: vor der Digitalisierung⟩* der gewünschte Geschäftsprozess der Jahresauswahlprobe nach Fertigstellung dieses Projektes. Hierbei ist es wichtig, die Schnittstelle zwischen den gleichbleibenden Arbeitsschritten und den zu digitalisierenden Arbeitsschritten zu beachten. Diese Schnittstelle sollte unverändert bleiben, um eine nahtlose Integration in den restlichen, bestehenden Workflow von WM zu gewährleisten. Diese Schnittstelle stellt eine Excel-Tabelle dar. Zuvor wurde diese Excel-Tabelle von Hand, aus den Anmeldeformularen, erstellt. Nach Fertigstellung dieses Projektes soll diese Tabelle in Form von CSV-Daten aus dem Redaktionssystem exportiert werden können. Diese können Mitarbeiter von WM in Excel importieren und in kommenden Planungsschritten der Jahresauswahlprobe ohne Umstellung weiter verwenden. Das ist wichtig, da solche Umstellungen, ohne nennenswerte Verbesserungen des restlichen, von dieser Ausarbeitung unberührten Workflows, Zeit kosten, ohne Vorteile zu erbringen. Dieser Zeitverlust würde den durch die Digitalisierung erzielten Vorteil schmälern.

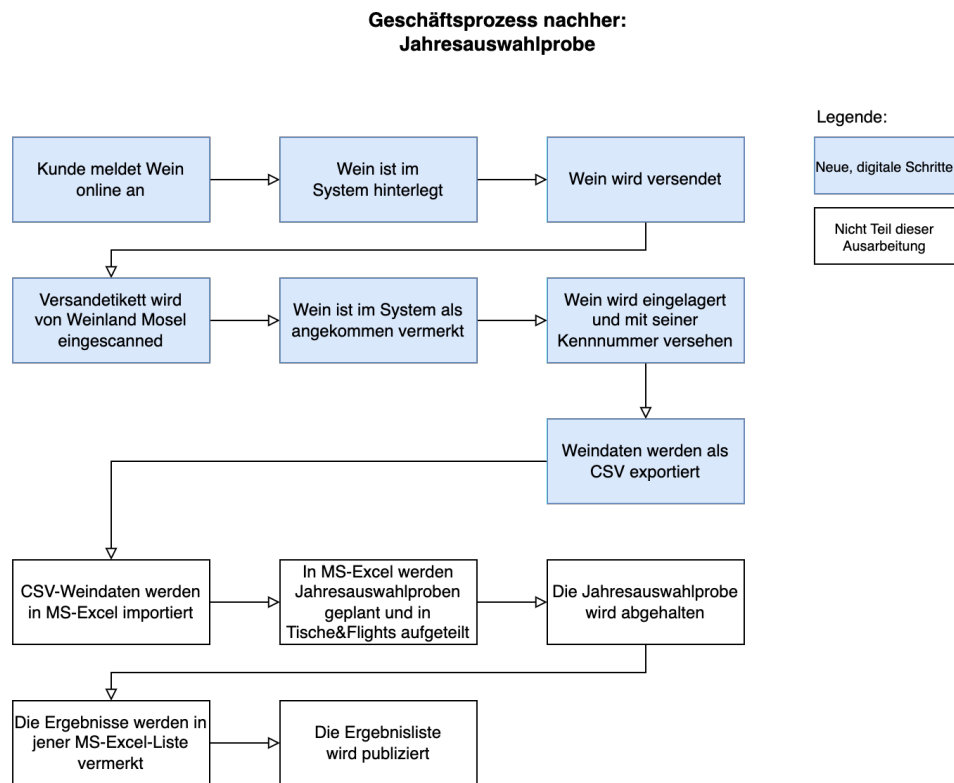


Abbildung 5.: Geschäftsprozess Jahresauswahlprobe: nach der Digitalisierung, Quelle: Eigene Darstellung

# 5. Konzeption und Umsetzung

Infolge der Anforderungsanalyse befasst sich das Kapitel “Konzeption und Umsetzung” mit der Implementation der Anforderungen in dem Brown-Field Projekt [Schwarzer, 2021] in Form einer TYPO3-Erweiterung. Es ist anzumerken, dass das aus *⟨⟨4 Anforderungserfassung⟩⟩* hervorgehende Pflichtenheft im Rahmen geplanter und opportunistischer Gespräche mit dem PO geringfügige Änderungen erfahren wird.

## 5.1. Setup der TYPO3-Erweiterung

TYPO3-Erweiterungen werden via Composer installiert [TYPO3 Contributors, 2023g]. Um eine TYPO3-Erweiterung zu erstellen, muss also ein Composer-Paket erstellt werden. Um vermeidbare Komplexität zu verhindern, wird das Composer-Paket, welches die hier betrachtete TYPO3-Erweiterung darstellt, lokal in den versionierten Ordner “packages” gelegt. Dieses Verzeichnis wird als Quelle für Composer-Pakete in der Haupt-composer.json-Datei hinterlegt. Somit wird ein Composer-Paket nur für dieses Projekt bereitgestellt, ohne den Aufwand zu betreiben, der üblicherweise mit dem Bereitstellen eines Paketes einhergeht.

Um das grundlegende Setup einer TYPO3-Erweiterung effizient durchzuführen, wird eine existierende Erweiterung mit vergleichbarem Funktionsumfang kopiert, angepasst und eingefügt, um eine valide Ordnerstruktur als Ausgangspunkt zu erhalten.

## 5.2. Digitization

Die Phase der Digitization nach Verhoef et al. befasst sich mit der digitalen Abbildung von Objekten aus der realen Welt in einer Art und Weise, die ermöglicht, dass diese elektronisch weiterverarbeitet werden können [Dougherty und Dunne, 2012, Loebbecke und Picot, 2015]. Des Weiteren befasst sich diese Phase mit der Automatisierung und Befüllung dieser Daten, wie zum Beispiel durch Web-Formulare [Verhoef et al., 2021]. Das bedeutet, dass in dieser Phase Datenobjekte definiert und implementiert werden. Ebenfalls werden anhand der Papier-Formulare Web-Formulare gebaut, damit Nutzer diese Datenobjekte erstellen können. Ein Datenobjekt besteht nach firmeninternen Konventionen aus zumindest vier Komponenten:

### **Datenbanktabelle**

Die Datenbanktabelle persistiert Informationen.

### **Domain Model**

Das Domain Model (auch Model genannt) ist eine PHP-Klasse, die jeweils die Daten einer Zeile der Datenbanktabelle abbildet.

### **Repository**

Ein Repository ist eine PHP-Klasse, die die Schnittstelle zwischen der Datenbank und der Model-Klasse darstellt.

### **Table configuration array (TCA)**

Der TCA des Modells definiert, wie diese Objekte im TYPO3-Backend dargestellt werden und bearbeitbar sind.

[TYPO3 Contributors, 2023c].

Im Folgenden wird ein semiformales Diagramm der Objekte und ihren Relationen angefertigt und in Rücksprache mit dem PO finalisiert.

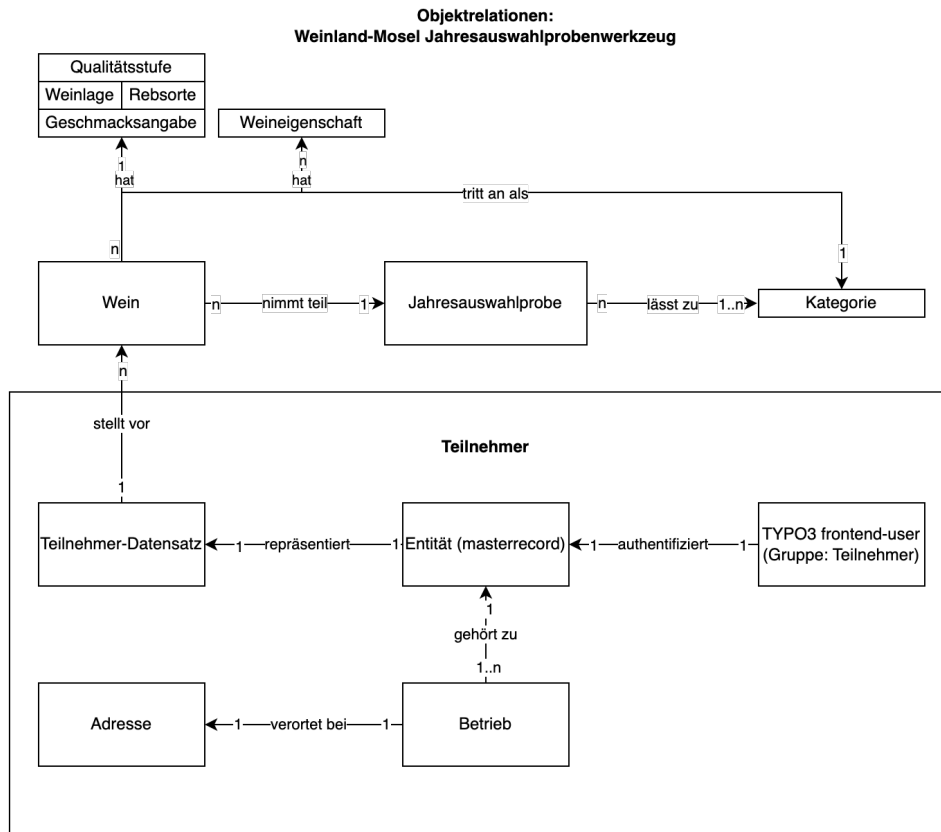


Abbildung 6.: Objektrelationen: Weinland Mosel Jahresauswahlprobenwerkzeug, Quelle: Eigene Darstellung

Nachdem in Erfahrung gebracht wird, welche konkreten Datenobjekte benötigt werden, werden Attribute dieser Objekte dem Pflichtenheft entnommen. Diese werden in einem formalen Klassendiagramm festgehalten und in Rücksprache mit dem PO weiter bis zu festen Datentypen und Auswahlmöglichkeiten konkretisiert. Beispielsweise dass Probenkategorien durch TYPO3-Categories repräsentiert werden. Das hat den Vorteil, dass TYPO3-Categories bereits native Bestandteile eines TYPO3-Redaktionssystems sind und alle relevanten Attribute anbieten. Diese sind Titel, Parentkategorie und Beschreibung. Das Parent-Attribut ist nötig, da  $n$  dieser Attribute einen Attributbaum bilden [TYPO3 Contributors, 2023k]. Somit ist es möglich, Unterkategorien zu

erstellen. Beispiele hierfür sind die Unterkategorien “Trockener Riesling” und “Halbtrockener Riesling” für die Überkategorie “Riesling”. Rebsorte, Geschmacksangabe, Weineigenschaften und Qualitätsstufe sollen eigene Datentypen, anstatt einfacher Zeichenfolgen sein. Ziel dessen ist, dass sich Nutzer für einen vorgefertigten, nominalen Eintrag in einem Dropdown-Menü entscheiden müssen und dass diese Auswahlmöglichkeiten im TYPO3-Backend pflegbar sind. Weinlagen sind im Brown-Field-Projekt bereits vorhanden, also sollen hierfür existierende Daten wiederverwendet werden. Je Wein sollen beliebig viele Weineigenschaften auswählbar sein. Probenkategorie, Geschmacksangabe, etc, sind jeweils nur ein Element. Weitere Notizen zu diesem Gespräch sind im Anhang unter *⟨⟨G Notizen: Digitization des Weinanmelde-Formulares⟩⟩* zu finden.

Da das Klassendiagramm nicht auf eine Textseite passt, befindet es sich vollseitig im Anhang unter *⟨⟨F Klassendiagramm⟩⟩*. Die weitere Implementation der Datenobjekte ist unkompliziert und besteht hauptsächlich aus repetitivem Schreiben von SQL-Tabellen, Domain-Model-Klassen und TCAs. Um  $m, n$ -Beziehungen, wie beispielsweise der Menge der für eine Probe zugelassenen Kategorien “allowedCategories”, zwischen  $m$  “Jahresauswahlprobe”-Objekten und  $n$  “Category”-Objekten zu ermöglichen, werden MM-Tabellen (many-to-many) benötigt, um diese Beziehungen in Form zweier Foreign Keys zu speichern. Die Repository-Klassen können “leer” gelassen werden, da zu diesem Zeitpunkt keine erweiterte Auswahllogik für Datenbankabfragen benötigt wird. Wichtig ist hierbei, dass eine Repository-Klasse existiert. Alle unverzichtbaren Schnittstellen werden über die Basisklasse “Repository” geerbt [TYPO3 Contributors, 2023j].

### 5.2.1. Teilnehmerregistrierung

Ein essenzieller Teil des Jahresauswahlprobenwerkzeuges ist die Registrierung von Teilnehmern. Dieses Modul repräsentiert den ersten Berührungspunkt der Winzer mit dem System. Dem Pflichtenheft ist zu

entnehmen, dass es zwei Kategorien von Teilnehmerregistrierungen gibt:

1. Nutzer ist WM-Mitglied
2. Nutzer ist kein WM-Mitglied

Der primäre Unterschied zwischen Mitgliedern und Nichtmitgliedern ist, dass Mitglieder bereits einen Stammdatensatz hinterlegt haben. Dieser Stammdatensatz bildet die Angaben zum Weingut des zu digitalisierenden Anmeldeformulars ab. Nichtmitglieder sind dem System noch gänzlich unbekannt und müssen im Zuge der Registrierung ihres Nutzers ihre Stammdaten angeben, während sich Mitglieder lediglich einloggen müssen und eine Schaltfläche “Teilnehmer werden” betätigen. Der mit dem PO ausgearbeitete UX-Flow der Registrierung sieht vor, dass der Nutzer zunächst gefragt wird, ob er Mitglied sei oder nicht. Hierzu gibt es je einen Button. Ist der Nutzer ein Mitglied, wird er auf ein Loginformular, mit der Option zur Registrierung weitergeleitet. Nach erfolgreichem Login, wird ein Teilnehmerobjekt erstellt. Wählt der Nutzer “Nein, ich bin kein Mitglied”, wird er auf ein Registrierungsformular weitergeleitet, um einen Nichtmitgliederaccount anzulegen. Im Zuge dieser Registrierung werden Stammdaten zum Weingut angefragt. Dieser Schritt übersetzt unter anderem den “Einreicher”-Teil des ursprünglichen Anmeldeformulars, anbei in  $\langle\langle A \text{ Anmeldeformular Jahresauswahlprobe} \rangle\rangle$ .



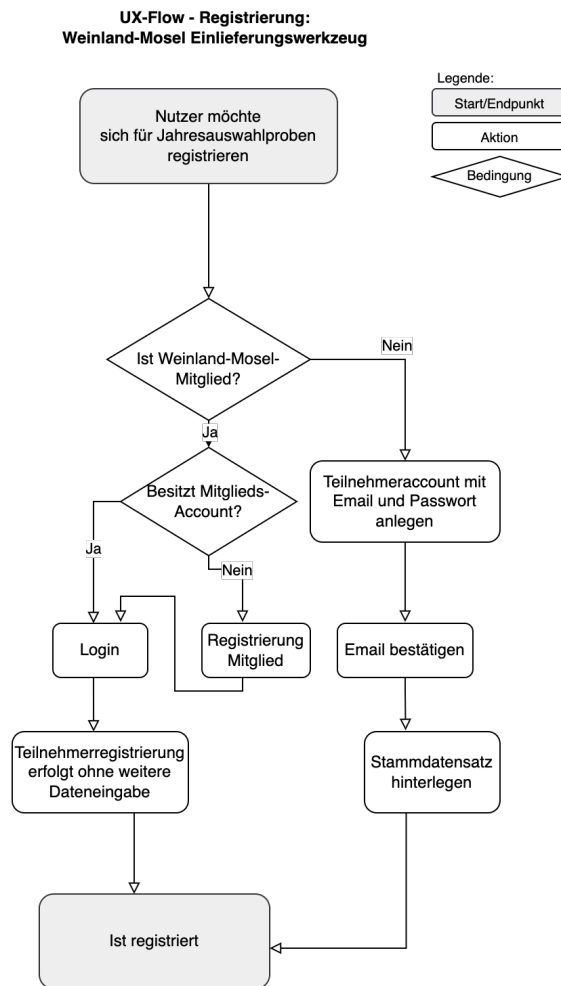


Abbildung 7.: UX-Flow: Registrierung, Quelle: Eigene Darstellung

Da das Brown-Field-Projekt bereits Accountlogins und -registrierungen implementiert und nutzt, wird auf diese Lösungen zurückgegriffen, um einen einheitlichen Workflow beizubehalten. Accountregistrierungen werden über den “femanager” [in2code, 2023] realisiert, während Logins via TYPO3s nativem Frontend-Nutzer-Login gelöst werden. Das ist explizit von “femanager” angedacht: *“Note: Login and a I forgot my password function is part of the core and not part of femanager.”* [in2code, 2023]. Im Folgenden wird der Registrierungsprozess im Detail beschrieben:

Grundlegend gibt es drei relevante Nutzerzustände vor der Registrierung:

1. Kein Mitglied
2. Mitglied, mit Konto
3. Mitglied, ohne Konto

Diese Prozesse sehen wie folgt aus:

### **Kein Mitglied**

Ist ein Nutzer kein Mitglied, so muss er zunächst einen Account erstellen. Anfangs wählt dieser Nutzer “Ich bin kein Mitglied” auf der Registrierungsseite aus. Daraufhin navigiert der Browser zu einem Registrierungsformular. Hierfür muss eine Email-Adresse, ein Passwort vergeben und den Nutzungsbestimmungen zugestimmt werden. Im Anschluss erhält der Nutzer eine Bestätigungsemail mit einem Aktivierungslink. Wird dieser Link geöffnet, wird der Account freigeschalten und ein Login-Feld erscheint. Nach erfolgreichem Login wird der Nutzer mit einem Stammdatenformular konfrontiert. Dabei handelt es sich um Angaben zum teilnehmenden Weingut. Wird dieses Formular abgeschickt, ist die Teilnehmerregistrierung abgeschlossen.

### **Mitglied, mit Konto**

Ist ein Nutzer ein WM-Mitglied und hat bereits ein Mitgliedskonto, muss dieser auf der Registrierungsseite “Ich bin ein Mitglied” auswählen. An dieser Stelle navigiert der Browser zu einem Login-Formular. Hier kann sich das Mitglied anmelden. Tut es dies erfolgreich, erstellt der Controller einen neuen Teilnehmer-Eintrag für den Frontend-Nutzer und fügt den Frontend-Nutzer der Nutzergruppe “Teilnehmer” hinzu. Damit ist die Teilnehmerregistrierung abgeschlossen.

### Mitglied, ohne Konto

Ist ein Nutzer ein Mitglied, hat aber kein Mitgliedskonto, muss dieser auf der Registrierungsseite “Ich bin ein Mitglied” auswählen. An dieser Stelle navigiert der Browser zu einem Login-Formular. Auf diesem Login-Formular existiert ein Button “Jetzt registrieren”, sowie ein Hinweisertext dazu. Da der Nutzer noch keinen Account hat, muss dieser auf “Jetzt registrieren” klicken. Daraufhin navigiert der Browser zu einem Registrierungsformular, das eine Email-Adresse, eine Mitgliedsnummer, ein Passwort und die Zustimmung zur Datenverarbeitung benötigt. Ist dieses Formular abgeschickt, erhält das Mitglied eine Email mit einem Bestätigungslink. Wird dieser Bestätigungslink angeklickt, öffnet sich ein Login-Formular, beschrieben in  $\langle \langle \text{Mitglied, mit Konto} \rangle \rangle$ . Da es sich hierbei um einen Mitgliederzugang handelt, muss dieser zunächst von einem WM-Mitarbeiter freigegeben werden, bevor ein Login möglich ist.

### Umsetzung

Zunächst wurde ein simples Weichen-Content-Element erstellt. Dieses Content-Element hat die Parameter “question”, “answ-1-link”, “answ-1-text”, “answ-2-link” sowie “answ-2-text”. Der Zweck dieses Content-Elementes ist es, Nutzer basierend auf einer ausformulierten Frage auf eine von zwei Seiten weiterzuleiten. Anschließend wurden Registrierungen über Femanager-Plugin-Content-Elemente realisiert. Anpassungen der versendeten Emails erfolgen durch Überschreiben der Email-Templates von “femanager”. Weiterleitungen zu bestimmten Seiten, nachdem ein Nutzer spezielle Events ausgelöst hat, können über TypoScript konfiguriert werden [in2code, 2023]. Logins werden über das TYPO3-native Loginformular gelöst. Im TYPO3-Loginformular können Weiterleitungen zu spezialisierten Seiten im Backend-UI festgelegt werden [TYPO3 Contributors, 2023a]. Für alle funktionalen Belange wird ein TYPO3-Pluginobjekt registriert. Dieses Pluginobjekt verfügt über einen ActionController, der Nutzeranfragen an PHP-Funktionen (“Actions”) bindet. In diesen Actions werden Fehlerbehandlungen durchgeführt, Datenmodelle der Domäne erstellt und in der Datenbank persistiert sowie

Daten für die Anzeige im Frontend der Webseite aufbereitet [TYPO3 Contributors, 2023i]. Neue Datenobjekte werden in Repository-Objekten registriert [TYPO3 Contributors, 2023j]. Diese Repositories sind Aggregate des Controllers, werden jedoch nach dem “Inversion of Control”-Prinzip via Dependency Injection instanziiert und ActionController-Objekten über öffentliche Methoden übergeben [TYPO3 Contributors, 2023b]. Als problematisch erweisen sich hierbei bidirektionale Verbindungen zwischen Datenmodellen, wenn die Foreign Keys über das SQL-Schlüsselwort “AUTO\_INCREMENT” in der Datenbank generiert werden. Beispielsweise muss ein Masterrecord, der Betriebsinformationen speichert, bidirektional an ein Teilnehmerobjekt gebunden werden. Hierzu wird jedem der Elemente jeweils der Foreign Key des anderen übergeben. Als Foreign Keys werden hierfür die jeweiligen Unique Identifier (UIDs) herangezogen, da diese Werte durch “AUTO\_INCREMENT” auf der Datenbankebene erzeugt werden und garantiert einzigartig je Datenbanktabelle sind [W3Schools, 2023c]. Es gilt also, dass ein Masterrecord  $a$  die TeilnehmerUID von einem Teilnehmer  $b$  hält und dass  $b$  die MasterrecordUID von  $a$  hält. Die Problematik hierbei ist, dass diese UIDs erst nach dem Persistieren in der Datenbank bekannt sind, da diese Werte erst im Zuge der Persistierung erstellt werden [W3Schools, 2023c]. Die Lösung hierfür ist, beide Elemente zu erstellen und zu persistieren und erst danach ihre UIDs gegenseitig bekannt zu machen um sie anschließend erneut zu persistieren.

### 5.2.2. Weinregistrierung

Ein Basismerkmal des Jahresauswahlprobenwerkzeuges ist die Möglichkeit, Weine zu Jahresauswahlproben anzumelden. Dieser Schritt übersetzt unter anderem die verbleibenden Formfelder des ursprünglichen Anmeldeformulares, anbei in  $\langle\langle A \text{ Anmeldeforumular Jahresauswahlprobe} \rangle\rangle$ , in den digitalen Workflow. Für die Weinanmeldung spielt die Mitgliederschaft eines Teilnehmers keine Rolle. Es wird lediglich ein Frontend-Nutzer der Nutzergruppe “Teilnehmer” erfordert. Dieser Nutzer hat, wenn angemeldet, Zugriff auf eine Auflistung aller zeitlich freigegebenen

Jahresauswahlproben. Soweit der Registrierungszeitraum dieser Jahresauswahlprobe den aktuellen Zeitpunkt miteinschließt, wird eine “Jetzt Wein anmelden”-Schaltfläche angeboten. Dadurch, dass Anmeldeformulare elektronisch und automatisiert verarbeitet werden, sinkt der Aufwand, der seitens WM für Anmeldungen geleistet werden muss. Das ist so, da diese Formulare nun nicht mehr von Mitarbeitern bearbeitet werden müssen. Davon profitiert WM, da diese Zeit nun für andere Aufgaben genutzt werden kann.

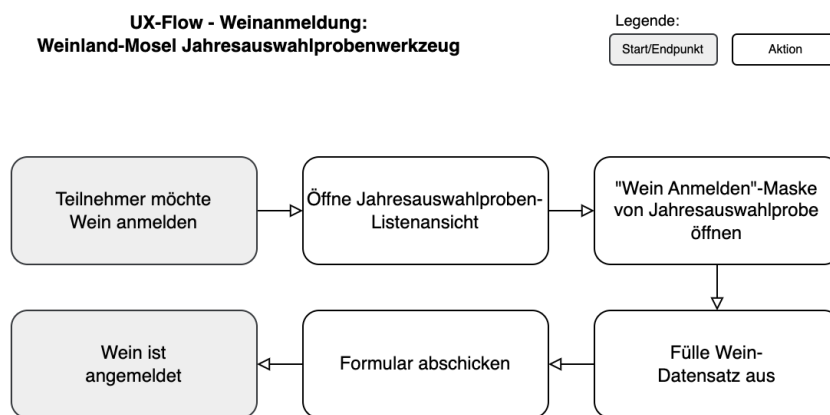


Abbildung 8.: UX-Flow: Weinanmeldung, Quelle: Eigene Darstellung

### IT-Sicherheit

Es ist wichtig zu erwähnen, dass solche Überprüfungen, wie das Aktivsein eines Registrierungszeitraumes einer Jahresauswahlprobe, grundsätzlich im Backend, d.h. serverseitig auf der betroffenen Webseite (in diesem Beispiel der Weinanmeldungsseite) durchgeführt werden. Das Verstecken der zugehörigen Schaltfläche im Frontend dient lediglich der User-Experience und stellt keine Sicherheitsvorkehrung dar. Das ist essenziell, da eine URL, auch wenn für sie keine Schaltfläche existiert, dennoch aufgerufen werden kann. Da Jahresauswahlprobennummern, also UUIDs, fortlaufend sind, ist es trivial URLs für Weinanmeldungen beliebiger Jahresauswahlproben herzustellen. Insofern ist es von großer Wichtigkeit, sicherzustellen, dass der Server solche Anfragen grundsätz-

lich selbst prüft und gegebenenfalls verneint.

### Das Formular

Aufgrund dessen, dass TYPO3 die Fluid Templating Engine verwendet [TYPO3 Association, 2023], werden Formulare und Formfelder mit den entsprechenden Fluid-Form-ViewHelpers aufgebaut. Diese View-Helper repräsentieren und erstellen gleichnamige HTML-Tags und fügen diesen spezielle Attribute zur Identifizierung in Submit-Aufrufen hinzu [TYPO3 Contributors, 2023e]. Grundsätzlich entstehen hierbei drei Kategorien von Werten, die es im Formular abzubilden gilt:

**Inputfelder** sind triviale Formfelder, die nicht durch andere Datensätze beschränkt werden. Beispiele für Inputfelder sind: Weinbeschreibung, Jahrgang und Alkoholgehalt. Inputfelder werden mit simplen Input-Tags umgesetzt und erhalten nach Bedarf *required* und *pattern*-Attribute. Diese Attribute beschreiben jeweils, ob ein Formfeld ein Pflichtfeld ist und mit welcher Regular Expression der Formfeldinhalt abzugleichen ist [W3Schools, 2023a]. Die Formfeldwerte können unverändert in der Datenbank persistiert werden.

**SelectSingle** sind Formfelder, die dem Nutzer eine Auswahl aus  $n$  Elementen aus anderen Datenbanktabellen geben. Der Nutzer muss sich für genau ein Element entscheiden. Beispiele für SelectSingle-Formfelder sind: Weinlage, Qualitätsstufe, Rebsorte und Geschmacksangabe. SelectSingle-Formfelder werden durch Select-HTML-Tags abgebildet. Der TYPO3-Form-ViewHelper für “Select” akzeptiert eine Liste an Auswahlmöglichkeiten und erstellt selbstständig Option-HTML-Tags für diese. Die Formfeldwerte von SelectSingle-Formfeldern sind die UUIDs des jeweils ausgewählten Elementes [TYPO3 Contributors, 2023e].

Aufgrund dessen, dass das Weinlagen-Dropdown-Menü über 170 Einträge führt, wird eine Suchmöglichkeit implementiert. Diese ist lediglich ein Textfeld, das bei jeder Eingabe allen Option-Tags der Weinlage, deren Anzeigewert nicht dem Suchwert entspricht, das Stilattribut “display: none;” auferlegt. Somit sind diese nicht mehr sichtbar.

Eine komplexe Ausnahme stellt das SelectSingle-Formfeld “Category” dar, da TYPO3-Kategorien als Baumstrukturen abgebildet werden [TYPO3 Contributors, 2023k]. Um die Eltern-Kind-Beziehungen der Baumstruktur ersichtlich zu machen, werden die Option-HTML-Tags rekursiv gerendert. Zunächst werden sämtliche Kategorien, deren Parent Identifier (PID) 0 ist, dargestellt. Diese Elemente sind direkte Kinder des unsichtbaren Wurzelementes. Für jede dieser Kategorien  $a$  wird nun ein Fluid-Partial aufgerufen, das alle Kategorien  $b$  darstellt, für die gilt:  $b.pid = a.uid$ . Diese Darstellung erfolgt durch einen erneuten rekursiven Aufruf dieses Partial. In jeder Darstellung wird der Kategorienname, geprefixt mit  $n$  Leerzeichen, dargestellt, mit  $n = Rekursionstiefe$ . Somit entsteht ein Dropdown-Menü, das alle Kategorien in einer eindimensionalen Liste darstellt. Diese Liste ist nach einer preorder-Traversierung des Kategoriebaumes sortiert und desto tiefer ein Element im Baum ist, desto weiter ist es eingerückt. Damit sieht das Dropdown-Menü aus wie eine Baumstruktur.

Diese Herangehensweise erzeugt schlüssigen, lesbaren Programmcode und lässt sich unkompliziert umsetzen. Das senkt Entwicklungskosten, wovon WM profitiert. Rekursiv aufgerufene For-Schleifen, die sich selbst erneut für alle Elemente aufrufen, können zu Performanzproblemen führen [Schwarzer, 2022]. Daher wird im Folgenden die Zeitkomplexität dieser Rekursionsfunktion betrachtet. Für diese Funktion kann kein Master-Theorem angewandt werden, da es sich hierbei nicht um einen “Divide-and-Conquer-Algorithmus” handelt. Das ist so, da das in der Rekursion weitergereichte Problem nicht kleiner wird, sondern gleich groß bleibt. Demnach ist  $b = 1$ . Das verletzt die Bedingung  $b > 1$  des Master-

Theorems, definiert als  $T(n) = a * T(\frac{n}{b}) + f(n)$  [Schwarzer, 2022]. Der Algorithmus besteht aus  $m \in \mathbb{N}$  verschachtelten For-Schleifen gleicher Länge. Somit ist die Zeitkomplexität  $O(n^m)$ . Normiert dargestellt beträgt die Zeitkomplexität  $O(n^2)$ . Das lässt sich experimentell bestätigen.

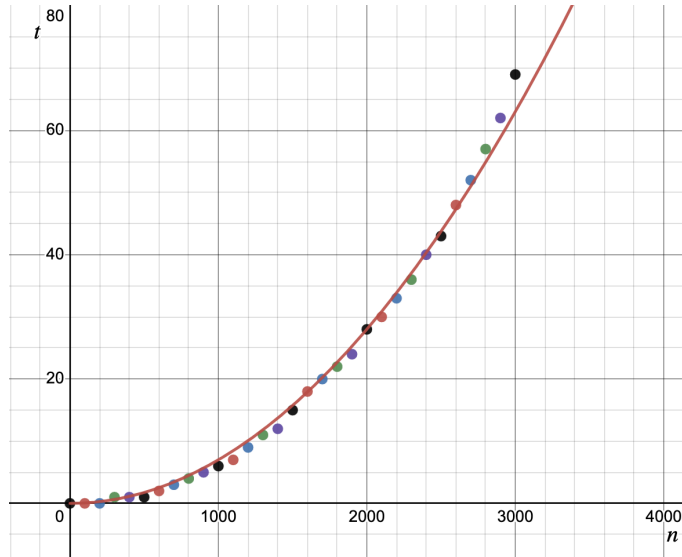


Abbildung 9.: Stichprobenartige Laufzeitanalyse des Kategorie-Renderers, gegenüber einer quadratischen Kurve, Quelle: Eigene Darstellung

Auf Optgroup-HTML-Tags wird bewusst verzichtet. Grund dafür ist, dass Optgroup-Elemente an sich nicht im Dropdown-Menü auswählbar sind. Das stellt ein Problem dar, da beispielsweise die Kategorie “Riesling”, die die Unterkategorien “Trockener Riesling” und “Halbtrockener Riesling” beinhalten könnte, auch direkt auswählbar sein sollte. Zudem besitzen Kategorie-Elemente kein Attribut das auf die Präsenz von Unterkategorien hindeutet [TYPO3 Contributors, 2023k], womit eine Unterscheidung zwischen Baumblättern und -zweigen nicht ohne weiteres möglich wäre. Diese Entscheidung wäre jedoch benötigt, um zwischen einem Optgroup-Tag und einem Option-Tag abzuwägen.



**SelectMultiple** sind Formfelder, die dem Nutzer eine Auswahl aus  $n$  verschiedenen Elementen aus einer anderen Datenbanktabelle geben. Der Nutzer kann sich für eine beliebige Auswahl dieser, eingeschlossen keinen, entscheiden. Ein Beispiel für SelectMultiple-Formfelder sind Weineigenschaften. TYPO3-Fluid implementiert hierfür keinen View-Helper [TYPO3 Contributors, 2023e], also wird eine eigene Lösung entworfen: Der Nutzer soll aus einer Menge  $A$  wählen. Für alle Elemente  $a \in A$  wird ein Checkbox-Feld erstellt. Dieses Element trägt den Anzeigewert “<a.title>”, den ID- und Namenswert “<formfeldname>-<a.uid>” und den Formularwert “<formfeldname>-true”. Ist also eine dieser Checkboxes angehakt, hat sie den zuvor genannten Formularwert. Falls nicht, trägt sie keinen Formularwert. Weil alle angehakten Checkboxes dieses Formfeldes den selben Namenswert tragen, ist es phpseitig trivial eine Liste aller angehakten Checkbox-Namen dieses Formfeldes aus der Liste aller Formfeldparameter zu extrahieren. Hierfür wird die eingebaute PHP-Funktion “array\_keys” verwendet. Diese Funktion gibt alle Keys eines Arrays in Form eines numerisch indizierten Arrays zurück. Der optionale Parameter “filter\_values” bestimmt, dass ausschließlich die Keys der Key-Value-Pairs, die einen bestimmten Wert (der Formwert) tragen, extrahiert werden [The PHP Group, 2023a]. D.h., der Funktionsaufruf filtert alle Keys und somit alle Formfeld-IDs des Formfeldparameter-Arrays heraus, die den Wert “<formfeldname>-true” haben. Das ist eine Liste aller Formfeld-Namen der Checkboxes des SelectMultiples, die angehakt wurden. Mit der eingebauten PHP-Funktion “array\_map” wird nun eine Operation auf alle Werte der Liste angewandt, die “strlen(<formfeldname>-)” Zeichen, von links ausgehend, vom Formfeld-Name entfernt. Somit wird beispielsweise der Formfeld-Name “winekind-18” zu “18” transformiert. Übrig bleiben die extrahierten UUIDs aller angehakten Elemente  $a$ , in Form einer Zeichenkette. Über die eingebaute PHP-Funktion “intval” ist es trivial diese zu Zahlen zu übersetzen, wodurch die tatsächlichen Objekte aus der Datenbank angefragt werden können.

### 5.2.3. PDF- und QR-Code-Generierung

Das dynamische Erstellen und Ausgeben des Datenblattes als PDF ist ein essenzieller Bestandteil des Jahresauswahlprobenwerkzeuges, da dieses PDF die Schnittstelle zwischen ankommenden Weinen und dem System darstellt. Daher ist der Gewinn, der durch dieses Werkzeug generiert wird, ohne dieses Dokument stark eingeschränkt, da einkommende Weine händisch, von Mitarbeitern, zugeordnet werden müssten. Wie im *⟨2 Stand der Forschung⟩* erläutert wird, werden für die dynamische Generierung dieses PDFs die Bibliotheken “chillerlan/php-qrcode” und “mpdf/mpdf” herangezogen und über Composer installiert.

#### QR-Code-Generierung

Der QR-Code beinhaltet lediglich die Wein-UID anstatt einer vollständigen URL. Hintergrund ist, dass die benötigte URL, um einen Wein einzuscannen, bis auf die Wein-UID immer identisch ist. Somit wird Redundanz vermieden. Es ist Aufgabe der QR-Code-App, die den Code einscann, aus der Wein-UID eine vollständige URL herzuleiten. Um effizient zu arbeiten, wird der QR-Code zu einem base64-kodierten Bild gerendert. Das ist der Standardrückgabewert des QR-Code-Generators und erfordert somit keine nähere Konfiguration. Ebenfalls lässt sich ein base64-kodiertes Bild als Quell-URL eines IMG-HTML-Tags angeben, womit das Bild eingebettet ist. Das spart Arbeitszeit, die sonst in das anderweitige Einbetten eines Bildes in einem PDF mit “mpdf” fließen müsste. Die QR-Codegenerierung funktioniert konkret, indem ein neues QRCode-Objekt der QRCode-Klasse erstellt wird. Diese Klasse nimmt ein QROptions-Objekt im Konstruktor, das in diesem Falle einige Stilattribute mit sich trägt. Das QRCode-Objekt bietet nun eine Methode “render” an, die sofern nicht anders konfiguriert, den QR-Code als base64-kodiertes Bild zurückgibt [Rochko, 2022b].

### PDF-Generierung

Firmenintern ist es Standard, das Aussehen sowie die Inhalte der PDF-Dokumente, die mit “mpdf” erzeugt werden, mit HTML zu definieren, das an “mpdf” gereicht wird. Um die Gestaltung und die Präsentation von Variablen in der HTML-Zeichenfolge technisch kontinuierlich mit dem restlichen Projekt zu halten und um eine gute Wartbarkeit zu gewährleisten, wird diese HTML-Zeichenfolge mit TYPO3-Fluid getemplated. Das heißt, dass eine HTML-Templatedatei bereitgestellt wird, diese mit TYPO3-Fluid befüllt und in PHP gerendert wird, um eine HTML-Zeichenkette als Ergebnis zu erhalten. Hierfür wird ein TYPO3-StandaloneView instanziiert, mit einem Pfad zur Template-Datei konfiguriert, Variablen angegeben, die in Fluid verfügbar sein sollen und anschließend über die “render”-Methode zu einer Zeichenkette gerendert [TYPO3 Contributors, 2023o]. Anschließend wird ein “mpdf”-Objekt erstellt und mit einer rudimentären Konfiguration in Form eines Arrays im Konstruktor konfiguriert. Diese Konfiguration definiert in diesem Falle Seitenabstände, Papierformat, Zeichenkodierung und Schriftarten. Abschließend wird dem “mpdf”-Objekt das zuvor generierte HTML übergeben, über die Methode “OutputBinaryData” in Form von Bytes zurückgegeben und in einer Variable gespeichert [Mpdf Contributors, 2023c]. Um dieses PDF-Dokument über die Verbindung an den Nutzer zu übertragen, wird ein TYPO3-Response-Objekt erstellt. Über dieses Response-Objekt werden einige Header gesetzt und direkt übertragen. Diese Header sind Content-Type und Content-Length. Abschließend werden als Response-Body die Bytes des generierten PDFs übertragen. Damit ist die Verbindung beendet und das PDF zum Nutzer übertragen.

#### 5.2.4. Jahresauswahlproben- und Wein-Detailansichten

Weine und Jahresauswahlproben sollen unter bestimmten Gegebenheiten einsehbar sein. Hierzu gibt es eine Auflistung aller Jahresauswahlproben. Diese sind anklickbar, um eine Detailansicht der ausgewählten Jahresauswahlprobe zu öffnen. Hier wird neben Metadaten der Jahresauswahlprobe auch eine Liste aller zur Einsicht berechtigten Weine angezeigt. Diese Weine sind anklickbar, um auf eine Detailansicht der Weine zu gelangen. Die Detailansichten für Jahresauswahlproben und Weine benötigen spezielle Autorisierung. Diese sind: Jahresauswahlproben sind nur einsichtig, wenn sich das aktuelle Datum innerhalb des Sichtbarkeitszeitraumes der Jahresauswahlprobe befindet. Detailansichten für Weine sind immer für den zugehörigen Teilnehmer einsichtig. Nach Abschluss einer Jahresauswahlprobe sind alle ihr angehörigen Weine öffentlich einsichtig. Das hat den Hintergrund, dass Jahresauswahlproben Blindverkostungen sind und niemand die Möglichkeit haben sollte, im Voraus Informationen über die teilnehmenden Weine in Erfahrung zu bringen. Die Ergebnisse der Jahresauswahlproben sind öffentlich, also sind es die Weine nach Abschluss einer Jahresauswahlprobe auch. Mitarbeiter von WM, wenn sie mit einem solchen Account angemeldet sind, haben immer Einsicht in Wein-Detailansichten. Die Wein-Detailseite verfügt außerdem über einen “Datenblatt drucken”-Button, der auf das Datenblatt-PDF verlinkt. Dieser Button ist nur für den zugehörigen Teilnehmer und Mitarbeiter verfügbar. Diese Daten, Fakten und Restriktionen werden serverseitig kontrolliert, um Manipulationen des Nutzers auszuschließen. Um das zu realisieren werden Daten mit Fluid-Templates konditionell dargestellt und über Fluid-ViewHelper Links zu anderen Ansichten generiert. Diese ViewHelper übergeben Parameter. Die hierfür relevanten Parameter der verschiedenen Ansichten sind beispielsweise Wein-UIDs und Jahresauswahlproben-UIDs. Um Informationen über den angemeldeten Nutzer, wie beispielsweise seiner Teilnehmernummer oder seiner Nutzergruppenzugehörigkeit, zu erlangen, wird sich der Extbase-nativen Domain-Model-FrontendUser-Klasse

bedient [TYPO3 Contributors, 2023n]. Mit Abschluss der Phase der Digitalization können alle Datenstrukturen im TYPO3-Backend händisch angelegt, eingesehen, gelöscht, bearbeitet und im Frontend von Nutzern befüllt werden.

## 5.3. Digitalization

In der Phase *Digitalization* werden bestehende Geschäftsprozesse so verändert, dass mit digitalen Werkzeugen und Datenmodellen gearbeitet werden kann [Li et al., 2016]. Damit baut diese Phase auf der vorherigen Phase “Digitization” auf, um mit den dort implementierten Datenmodellen zu arbeiten. Im Folgenden werden die Umsetzungen der erforderlichen Geschäftsprozesse beschrieben.

### 5.3.1. Schnittstelle QR-Code-Scanner

Es ist angedacht, dass Mitarbeiter über die App “QRBot” den QR-Code auf dem Datenblatt einscannen. Diese App ermöglicht es Nutzern für jeden eingescannten Code eine URL zu öffnen und den Wert des QR-Codes anstelle eines Platzhalters in der URL einzufügen. Hierfür bietet das Jahresauswahlprobenwerkzeug eine Schnittstelle bzw. eine Seite an, die eine Wein-UID als URL-kodierten GET-Parameter annimmt und diesen Wein als “eingegangen” markiert. Um sicherzustellen, dass Weine nicht unautorisiert markiert werden, müssen sich Mitarbeiter mit einem Nutzerkonto authentifizieren. Dieses Nutzerkonto muss Teil einer Mitarbeiter-Nutzergruppe sein. Das Nutzerkonto wird von Redakteuren oder Administratoren im TYPO3-Backend erstellt und benötigt keiner Registrierung. Nach einer Anmeldung bleibt diese Sitzung aktiv und verfällt erst nach längerer Inaktivität. Wird ein Wein als “eingegangen” markiert, wird der betroffene Teilnehmer per Email informiert. Hierzu wird die FluidEmail-Klasse des TYPO3-Cores herangezogen. Sollte ein Wein bereits als “eingegangen” markiert sein, wird keine Email verschickt und dem Mitarbeiter kommuniziert, dass keine Änderungen vorgenommen wurden. Abschließend werden im Frontend allgemeine Daten

zum Wein angezeigt, damit sich Mitarbeiter sicher sein können, den richtigen Wein eingescannt zu haben.

### 5.3.2. CSV-Export

Das letzte Glied des analogen Prozesses, den es zu digitalisieren gilt, ist der Datenexport der Weindaten je Jahresauswahlprobe. TYPO3s ListView bietet einen nativen CSV-Exporter an [Pixelant, 2023], jedoch wird dieser nicht den Anforderungen gerecht, da das benötigte CSV-Dokument Daten verschiedener Datenbanktabellen im Verein beinhalten muss. Der Native CSV-Exporter gibt jedoch lediglich Inhalte einer Datenbanktabelle aus [Pixelant, 2023]. Um eine reibungslose Integration in die restlichen Prozesse von WM zu gewährleisten, muss das exportierte CSV das selbe Format haben, wie bisher bestehende Excel-Dateien. Dieses Format ist durch genaue Spaltennamen, Spalteneihenfolgen und den Arten von Daten, die angefragt werden, definiert. Um einen benutzerdefinierten Exporter im Backend bereitzustellen, wird ein Backendmodul registriert, das immer im linken Reitermenü verfügbar ist. Dieses Backendmodul stellt eine Listenansicht aller Jahresauswahlproben bereit. Restriktionen gibt es hierbei keine, da diese Listenansicht nur im Backend, nur für Administratoren und Redakteure, verfügbar ist. Jeder Listeneintrag ist eine Verlinkung auf eine Detailansicht der jeweiligen Jahresauswahlprobe. Diese Detailansicht ist der CSV-Exporter. Hierfür werden alle Weine aus der Datenbank angefragt, die der jeweiligen Jahresauswahlprobe zugeordnet sind. Zusammenhängende Relationen, wie zum Beispiel Teilnehmerdatensätze, die Attribute dieser Weine sind, werden automatisch von Extbase angefragt [TYPO3 Contributors, 2023h]. Das CSV-Dokument wird über die PHP-native Funktion “fputcsv” generiert. Diese Funktion nimmt einen numerisch indizierten Array und schreibt eine darauf basierende CSV-Zeile in eine Datei [The PHP Group, 2023b]. Um gut wartbaren PHP-Code zu erzeugen, werden alle CSV-Zeilen in einem zweidimensionalen Array, der das gesamte CSV-Dokument darstellt, durch alphanumerisch indizierte, innere Arrays abgebildet. Somit ist bei jeder Wertzu-

weisung der zugehörige Spaltenname als Array-Key ersichtlich. Über die nativen PHP-Funktionen “array\_keys” und “array\_values” wird dieser zwar gut lesbare, aber mit “fputcsv” inkompatible Array zu einer Reihe kompatibler Array konvertiert. Hierbei werden durch “array\_keys” die Kopfzeile und durch “array\_values” die Datenzeilen extrahiert. Diese Zeilen werden anschließend mit “fputcsv” in die Zielfeld geschrieben. Als Zielfeld wird eine temporäre Datei im Arbeitsspeicher angegeben, um das fertig erzeugte CSV-Dokument als Zeichenkette in PHP zu erhalten. Das spart vermeidbare Festplattenschreib- und Lesezugriffe. Im Fluid-Template des Backendmoduls wird das generierte CSV-Dokument in einem nicht beschreibbaren Textarea-Feld präsentiert. Um WM weitere Arbeitszeit zu ersparen, wird eine Download-Funktion für CSV-Dateien angeboten. Das erspart das manuelle Kopieren und Abspeichern von CSV-Zeichenketten durch IT-Fachfremde, reduziert damit die Anzahl an benötigten Übergangsschritten in weitere Prozesse und reduziert somit die Komplexität der Umstellung. Auch im Interesse, Arbeitszeit in der Umsetzung zu sparen, wird diese Downloadfunktion javascript-seitig umgesetzt. Dadurch ist der Download in derselben Action, die CSV für das Textarea-Feld generiert, implementiert. Somit muss weder abstrahiert werden, noch ein weiterer CSV-Exporter gebaut werden. Hierfür wird ein EventHandler auf den Download-Button angewandt, der bei Betätigung ein verstecktes “a”-Element erstellt, über das HTML-Attribut “download” des “a”-Elementes den Download-Dateinamen festlegt und als “href”-HTML-Attribut eine Blob-URL zuweist. Diese Blob-URL wird über ein Blob-Objekt generiert. Wird nun der Download-Button gedrückt, wird javascript-seitig eine Datei erzeugt und gespeichert [Vora, 2022].

# 6. Ergebnisse

## 6.1. Praktische Ergebnisse

Die TYPO3-Erweiterung ist umgesetzt und erfüllt die vereinbarten Anforderungen. Mitglieder und Nichtmitglieder können Teilnehmernutzer anlegen und Weine zu Jahresauswahlproben anmelden. Teilnehmer bekommen ein PDF-Dokument mit Weindaten und einem QR-Code zur späteren Zuordnung bereitgestellt. WM-Mitarbeiter können diesen QR-Code einscannen, um Weine als “angekommen” zu markieren. Redakteure von WM können Zugriffsrechte und Verhalten der Jahresauswahlproben auf verschiedene Weisen einschränken. Sie können den Sichtbarkeitszeitraum, den Anmeldezeitraum und den Probezeitraum festlegen, der definiert ab wann teilnehmende Weine öffentlich sind. Sie können Jahresauswahlproben und damit deren Anmeldeformulare auf festgelegte Probenkategorien beschränken. Mitarbeiter können Weindatensätze, getrennt nach Jahresauswahlproben, als CSV-Dokumente exportieren und somit den verbleibenden Geschäftsprozess wie gehabt fortsetzen. Praktische Präsentationen gegenüber dem PO bestätigen die Umsetzung der Anforderungen und stellen die Basis für weitere Iterationen der Entwicklung dar.

## 6.2. Forschungsergebnisse

Im Zuge dieser Ausarbeitung wurden Teilfragen beantwortet. Diese Antworten zeigen auf, wie die Anmeldung und Zustellung von Weinen für Weinproben des Regionalverbunds für Weine in der Weinregion Mosel effizient und profitabel durch eine TYPO3-Erweiterung realisiert werden können.



**Nach welcher Methodik sollten Digitalisierungsprojekte im Maßstab des behandelten Projektes entwickelt werden, um effizient und profitabel zu sein?**

In der Literaturrecherche zur Wahl von Methodiken in der Softwareentwicklung und Digitalisierung im Kontext der effizienten und profitablen Umsetzung des hier behandelten Projektes stellten sich komplexe und aufwändige Modelle als ungeeignet heraus. Vielmehr beeindruckten simple, agile Modelle durch ihren geringen Mehraufwand, der bei kleinen Projekten, wie der hier beleuchteten Aufgabenstellung, unabdinglich ist, um eine effiziente und profitable Umsetzung gewährleisten zu können.

**Welche Anforderungen sind an die TYPO3-Erweiterung gestellt?**

Um ein Pflichtenheft für die hier beleuchtete TYPO3-Erweiterung zu erarbeiten, wurde eine Anforderungsanalyse in Form eines Interviews mit dem PO durchgeführt. Auch wurde eine quantitative Studie in Form eines Online-Fragebogens bezüglich der Bedürfnisse der Teilnehmer angestrengt, die ohne Ergebnisse verblieb. Dieses Pflichtenheft zeigt unter anderem auf, dass Mitglieder sowie Nichtmitglieder Teilnehmer sein können, wie die Nutzerführung aussieht, welche Werkzeuge WM-Mitarbeitern zur Verfügung stehen und wie verschiedene Schnittstellen aussehen. Auch ist eine wichtige Erkenntnis, dass regelmäßige Statusmeetings mit dem PO durchgeführt werden sollten, damit pragmatisch inkorrekte Anforderungen und Merkmale rechtzeitig erkannt und korrigiert werden können. Das vollständige Pflichtenheft liegt im Anhang anbei, unter  $\langle\langle E \text{ Pflichtenheft} \rangle\rangle$ .

### **Welche QR-Code-Bibliothek ist für das behandelte Projekt gut geeignet?**

Um die Anmeldung und Zustellung von Weinen digital umsetzen zu können, ist lt. Anforderungen ein QR-Code-Generator notwendig. Generell sollten erwägte Bibliotheken aktiv gepflegt, einen gewissen Grad an Funktionalität aufweisen und für den angedachten Workflow geeignet sein. Das ist wichtig, damit sich dieser effizient, schnell und somit kostengünstig integrieren lässt. Vergleiche zwischen sechs QR-Code-Bibliotheken legen nahe, dass *chillerlan/php-qrcode* die beste Eignung der betrachteten QR-Code-Generatoren aufweist.

### **Wie wird sichergestellt, dass sich der digitalisierte Teilprozess der Weinanmeldung und -zustellung nahtlos mit dem verbleibenden Prozess integriert?**

Es ist essenziell, die Schnittstelle zwischen dem digitalisierten Teilprozess und dem verbleibenden analogen Teilprozess zu schützen. Das wird sichergestellt, indem die Ausgabe des digitalisierten Teilprozesses der Ausgabe des vorherigen, analogen Teilprozesses gleicht. Ist das gegeben, kann der neue, digitale Teilprozess effizient in den Geschäftsprozess der Jahresauswahlproben integriert werden, dies weil die darauf aufbauenden Schritte mit der Ausgabe der digitalisierten Schritte kompatibel sind.

### 6.3. Diskussion

#### **Nach welcher Methodik sollten Digitalisierungsprojekte im Maßstab des behandelten Projektes entwickelt werden, um effizient und profitabel zu sein?**

Um eine Entwicklungsmethodik für die Umsetzung einer wie in der Problemstellung beschriebenen TYPO3-Erweiterung auszuwählen, wurde eine Literaturrecherche durchgeführt. Diese Literaturrecherche lässt darauf schließen, dass sich simple, agile Methodiken, ohne nennenswerten Mehraufwand, gut für die effiziente und profitable Umsetzung kleiner Digitalisierungsprojekte, wie dem hier behandelten, eignen. Somit empfehlen sich solche Modelle für die Digitalisierung von Anmeldungen und Zustellungen der Weine für Jahresauswahlproben, da hiermit die Umsetzung effizienter und profitabler wird. Zu Beginn wurde davon ausgegangen, dass agile Entwicklungsmethodiken oder sogar “extreme programming” insbesondere für kleine, zeitkritische Problemstellungen mit veränderlichen Anforderungen und somit auch für die behandelte Problemstellung, effizient seien. M. Ayman Al Ahmar bestätigt diese Hypothese. Ahmar erklärt das durch die Agilität des “extreme programmings”, die die Entwicklungsgeschwindigkeit fördert und kurzfristige Änderungen der Anforderungen effizient gestaltet [Ahmar, 2005]. Es muss jedoch berücksichtigt werden, dass sich diese Literaturrecherche explizit auf die in dieser Bachelorarbeit beleuchtete Problemstellung bezieht. Bei Einbezug anderer Projekttypen und -beschaffenheiten weichen die geeigneten Entwicklungsmethodiken wahrscheinlich ab. Eine Empfehlung für weitere Forschung ist es daher, ähnliche Literaturrecherchen bezüglich angemessener Entwicklungsmethodiken für abweichende Projekttypen und -beschaffenheiten durchzuführen.

## **Welche Anforderungen sind an die TYPO3-Erweiterung gestellt?**

Um detaillierte Anforderungen an die TYPO3-Erweiterung in Erfahrung zu bringen, wurde eine Anforderungsanalyse durch verschiedene Befragungstechniken durchgeführt. Die verwendeten Befragungstechniken sind “Interview” und “Online-Fragebogen”. Auch ist es eine wichtige Erkenntnis des Interviews, regelmäßige Statusmeetings mit dem PO zu führen. Diese Statusmeetings erlauben das frühzeitige Erkennen und Beheben von pragmatisch inkorrekten Merkmalen und Anforderungen. Es ist wichtig zu erwähnen, dass der Online-Fragebogen unbeantwortet blieb. Das Ergebnis dieser Anforderungsanalyse ist ein detailliertes Pflichtenheft, das die Anforderungen an die TYPO3-Erweiterung detailliert beschreibt. Dieses zeigt unter anderem auf, dass Mitglieder und Nichtmitglieder Teilnehmer sein können, wie detaillierte Konzepte der Nutzerführung und verschiedene Schnittstellen aussehen und welche Werkzeuge WM-Mitarbeitern zur Verfügung stehen. Einige Anforderungen waren bereits im Voraus bekannt. Das ermittelte Pflichtenheft ist eine Erweiterung dieser. Da diese Anforderungsanalyse lediglich die Anmeldung und Zustellung von Weinen betrachtet, ist es empfehlenswert, in weiteren Ausbauten des Jahresauswahlprobenwerkzeuges Forschungen bezüglich darauf aufbauender Anforderungen durchzuführen.

## **Welche QR-Code-Bibliothek ist für das behandelte Projekt gut geeignet?**

Im Interesse eine Bibliothek zur Generierung von QR-Codes für die Umsetzung dieses Softwareprojektes zu finden, die sich effizient, schnell und somit kostengünstig integrieren lässt, wurden sechs QR-Code-Bibliotheken einander gegenübergestellt und in drei verschiedenen Bewertungskategorien verglichen. Diese Bewertungskategorien sind “Funktionalität”, “Gepflegtheit” und “Workflow-Eignung”. Jede dieser Kategorien wurde mit null bis zehn Punkten bewertet. Jede Bibliothek konnte maximal 30 Punkte erhalten. Die Bewertung erfolgte nach subjektiver Ein-

schätzung des Autors, basierend auf Faktoren wie dem Zustand der Github-Seite, der Produktwebseite, den tätigen Entwicklern, der Lizenz, der Beschaffenheit des Projektes, der Dokumentation und dem dort ersichtlichen Funktionsumfang. Dieser Vergleich deutet darauf hin, dass “chillerlan/php-qrcode” mit 30 Gesamtpunkten die beste Eignung der verglichenen Bibliotheken aufweist. Es wurde davon ausgegangen, dass “chillerlan/php-qrcode” eine gute Wahl sei, da diese Bibliothek bereits firmenintern nahegelegt wurde. Der abgehaltene Vergleich bestätigt diese Empfehlung. Dieser Erfolg erklärt sich durch ein aktiv gepflegtes Softwareprodukt mit einer Vielzahl an Entwicklern, Verwendungen, aktueller Versionsunterstützung, guter Dokumentation und projektspezifischer Eignung. Es muss jedoch beachtet werden, dass dieser Vergleich das spezifische Projekt als wichtigen Faktor mit einbezieht. Somit ist dieser Vergleich nur gültig, um eine QR-Code-Bibliothek für ein PHP-Projekt mit dem Composer-Paketmanager zu evaluieren. Der Autor empfiehlt ähnliche Vergleiche für andere Arbeitsumgebungen durchzuführen, um zu bestimmen, welche QR-Code-Bibliotheken für andere Projektkontexte angemessen sind. Ebenso nimmt dieser Vergleich nur sechs QR-Code-Bibliotheken in Betracht. Eine weitere Forschungsempfehlung ist es daher, weitere Vergleiche mit mehr Bibliotheken durchzuführen.

### **Wie wird sichergestellt, dass sich der digitalisierte Teilprozess der Weinanmeldung und -zustellung nahtlos mit dem verbleibenden Prozess integriert?**

Im Zuge der praktischen Umsetzung der in dieser Forschungsfrage beschriebenen TYPO3-Erweiterung zeigte sich das Problem der Integration der digitalisierten Weinanmeldung und -zustellung in den restlichen, von dieser Ausarbeitung unberührten Geschäftsprozess der Jahresauswahlprobe. Diese Umsetzung zeigt auf, dass es für eine nahtlose Integration in den existierenden Geschäftsprozess unabdinglich ist, dass die Ausgabe des digitalisierten Teilprozesses der Ausgabe des ersetzten, manuellen Teilprozesses gleicht. Dieser Aspekt wurde zuvor nicht bedacht.

Das könnte daran liegen, dass diese Schnittstelle nicht der primäre und auch nicht der sekundäre Fokus in der Umsetzung ist. Sie wird nicht benötigt, damit das umgesetzte Produkt intrinsisch funktioniert, ist aber dennoch unverzichtlich für eine reibungslose, praktische Verwendung. Hierbei muss jedoch berücksichtigt werden, dass es sich um ein einzelnes, konkretes Projekt handelt und sich aus diesem Grund nicht unbedingt allgemeingültige Schlüsse ableiten lassen. Eine Forschungsempfehlung ist es daher, weitere Möglichkeiten zur Integration verschiedener Teilprozesse zu recherchieren und zu evaluieren.

## 7. Fazit

Die vorliegende Bachelorarbeit befasste sich mit der Frage: “Wie kann die Anmeldung und Zustellung von Weinen für Weinproben des Regionalverbunds für Weine in der Weinregion Mosel effizient und profitabel durch eine TYPO3-Erweiterung realisiert werden?” Für die Beantwortung wurde eine Literaturrecherche bezüglich Entwicklungsmethodiken, eine Gegenüberstellung existierender Technik zur Erstellung von QR-Codes, verschiedene Befragungstechniken zur Anforderungserfassung sowie die praktische Umsetzung der TYPO3-Erweiterung angestrengt.

Aus den Ergebnissen lässt sich schließen, dass sich insbesondere “extreme-programming”-Entwicklungsmethodiken eignen, um das aus der Forschungsfrage hervorgehende Projekt zu realisieren. Es wurde gezeigt, dass sich die Bibliothek “chillerlan/php-qrcode” aufgrund herausragender Projektpflege und Workflow-Eignung anbietet, die mit dem Projekt verbundenen Anforderungen zu erfüllen, die die Erstellung von QR-Codes beschreiben. Die Umsetzung zeigte, dass es unverzichtlich ist, die Schnittstelle zu den verbleibenden Teilprozessen zu schützen, um eine nahtlose Integration mit diesen zu ermöglichen. Es wurde begründet, dass die Durchführung einer Anforderungsanalyse wichtig ist, um ein ausführliches Pflichtenheft auszuarbeiten und es zeigte sich, dass regelmäßige Statusmeetings mit dem PO von Vorteil sind, um die pragmatische Korrektheit laufend zu überprüfen.

Durch diese Forschung wurde somit erwiesen, dass die Anmeldung und Zustellung von Weinen für Weinproben des Regionalverbunds für Weine in der Weinregion Mosel effizient und profitabel durch eine TYPO3-Erweiterung realisiert werden kann, indem für die technische Umsetzung “extreme-programming”-Entwicklungsmethodiken herangezogen werden, durch eine Anforderungsanalyse ein Pflichtenheft ausgearbeitet wird, der aktuelle Stand regelmäßig in Statusmeetings mit dem PO bespro-

chen wird, “chillerlan/php-qrcode” als QR-Code-Bibliothek verwendet wird und die Schnittstelle zu den verbleibenden Teilprozessen geschützt wird.

## 7.1. Ausblick

Nach der Umsetzung der Anmeldung und Zustellung von Weinen für Jahresauswahlproben, gilt es diese TYPO3-Erweiterung in ihrer Funktionalität zu erweitern, sodass der gesamte Geschäftsprozess der Jahresauswahlprobe abgebildet ist.

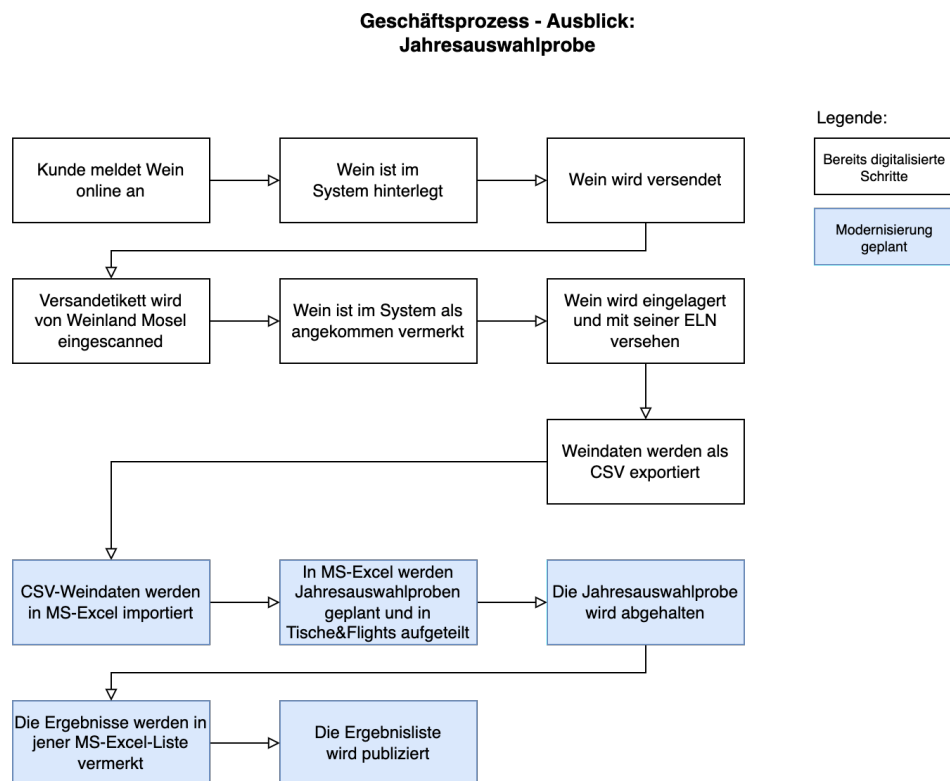


Abbildung 10.: Geschäftsprozess Jahresauswahlprobe: Ausblick, Quelle: Eigene Darstellung

Der vom Kunden langfristig erwünschte Zustand ist, dass der gesamte Geschäftsprozess der Jahresauswahlprobe teilautomatisiert wird. Das betrifft auch, aber nicht nur, das Aufteilen von Jahresauswahlproben



in Tische und Flights, das sichere Sammeln und Auswerten der Bewertungen und das Publizieren der daraus resultierenden Ergebnisse. Ein Flight ist eine weitere Unterteilung eines Tisches in der Zeit. So werden pro Tisch immer sechs Weine auf einmal probiert und bewertet. Die Bewertung eines Weines erfolgt derzeit über eine Bepunktung, die im Wertebereich  $[0, 100]$  auf Papier notiert wird. Anschließend wird mithilfe eines Taschenrechners das arithmetische Mittel der für diesen Wein vergebenen Punkte ausgerechnet. Das so zustandegekommene Ergebnis wird wiederum händisch in die Excel-Tabelle aus vorherigen Schritten übertragen. Ein solch umständliches Verfahren lenkt nicht nur von der eigentlichen Aufgabe der Bewertung ab, sondern stellt eine beachtliche Fehlerquelle dar. Auch ist es angedacht, die Weinanmeldung zu Jahresauswahlproben um komplexe Funktionalität zu erweitern, die sicherstellen soll, dass nicht gegen Restriktionen der jeweiligen Probenkategorien verstoßen werden kann. Somit soll vermieden werden, dass Weinanmeldungen aufgrund von Formfehlern abgelehnt werden. Ein triviales Beispiel hierfür wäre ein Wein mit zu hohem Restzuckeranteil, der in einer Jahresauswahlprobenkategorie für trockene Weine antreten soll. Ziel ist es, dass das Weinanmeldeformular fehlerhafte Anmeldeversuche als solche erkennt und verhindert. Kompliziert wird diese Technik dadurch, dass Kategorien beliebig viele Restriktionen haben können und diese von Redakteuren, im TYPO3-Backend, gepflegt werden sollen. Ein komplexeres, realistisches Beispiel wäre eine Kategorie, die nur vegane Barriqueweine der Rebsorte Merlot zulässt, die einen Mindest- und Maximalrestzuckeranteil, einen Maximalalkoholanteil und einen Maximalpreis erfüllen.

# Literaturverzeichnis

- [Abraham, 2005] Abraham, M. (2005). *Sustainability Science and Engineering: Defining Principles*. ISSN. Elsevier Science.
- [Ahmar, 2005] Ahmar, M. A. A. (2005). Rule based expert system for selecting software development methodology. *Journal of Theoretical and Applied Information Technology*.
- [Bettencourt, 2018] Bettencourt, R. G. (2018). barcode.php - Generate barcodes from a single PHP file. MIT license. <https://github.com/kreativekorp/barcode>. Zugriff: Januar 2023.
- [Day, 2020] Day, R. (2020). QR code/2d barcode generator. <https://github.com/soldair/node-qrcode>. Zugriff: Januar 2023.
- [Die SOPHISTen, 2016] Die SOPHISTen (2016). Die kleine RE-Fibel. 3.
- [Dougherty und Dunne, 2012] Dougherty, D. und Dunne, D. D. (2012). Digital Science and Knowledge Boundaries in Complex Innovation. *Organization Science*, 23(5):1467–1484.
- [Ebert, 2019] Ebert, C. (2019). Systematisches Requirements Engineering: Anforderungen ermitteln. 6.
- [Github, 2023] Github (2023). Saving repositories with stars. <https://docs.github.com/en/get-started/exploring-projects-on-github/saving-repositories-with-stars>. Zugriff: Januar 2023.
- [Heinemann, 2021] Heinemann, E. (2021). *Vorlesung: Requirements Engineering*. Hochschule Worms.
- [in2code, 2023] in2code (2023). femanager. <https://docs.typo3.org/p/in2code/femanager/11.5/en-us/Introduction/Index.html>. Zugriff: März 2023.

- [Jung, 2020a] Jung, L. (2020a). jquery-qrcode readme.md. <https://github.com/lrsjng/jquery-qrcode/blob/master/README.md>. Zugriff: Januar 2023.
- [Jung, 2020b] Jung, L. (2020b). kjua dynamically generated QR codes. <https://larsjung.de/kjua/>. Zugriff: Januar 2023.
- [König, 2022] König, W. (2022). *Vorlesung: User Experience*. Hochschule Worms.
- [Li et al., 2016] Li, F., Nucciarelli, A., Roden, S., und Graham, G. (2016). How smart cities transform operations models: a new research agenda for operations management in the digital economy. *Production Planning & Control*, 27(6):514–528.
- [Loebbecke und Picot, 2015] Loebbecke, C. und Picot, A. (2015). Reflections on societal and business model transformation arising from digitization and big data analytics: A research agenda. *The Journal of Strategic Information Systems*, 24(3):149–157.
- [Mpdf Contributors, 2023a] Mpdf Contributors (2023a). mPDF. <https://mpdf.github.io/>. Zugriff: März 2023.
- [Mpdf Contributors, 2023b] Mpdf Contributors (2023b). mpdf/mpdf PHP library generating PDF files from UTF-8 encoded HTML. <https://github.com/mpdf/mpdf>. Zugriff: März 2023.
- [Mpdf Contributors, 2023c] Mpdf Contributors (2023c). Overview mPDF Functions by Category. <https://mpdf.github.io/reference/mpdf-functions/overview.html>. Zugriff: März 2023.
- [npm, inc, 2020] npm, inc (2020). QR code/2d barcode generator. <https://www.npmjs.com/package/qrcode>. Zugriff: Januar 2023.
- [OpenJS Foundation, 2023] OpenJS Foundation (2023). Command-line API. <https://nodejs.org/api/cli.html#command-line-api>. Zugriff: Januar 2023.

- [Opensource.org, a] Opensource.org. The 2-Clause BSD License. <https://opensource.org/licenses/BSD-2-Clause>. Zugriff: Januar 2023.
- [Opensource.org, b] Opensource.org. The MIT License. <https://opensource.org/licenses/MIT>. Zugriff: Januar 2023.
- [Packagist, 2023] Packagist (2023). BaconQrCode is a QR code generator for PHP. <https://packagist.org/packages/bacon/bacon-qr-code>. Zugriff: Januar 2023.
- [Pagani und Pardo, 2017] Pagani, M. und Pardo, C. (2017). The impact of digital technology on relationships in a business network. *Industrial Marketing Management*, 67:185–192.
- [Parviainen et al., 2022] Parviainen, P., Tihinen, M., Kääriäinen, J., und Teppola, S. (2022). Tackling the digitalization challenge: how to benefit from digitalization in practice. *International Journal of Information Systems and Project Management*, 5(1):63–77.
- [Pixelant, 2023] Pixelant (2023). Tutorial on exporting Data from TYPO3. <https://www.typo3manual.com/typo3-basics/tips-and-functions/exporting-data-through-list-view>. Zugriff: März 2023.
- [Rochko, 2022a] Rochko, E. (2022a). A container class for immutable settings objects. Not a DI container. <https://github.com/chillerlan/php-settings-container>. Zugriff: Januar 2023.
- [Rochko, 2022b] Rochko, E. (2022b). A QR code generator and reader with a user friendly API. PHP 7.4+. <https://github.com/chillerlan/php-qrcode>. Zugriff: Januar 2023.
- [Rochko, 2022c] Rochko, E. (2022c). composer.json. <https://github.com/chillerlan/php-qrcode/blob/main/composer.json>. Zugriff: Januar 2023.

- [Rochko, 2023] Rochko, E. (2023). SVG QR Code Generator. <https://smiley.codes/qrcode>. Zugriff: Januar 2023.
- [Scholzen, 2018] Scholzen, B. (2018). barcode.php - Generate barcodes from a single PHP file. MIT license. <https://github.com/bacon/baconqrcode>. Zugriff: Januar 2023.
- [Schwarzer, 2021] Schwarzer, V. (2021). *Vorlesung: Software Architecture*. Hochschule Worms.
- [Schwarzer, 2022] Schwarzer, V. (2022). *Vorlesung: Algorithmen und Datenstrukturen*. Hochschule Worms.
- [Tabrizi et al., 2019] Tabrizi, B., Lam, E., Girard, K., und Irvin, V. (2019). Digital transformation is not about technology. *Harvard business review*, 13(March):1–6.
- [TeaCapps, 2023] TeaCapps (2023). QRBot | QRCode und Barcode Scanner. <https://qrbot.net>. Zugriff: März 2023.
- [The PHP Group, 2023a] The PHP Group (2023a). PHP: array-keys - Manual. <https://www.php.net/manual/en/function.array-keys.php>. Zugriff: März 2023.
- [The PHP Group, 2023b] The PHP Group (2023b). PHP: fputcsv Manual. <https://www.php.net/manual/de/function.fputcsv.php>. Zugriff: März 2023.
- [TYPO3 Association, 2023] TYPO3 Association (2023). Fluid - A fast and secure templating engine. <https://typo3.org/fluid>. Zugriff: März 2023.
- [TYPO3 Contributors, 2023a] TYPO3 Contributors (2023a). Configuration - Frontend Login 11.5 documentation. <https://docs.typo3.org/c/typo3/cms-felogin/11.5/en-us/Configuration/Index.html>. Zugriff: März 2023.

- [TYPO3 Contributors, 2023b] TYPO3 Contributors (2023b). Dependency injection - TYPO3 Explained 11.5 documentation. <https://docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/ApiOverview/DependencyInjection/Index.html>. Zugriff: März 2023.
- [TYPO3 Contributors, 2023c] TYPO3 Contributors (2023c). Extbase reference - TYPO3 Explained 11.5 documentation. <https://docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/ExtensionArchitecture/Extbase/Reference/Index.html>. Zugriff: Februar 2023.
- [TYPO3 Contributors, 2023d] TYPO3 Contributors (2023d). File structure - TYPO3 Explained 11.5 documentation. <https://docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/ExtensionArchitecture/FileStructure/Index.html>. Zugriff: Februar 2023.
- [TYPO3 Contributors, 2023e] TYPO3 Contributors (2023e). Form - Fluid ViewHelper Reference 11.5 documentation. <https://docs.typo3.org/other/typo3/view-helper-reference/11.5/en-us/typo3/fluid/latest/Form/Index.html>. Zugriff: März 2023.
- [TYPO3 Contributors, 2023f] TYPO3 Contributors (2023f). Introduction - TYPO3 Explained 11.5 documentation. <https://docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/ApiOverview/ContentElements/Introduction.html>. Zugriff: Februar 2023.
- [TYPO3 Contributors, 2023g] TYPO3 Contributors (2023g). Managing Extensions - Getting Started 11.5 documentation. <https://docs.typo3.org/m/typo3/tutorial-getting-started/11.5/en-us/Extensions/Management.html>. Zugriff: Februar 2023.
- [TYPO3 Contributors, 2023h] TYPO3 Contributors (2023h). Model - TYPO3 Explained 11.5 documentation. <https://docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/ApiOverview/Model/Index.html>. Zugriff: Februar 2023.

[//docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/ExtensionArchitecture/Extbase/Reference/Domain/Model.html](https://docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/ExtensionArchitecture/Extbase/Reference/Domain/Model.html). Zugriff: März 2023.

[TYPO3 Contributors, 2023i] TYPO3 Contributors (2023i). Repository - TYPO3 Explained 11.5 documentation. <https://docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/ExtensionArchitecture/Extbase/Index.html>. Zugriff: März 2023.

[TYPO3 Contributors, 2023j] TYPO3 Contributors (2023j). Repository — TYPO3 Explained 11.5 documentation. <https://docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/ExtensionArchitecture/Tutorials/Tea/Repository.html>. Zugriff: Februar 2023.

[TYPO3 Contributors, 2023k] TYPO3 Contributors (2023k). System categories - TYPO3 Explained 11.5 documentation. <https://docs.typo3.org/m/typo3/reference-coreapi/11.5/en-us/Api0verview/Categories/Index.html>. Zugriff: März 2023.

[TYPO3 Contributors, 2023l] TYPO3 Contributors (2023l). TYPO3 - Getting Started Tutorial - Getting Started 11.5 documentation. <https://docs.typo3.org/m/typo3/tutorial-getting-started/11.5/en-us/Index.html>. Zugriff: Februar 2023.

[TYPO3 Contributors, 2023m] TYPO3 Contributors (2023m). TYPO3 Concepts - Getting started 11.5 documentation. <https://docs.typo3.org/m/typo3/tutorial-getting-started/11.5/en-us/Concepts/Index.html>. Zugriff: Februar 2023.

[TYPO3 Contributors, 2023n] TYPO3 Contributors (2023n). TYPO3\CMS\Extbase\Domain\Model\FrontendUser Class Reference. [https://api.typo3.org/10.4/class\\_t\\_y\\_p\\_o3\\_1\\_1\\_c\\_m\\_s\\_1\\_1\\_extbase\\_1\\_1\\_domain\\_1\\_1\\_model\\_1\\_1\\_frontend\\_user.html](https://api.typo3.org/10.4/class_t_y_p_o3_1_1_c_m_s_1_1_extbase_1_1_domain_1_1_model_1_1_frontend_user.html). Zugriff: März 2023.

- [TYPO3 Contributors, 2023o] TYPO3 Contributors (2023o). TYPO3\CMS\Fluid\View\StandaloneView Class Reference. [https://api.typo3.org/10.4/class\\_t\\_y\\_p\\_o3\\_1\\_1\\_c\\_m\\_s\\_1\\_1\\_fluid\\_1\\_1\\_view\\_1\\_1\\_standalone\\_view.html](https://api.typo3.org/10.4/class_t_y_p_o3_1_1_c_m_s_1_1_fluid_1_1_view_1_1_standalone_view.html). Zugriff: März 2023.
- [Udjaja, 2018] Udjaja, Y. (2018). EKSPANPIXEL BLADSY STRANICA: Performance Efficiency Improvement of Making Front-End Website Using Computer Aided Software Engineering Tool. *Procedia Computer Science*, 135:292–301. The 3rd International Conference on Computer Science and Computational Intelligence (ICCS CI 2018) : Empowering Smart Technology in Digital Era for a Better Life.
- [Verhoef et al., 2021] Verhoef, P. C., Broekhuizen, T., Bart, Y., Bhattacharya, A., Qi Dong, J., Fabian, N., und Haenlein, M. (2021). Digital transformation: A multidisciplinary reflection and research agenda. *Journal of Business Research*, 122:889–901.
- [Vora, 2022] Vora, S. (2022). How to Create and Save text file in JavaScript? <https://www.tutorialspoint.com/how-to-create-and-save-text-file-in-javascript>. Zugriff: März 2023.
- [W3Schools, 2023a] W3Schools (2023a). HTML <input> Tag. [https://www.w3schools.com/tags/tag\\_input.asp](https://www.w3schools.com/tags/tag_input.asp). Zugriff: März 2023.
- [W3Schools, 2023b] W3Schools (2023b). Sass Introduction. [https://www.w3schools.com/sass/sass\\_intro.php](https://www.w3schools.com/sass/sass_intro.php). Zugriff: Januar 2023.
- [W3Schools, 2023c] W3Schools (2023c). SQL AUTO INCREMENT Field. [https://www.w3schools.com/sql/sql\\_autoincrement.asp](https://www.w3schools.com/sql/sql_autoincrement.asp). Zugriff: März 2023.
- [Zimmermann, 2017] Zimmermann, J. (2017). Webpack - A Detailed Introduction. <https://www.smashingmagazine.com/2017/02/a-detailed-introduction-to-webpack/>. Zugriff: Januar 2023.



# Anhang

# **A. Anmeldeforumular Jahresauswahlprobe**

## Kategorien

Nr.	Bezeichnung	Unterkategorien/Kriterien
1.	2021 PiWi-Weißweine trocken	Rebsortenreine Weine sowie Rebsorten-Cuvées
2.	2021 Elbling Qualitätswein trocken	
3.	2021 Rivaner/Müller-Thurgau Qualitätswein trocken	
4.	2021 Weiße Burgundersorten Qualitätswein trocken	Zugelassen sind: Weißer Burgunder, Grauer Burgunder, Chardonnay und Auxerrois oder Cuvées daraus
5.	2021 Riesling Qualitätswein trocken - Gutswein	
6.	2021 Riesling Qualitätswein trocken - Ortswein	Auch trockene Kabinettweine
7.	2021 Riesling Qualitätswein feinherb - Gutswein	
8.	2021 Riesling Qualitätswein feinherb - Ortswein	
9.	2021 Riesling Kabinett feinherb (bis max. 11,5 Vol.-Prozent)	Bis 11,5 Volumen-Prozent vorhandener Alkoholgehalt
10.	2021 Riesling Kabinett lieblich/fruchtsüß (bis max. 70gr/Ltr. Restzucker)	Bis maximal 70 Gramm Restzuckergehalt je Liter
11.	2021 Spätburgunder Rosé Qualitätswein trocken	
12.	Sekt/Crémant (Sekt ausschließlich b.A. bzw. Crémant Mosel und aus traditioneller Flaschengärung, mindestens neun Monate Hefelager)	12.1 Elbling zero dosage bis brut 12.2. Riesling zero dosage bis trocken 12.3. Burgundersorten zero dosage bis brut

## Anmeldeformular

Zu jeder Probe nachstehende Angaben ausfüllen und den 2 Probeflaschen (auch bei Schraubverschluss) beifügen!

<b>Kategorie Nr.</b>	
----------------------	--

**Nur vollständig ausgefüllt gültig**

Jahrgang:		Rebsorte:			
Weinbezeichnung (z.B. Lagename und/oder andere Bezeichnungen wie Steillage, Alte Reben, Schiefer etc.)					
Qualitätsstufe:					
Geschmacksangabe:					
Biowein (Zutreffendes bitte ankreuzen)			veganer Wein (Zutreffendes bitte ankreuzen)		
Spontangärung (Zutreffendes bitte ankreuzen)			Barrique (Zutreffendes bitte ankreuzen)		
Sonstige Angaben zu Besonderheiten:					
AP-Nr. (sofern bereits erteilt):					
Säure (g/l):		Restzucker (g/l):		Alkoholgehalt (% Vol.):	
			Endverbraucherpreis (inkl. MwSt./Sektsteuer):		

### ***EINREICHER:***

Betriebsname:	
Ansprechpartner (Vorname/Name):	
Straße:	PLZ/Ort:
Tel:	E-Mail:
Internet:	
Ich erkläre mich hiermit mit den Bedingungen der Ausschreibung einverstanden. Den Datenschutzhinweis habe ich zur Kenntnis genommen.	
Datum und Unterschrift:	

## B. Interview-Fragebogen

Wie stellen Sie sich den Prozess des Einscannens der QR-Codes beim Entgegennehmen der Flaschen vor? Beschreiben Sie den Ablauf.

Mit was soll dieser Code gescannt werden? Soll der Scanner in der Applikation eingebaut sein, oder soll das System auch mit Drittanbieter-Apps funktionieren?

Welche Fallbacks soll es geben, sollte ein Code nicht scanbar sein? Z.B.: Der Code-Inhalt in Text unter dem Code, der auch von Hand eintippbar sei.

Gegen welche Missbrauchsszenarien sollte der QR-Code geschützt sein? Sollte ggf. ein Passwort nach dem Einscannen verlangt werden? -> Diskurs über versch. Authentifizierungsmethoden und Abwägung über Aufwand der dadurch entsteht.

Was soll der Kunde beim Versand selbst machen?

Wenn es ein PDF zum Ausdruck zum Aufbringen des Empfängers gibt, sollte man auch ein PDF zum Ausdruck des Absenders generieren, falls die relevanten Daten vorliegen?

In welcher Form sollten Mitarbeiter die ausstehenden und eingegangenen Weine sehen? Reicht eine einfache Liste, oder sind Export- und Filtermöglichkeiten erwünscht? Wenn ja: Welche Filter (auch Sortierungen)? Welche Exportformate (Excel kann auch csv öffnen)?

Wenn nun ein Wein als “eingegangen” vermerkt ist, sollte ein Mitarbeiter das rückgängig machen können? Sollte ein Mitarbeiter Weine löschen können? Wenn eines der beiden ja: Einzeln, oder als Bulk-Action? (Bulk-Actions sind teuer/aufwändig umzusetzen)

Welche Informationen soll der Kunde über seine Sendunge(n) sehen?

Soll auch ein Kunde in der Lage sein, seine eigene Weinsendung(en) aus dem System zu löschen oder zu verändern? (Eventuell vertippt man sich) Wenn ja, bis wann? Bis zur Eingangsbestätigung? Ist hierbei eine Bulk-Action wichtig?

Die ELN ist eine inkrementell erhöhte Ganzzahl. Ist der tatsächliche Wert dieser wichtig, oder ist es lediglich wichtig, dass sie eindeutig ist? Der einfachste/günstigste Weg wäre es, sie in der Datenbank als “auto\_increment” zu deklarieren. Dann hätte man niemals, auch über x Auswahlproben hinweggehend, die selbe ELN zwei mal. Das verkürzt und vereinfacht die Entwicklung, den entstehenden Code, und die Nutzererfahrung auf Seiten von WM.

## C. Online-Fragebogen Weingüter

# Jahresauswahlprobe - Registrierungstool

1. Sind Sie mit der aktuellen Arbeitsweise der Anmeldung ihrer Weine zur Jahresauswahlprobe zufrieden?

*Markieren Sie nur ein Oval.*

- ☐ Ja  
☐ Nein

2. Wünschen Sie sich eine Digitalisierung dieser Arbeitsweise, die ihre Formulare verkürzt, an den Rechner verschiebt und es ihnen ermöglicht zu verfolgen, ob ihre Sendungen eingegangen sind?

*Markieren Sie nur ein Oval.*

- ☐ Ja  
☐ Nein

3. Wie viele Weine stellen Sie pro Jahresauswahlprobe ungefähr auf?

---



4. Mit welchem Gerät greifen Sie primär auf das Internet zu?

*Wählen Sie alle zutreffenden Antworten aus.*

- ☐ iPhone
- ☐ Android Phone
- ☐ iPad
- ☐ Android Tablet
- ☐ Laptop/Macbook
- ☐ Desktop-Rechner

5. Haben Sie Zugriff auf einen Drucker?

*Markieren Sie nur ein Oval.*

- ☐ Ja
- ☐ Nein

6. Haben Sie sonstige Wünsche und Anregungen bezüglich der Anmeldung zu Jahresauswahlproben?

---

---

---

---

---

---

Dieser Inhalt wurde nicht von Google erstellt und wird von Google auch nicht unterstützt.

Google

## D. Interview-Protokoll

Interviewer	Leon Etienne (Entwickler)
Befragter	Jochen Stange (Product Owner)
Zeit	27.01.2023, 10:00
Ort	Zoom
Notizen	Dieses Protokoll ist nicht wortlautkorrekt, sondern zeigt lediglich zeitgleich verfasste Notizen

**Wie stellen Sie sich den Prozess des Einscannens der QR-Codes beim Entgegennehmen der Flaschen vor? Beschreiben Sie den Ablauf.** - Paket kommt an - Wird aufgemacht - Nehmen den QR-Code entgegen - Haben entweder einen Handscanner oder ein Handy und scannen den Code ein - System gibt an, welche ELN aufgeklebt werden muss - Wein wird eingelagert

Die ELN muss unique sein. Die ELN könnte schlicht die von TYPO3 vergebene UID sein.

**Mit was soll dieser Code gescannt werden? Soll der Scanner in der Applikation eingebaut sein, oder soll das System auch mit Drittanbieter-Apps funktionieren?** Smartphone ist am sinnvollsten, da Feedback von Webseite damit möglich ist. Backup-Funktion, die ELN händisch einzugeben. Das ginge dann auch mit einem Handscanner.

**Welche Fallbacks soll es geben, sollte ein Code nicht scanbar sein? Z.B.: Den Code-Inhalt in Text unter dem Code, der auch von Hand eintippbar sei.** Die ELN muss auch unter dem QR-Code stehen.

**Gegen welche Missbrauchsszenarien sollte der QR-Code geschützt sein? Sollte ggf. ein Passwort nach dem Einscannen verlangt werden? ->Diskurs über versch. Authentifizierungsmethoden und Abwägung über Aufwand**

**der dadurch entsteht.** Mitarbeiter authentifizieren sich mit einem Frontend-Nutzer. Dieser Nutzer ist Teil einer Mitarbeiter-Nutzergruppe.

**Gibt es im Lager Wlan?** Out of scope.

**Wie wollen Sie Weinanmeldungen zwischen Jahresauswahlproben im Frontend trennen? Wichtig: Aktuelle Proben nach Anmeldeschluss sollten immer noch sichtbar sein und alte Proben ggf. gar nicht mehr angezeigt werden.** Es gibt eine Liste mit Jahresauswahlproben. Da werden verschiedene aufgelistet. Backend-User können die Anmeldung (Weine an Jahresauswahlproben anmelden) an- und abschalten. Vergangene Proben werden nicht angezeigt. Die Probe gilt als vergangen, wenn die Proben probiert wurden. Die Anmelde-möglichkeit und “Probe ist vorbei”-Feststellung im Anmeldetool ist ein Clone vom „active until“ im „access-„panel. Mit Dateselector.

**Was soll der Kunde beim Versand selbst machen?** Wenn Weinland-Mosel-Mitglied: Ggf Account erstellen mit Mitgliedsnummer <schon implementiert> <muss händisch von WM freigeschalten werden>. In Account einloggen.

Wenn Nichtmitglied: Ggf. Account erstellen ohne Mitgliedsnummer <muss händisch freigeschalten werden>

Eine Jahresauswahlprobe auswählen. Einen Wein anmelden, Webform zu Wein-daten ausfüllen. PDF downloaden, ausdrucken im Paket dem Wein beilegen. Das PDF muss Daten zum Wein beinhalten und den QR-Code mit der ELN. (zur Zuordnung) Nummernschwund, falls Weine gelöscht werden, ist kein Problem.

Adressierung wird selbst gemacht. Frankierung auch.

**In welcher Form sollten Mitarbeiter die ausstehenden und eingegangenen Weine sehen? Reicht eine einfache Liste, oder sind Export- und Filtermöglichkeiten erwünscht? Wenn ja: Welche Filter (auch Sortierungen)? Welche Exportformate?** - CSV soll ausgebbbar sein. - Filter nicht notwendig, aber probenspezifisch.

**Wenn nun ein Wein als “eingegangen” vermerkt ist, sollte ein Mitarbeiter das rückgängig machen können? Sollte ein Mitarbeiter Weine löschen können? Wenn eines der beiden ja: Einzeln, oder als Bulk-Action? (Bulk-Actions sind teuer/aufwändig umzusetzen) - Was TYPO3 by-default kann, nichts eigenes bauen.**

**Welche Informationen soll der Kunde über seine Sendunge(n) sehen?**  
- Ob sie eingegangen ist, mit Emailbenachrichtigung.

**Soll auch ein Kunde in der Lage sein, seine eigene Weinsendung(en) aus dem System zu löschen oder zu verändern? (Eventuell vertippt man sich) Wenn ja, bis wann? Bis zur Eingangsbestätigung?** Nach Einreichung (nicht Eingang bei WM) nicht mehr veränderbar und auch nicht löschar.

**Die ELN ist eine inkrementell erhöhte Ganzzahl. Ist der tatsächliche Wert dieser wichtig, oder ist es lediglich wichtig, dass sie eindeutig ist? Der einfachste/günstigste Weg wäre es, sie in der Datenbank als “auto\_increment” zu deklarieren. Dann hätte man niemals, auch über x Auswahlproben hinweggehend, die selbe ELN zwei mal. Das verkürzt und vereinfacht die Entwicklung, den entstehenden Code und die Nutzererfahrung auf Seiten von WM. - Siehe oben.**

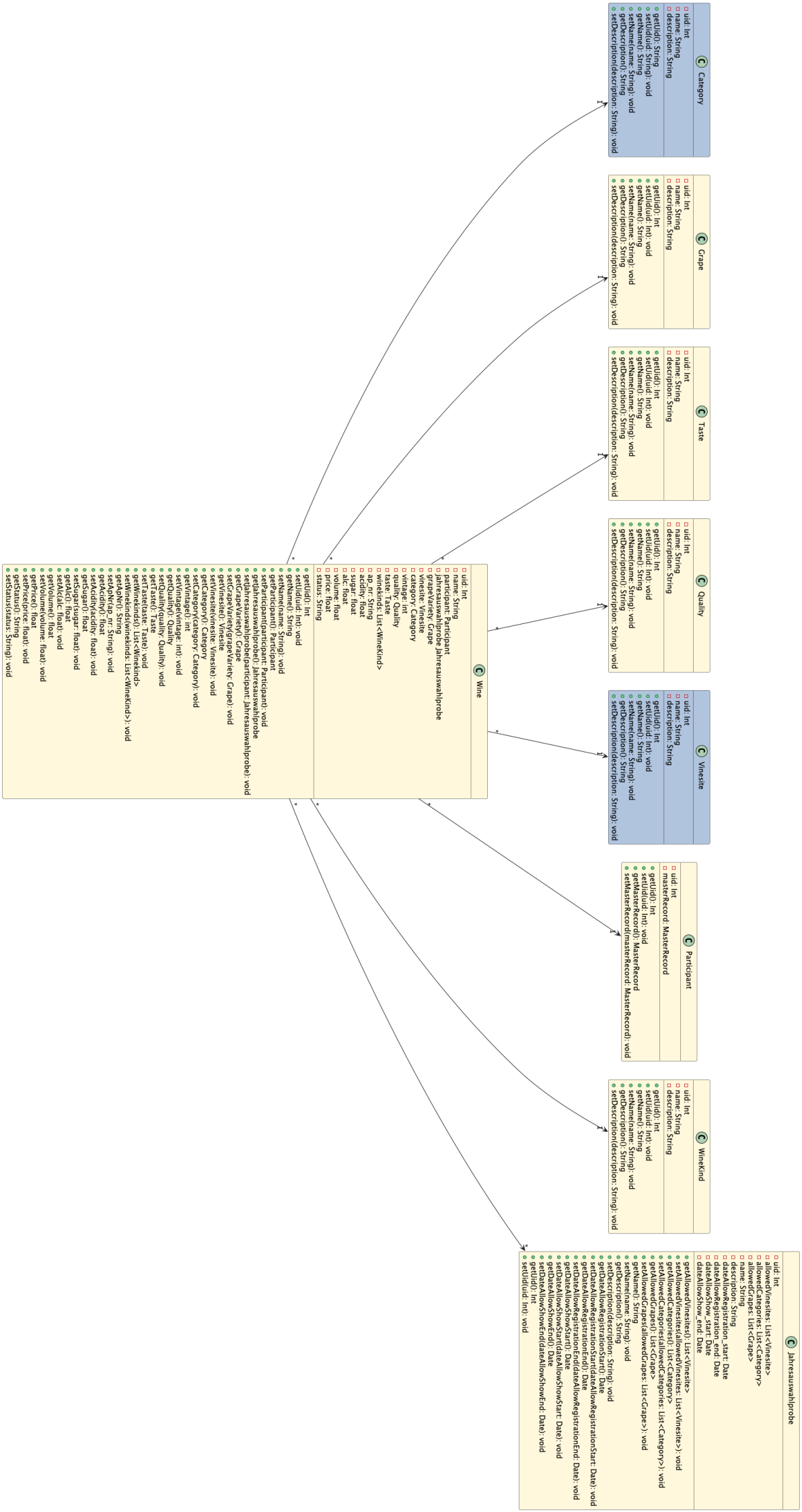
**Weiteres** Es soll einen Single-View für Weine geben, der anhand einer ELN aufrufbar ist. Hier werden bekannte Details gezeigt.

**Zum Formular, das Winzer ausfüllen:** Das Formular besteht zwei Schritten: Schritt 1: Daten ausfüllen. Schritt 2: Zusammenfassung anzeigen entweder zurück, weiter bearbeiten, oder submit “Wein verbindlich anmelden”.

## E. Pflichtenheft

Art der Anforderung	Anforderungsbeschreibung
FA	Angemeldete Nutzer können mehrere Weine bei aktiven Jahresauswahlproben anmelden.
FA	Angemeldete Nutzer sehen angemeldete Weine für Jahresauswahlproben inklusive Versandstatus.
FA	Angemeldete Nutzer sehen eine Übersicht aller aktiven Jahresauswahlproben.
FA	Das Formular lässt sich mit einem "Jetzt verbindlich anmelden"-Button abschicken.
NFA	Das Formular mit Weindaten und dem QR-Code lässt sich auch direkt nach Anmeldung eines Weines herunterladen.
Randbedingung	Das Kunden-UI soll im Frontend sein.
NFA	Der API-Link der QR-Codes ist mittels der TYPO3-Frontend-Nutzer-Authentifizierung gesichert.
Randbedingung	Der Einbau in einem Brown-Field TYPO3 statt.
NFA	Der Wein-Single-View zeigt bekannte Weindaten.
NFA	Die ELN wird beim Einscannen von QR-Codes Url-kodiert übergeben.
FA	Eine Weinanmeldung fragt Weindaten mit einem Webform ab.
FA	Es gibt einen Single-View für Weine, der über eine ELN in der URL erreichbar ist.
FA	Für angemeldete Weine lassen sich PDF-Dateien herunterladen, die die als Barcode und eine Beschreibung des Weines enthalten.
FA	Mitarbeiter von WM können einen gesicherten API-Link anhand einer ELN aufrufen um einen Wein als "eingegangen" zu markieren
FA	Mitarbeiter von WM können eingetragene Weine als TYPO3-Objekte manipulieren.
FA	Mitarbeiter von WM können Jahresauswahlproben im Backend zeitgesteuert für Anmeldungen freigeben.
FA	Mitarbeiter von WM können Jahresauswahlproben im Backend zeitgesteuert für die Einsicht freigeben
FA	Mitarbeiter von WM können Weineintragungen je nach Jahresauswahlprobe als CSV exportieren.
FA	Mitglieder bei WM können Nutzerkonten erstellen.
NFA	Nach Aufruf eines API-Links wird gezeigt, welcher Wein gescannt wurde.
FA	Nicht-Mitglieder bei WM können Nutzerkonten erstellen.
FA	Vor Abschluss des Formulars wird eine Übersicht der Daten gezeigt.
FA	Wird ein Wein via dem QR-Code als "eingegangen" markiert, bekommt das Weingut eine Email.

## F. Klassendiagramm





## G. Notizen: Digitization des Weinanmelde-Formulares

Moselwein e.V., Gartenfeldstr. 12 a, 54295 Trier  
Jahresauswahlprobe „Sommer 2022“

MOSEL

FASZINATION WEIN

## Anmeldeformular

Zu jeder Probe nachstehende Angaben ausfüllen und den 2 Probeflaschen (auch bei Schraubverschluss) beifügen!

Kategorie Nr.	Kategorie je Probe			Nur vollständig ausgefüllt gültig		
Jahrgang:	Eigentlich / FE-Markt			Rebsorte:		
Weinbezeichnung (z.B. Lagename und/oder andere Bezeichnungen wie Steillage, Alte Reben, Schiefer etc.)	select the be-records with records be					
Qualitätsstufe:	select nach be-records strings					
Geschmacksangabe:	select be-records strings					
Biowein (Zutreffendes bitte ankreuzen)			veganer Wein (Zutreffendes bitte ankreuzen)			
Spontangärung (Zutreffendes bitte ankreuzen)	check boxes		Barrique (Zutreffendes bitte ankreuzen)			
Sonstige Angaben zu Besonderheiten:	Freitext					
AP-Nr. (sofern bereits erteilt):	integer					
Säure (g/l):		Restzucker (g/l):		Alkoholgehalt (% Vol.):		
				Endverbraucherpreis (inkl. MwSt./Sektsteuer):	+ Flaschengröße in l	

EINREICHER:

Freitext de zimal

In

Liteo

Betriebsname:	
Ansprechpartner (Vorname/Name):	
Straße:	PLZ/Ort:
Tel:	E-Mail:
Internet:	
Ich erkläre mich hiermit mit den Bedingungen der Ausschreibung einverstanden. Den Datenschutzhinweis habe ich zur Kenntnis genommen.	
Datum und Unterschrift:	