

Hochschule Worms  
Fachbereich Informatik  
Studiengang Angewandte Informatik - dual (B.Sc)

Abschlussarbeit

# Bachelorarbeit

Digitalisierung der Anmeldung und Zustellung von Weinen  
zu Weinproben des Regionalverbands für Weine in der  
Weinregion Mosel als TYPO3-Extension

Version 1.0

Vorgelegt von

Leon Etienne, 676838  
inf3642@hs-worms.de  
mit Medienagenten Stange & Ziegler oHG  
Im Sommersemester 2023  
Bad Dürkheim

bei Volker Schwarzer  
schwarzer@hs-worms.de  
Worms

# 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, 22. Dezember 2022

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>3</b>
2.1. Modell nach Parviainen et al . . . . .	3
2.2. Modell nach Verhoef et al . . . . .	6
<b>3. Stand der Technik</b>	<b>7</b>
3.1. Die bestehende Webseite . . . . .	7
3.2. QR-Codes erstellen . . . . .	7
<b>4. Anforderungserfassung</b>	<b>11</b>
<b>5. Konzeption und Entwicklung</b>	<b>12</b>
<b>6. Ergebnisse</b>	<b>13</b>
<b>7. Ausblick</b>	<b>14</b>
<b>Literaturverzeichnis</b>	<b>15</b>
<b>Anhang</b>	<b>18</b>
<b>A. Interview-Fragebogen</b>	<b>19</b>

---

# Abbildungsverzeichnis

2.1. Model for tracking digital transformation . . . . .	4
2.2. Plan-Do-Check-Act (PDCA) . . . . .	5

# Tabellenverzeichnis

4.1. Initial bekannte Anforderungen . . . . .	11
---	----

# Abkürzungsverzeichnis

**WM** “Weinland Mosel” e.V.

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

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

**FA** Funktionale Anforderung

**NFA** Nichtfunktionale Anforderung

# Glossar

## **Frontend**

Das Frontend einer Webseite ist der Teil der Webseite, der Endnutzern präsentiert wird. Ein Frontend besteht aus HTML und Cascading Style Sheets [Udjaja, 2018].

# 1. Einleitung

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

## 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. Diese Daten sind redundant, da sie keine Eigenschaften der Weine, sondern die des Weingutes selbst sind. Da sich das Weingut zwischen den Weinen nicht ändert, ändern sich die darauf bezogenen Daten auch nicht. Sie müssen aber für jeden Wein erneut ausgefüllt werden. Abgesehen davon, dass solche Redundanzen auf Weinguts- und Verbandsseite die hedonische Qualität schädigen, bietet so ein Workflow Freiraum für Fehler und Inkonsistenzen. Dieser Workflow mit den zuvor genannten Nachteilen wird auf Verbandsseiten, nach Zustellung der Weine, weiter fortgeführt: WM erfährt erstmalig mit Zustellung eines Weines von dessen Teilnahme. Das erschwert das Planen der Logistik, da im Voraus keine konkrete Zahl 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 Lücke 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, weil es oft vorkommt, dass die zuteils dysgraphisch verfassten Formulare nur schwer, mehrdeutig, oder gar nicht dechiffriert werden können. In diesem Prozess wird der Teilnahme des Weines eine inkrementell aufsteigende Einlieferungsnummer zugewiesen, die anschließend in Form eines Aufklebers an der Flasche befestigt wird. Anschließend wird die Flasche im Lager verstaut.

### 1.2. Zielsetzung

Ziel dieser Arbeit ist es, in Erfahrung zu bringen, wie der zuvor genannte Prozess bestmöglichst, im Rahmen bestimmter Constraints und funktionalen- sowie nichtfunktionalen Anforderungen, digitalisiert werden kann. Während die Constraints bereits bekannt sind, werden detaillierte Anforderungen im Rahmen der Anforderungstechnik ausgearbeitet [Heinemann, 2021]. Die Constraints 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 System zugehörig sind, müssen im Frontend der Webseite stattfinden. Oberflächen für Mitarbeiter von WM dürfen in der TYPO3-Backend-Oberfläche implementiert werden.

Somit lautet die **Forschungsfrage**:

*Was ist die beste Möglichkeit, um die Anmeldung und Zustellung von Weinen zu Weinproben des Regionalverbands für Weine in der Weinregion Mosel als TYPO3-Erweiterung zu realisieren, um davon zu profitieren?*

## 2. Stand der Forschung

Der Stand der Forschung beleuchtet verschiedene Erkenntnisse zur Digitalisierung und zur Digitalen Transformation.

### 2.1. Modell nach Parviainen et al

*“The importance of digitalization is becoming understood, but the question now is how to do it in practice in order to best benefit from it.”*  
[Parviainen et al., 2022].

Parviainen et al. stellten sich diese Frage 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 drastische Auswirkungen zu verhindern oder dras-

tische 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 werden. Relevant sind hier die Vorteile der Umsetzung des Szenarios, die Kosten der Umsetzung sowie die Risiken, das Szenario nicht umzusetzen oder doch umzusetzen. 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.

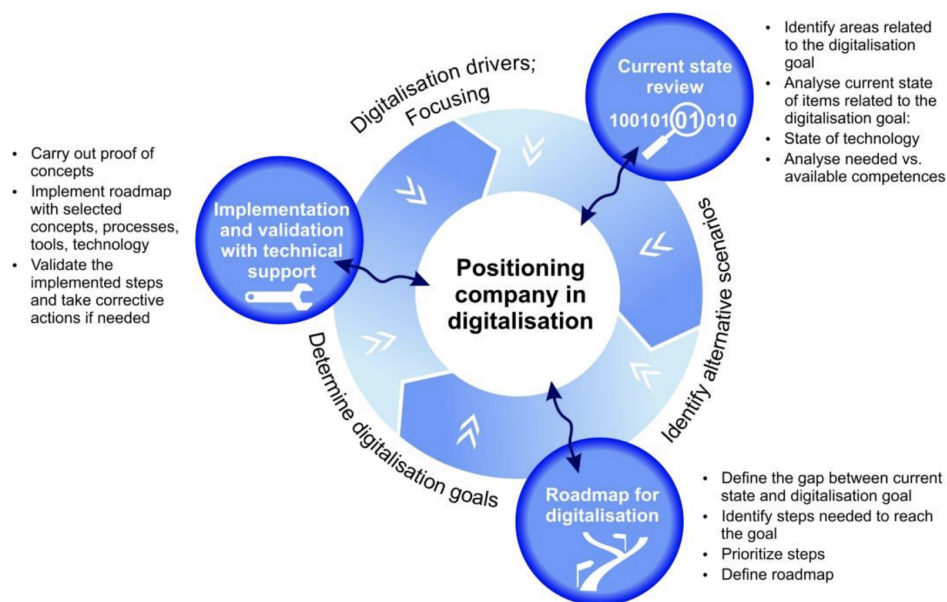


Abbildung 2.1.: Model for tracking digital transformation

Quelle: [Parviainen et al., 2022]

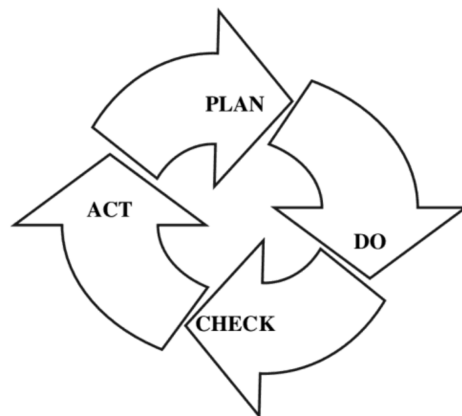


Abbildung 2.2.: Plan-Do-Check-Act (PDCA)

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.

**Der dritter Schritt** ist die Festlegung der konkreten Schritte, die für den Übergang vom Ist-Zustand zum 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** Der vierte Schritt befasst sich mit der Umsetzung der in Schritt 3 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 bestimmte 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 so genannte “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, so dass diese digital, z.B. in Form von Nullen und Einsen, gespeichert und elektronisch weiterverarbeitet werden können [Dougherty and Dunne, 2012, Loebbecke and Picot, 2015].

*Digitalisierung* beschreibt den Prozess der Veränderung bestehender Geschäftsprozesse, um mit digitalen Werkzeugen und Datenmodellen zu arbeiten [Li et al., 2016]. Die letzte Phase, die *Digitale Transformation* beschreibt eine firmenweite Veränderung, die beispielsweise Ergründungen neuer Geschäftsmodelle mit sich bringen könnte [Pagani and Pardo, 2017].

## 3. Stand der Technik

Der Stand der Technik bezieht sich auf bestehende praktische Umsetzungen der erforderlichen Technologien. Im Wesentlichen gibt es zwei Arten von Technologien, die untersucht werden müssen: Die bestehende Website von WM und eine PHP-Bibliothek zur Erzeugung von QR-Codes.

### 3.1. Die bestehende Webseite

Als Mitentwickler des Projektes ist dem Author dieser Ausarbeitung bekannt, dass die bestehende Webseite von WM 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, 2023].

### 3.2. QR-Codes erstellen

Um mit QR-Codes zu arbeiten, ist es unabdinglich, QR-Codes zu erstellen, da dieselben sonst nicht vorhanden sind. Prinzipiell gibt es zwei Möglichkeiten QR-Codes zu erstellen: Auf Browserseite in JavaScript und auf Serverseite in PHP. Das ist so, da es lediglich diese zwei Domänen gibt.

#### Javascript-Implementationen

##### **jquery-qrcode**

*jquery-qrcode* 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* für nähere Evaluationen aus, da die Software nicht mehr gepflegt wird. *jquery-qrcode* wurde bis zum heutigen Tage 853 mal von Nutzern mit einem Stern markiert

[Jung, 2020a]. Wenn ein Nutzer ein Github-Repository mit einem Stern markiert, interessiert sich der Nutzer weit genug, um über Neuigkeiten auf dem Laufenden gehalten werden zu wollen [Github, 2023].

#### **kjua**

*Kjua* ist eine Javascript-Bibliothek, um dynamisch QR-Codes auf Browserseite zu generieren. Im Gegensatz zu *jquery-qrcode* funktioniert *Kjua* auch ohne JQuery. Es werden diverse Stilattribute für gestaltete QR-Codes unterstützt [Jung, 2020b]. *Kjua* setzt QR-Codes über HTML-Canvas um. Das ist bei näherer Betrachtung der *Kjua* Tech Demo (<https://larsjung.de/kjua/latest/demo>) ersichtlich, jedoch nicht explizit erwähnt. Nach dem aktuellen Stand wurde *Kjua* von 243 Nutzern mit einem Stern markiert [Jung, 2020b].

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

*Soldair/node-qrcode* ist eine node.js-basierte Implementation eines QR-Code Generators und bietet somit Funktionalität serverseitig, als CLI, sowohl als auch Browserseitig an. Die Readme-Datei zeugt von Länge, ist reich an Beispielen und detailreichen Erklärungen. Der letzte Commit ist zu diesem Zeitpunkt knapp älter als ein halbes Jahr. Somit macht das Projekt einen moderat gepflegten Eindruck. Die Readme-Datei verweist auf Unit Tests bei Travis, jedoch lief die letzte Pipeline vor circa zwei Jahren, Februar 2021, durch und schlug zudem fehl. Einige Pull-Requests und Issues werden seit Jahren ignoriert [Day, 2020]. Die Bibliothek wurde 74 Millionen mal heruntergeladen mit 6308 Sternen markiert [npm, inc, 2020].

## **PHP-Implementationen**

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

Eine arrivierte Lösung für QR-Code Generierung in PHP ist *chillerlan/php-qrcode*, umgesetzt durch Eugen Rochko und weitere. Das zeigt sich durch den zu diesem Zeitpunkt auf 5 Millionen stehenden Downloadzähler,

mehrere Entwickler und den letzten Commit, der zu diesem Zeitpunkt nicht älter als einen Monat alt ist. Insgesamt erfolgten bis dato 808 Commits von 6 Entwicklern. Das Projekt verfügt über Unit-Tests, die 90% der Zeilen in der Codebase abdecken. Rochko übernahm Teile der Codebase aus dem Java-Projekt “ZXing” und übersetzte diese zu PHP. Issues und Pull Requests sind alle bearbeitet. Es gibt 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 (1) Emoji bestehen. *chillerlan/php-qrcode* wurde von 1212 Github-Nutzern mit einem Stern markiert [Rochko, 2022].

#### **kreativekorp/barcode**

*kreativekorp/barcode* ist eine PHP-Bibliothek zur Generierung von QR-Codes, bereitgestellt von *Kreative Software*, R.G. Bettencourt. Diese Implementation umfasst eine Vielzahl an Barcode-Formaten und unterstützt eine Vielzahl an Anpassungsmöglichkeiten. Das Projekt wurde bis zum heutigen Tage 189 mal mit einem Stern markiert [Bettencourt, 2018]. Die letzte Codeänderung dieses Projektes ist auf 2018 datiert [Bettencourt, 2018]. Das liegt zu diesem Zeitpunkt vier Jahre in der Vergangenheit. Vier von den sechs 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. Damit ist diese Bibliothek de-facto sechseinhalb Jahre alt und wurde seitdem ein mal um Features erweitert [Bettencourt, 2018].

### **Vergleich in Bezug auf die Problemstellung**

Um eine Bibliothek als “die Beste” für einen Anwendungsfall zu kurieren, müssen die konkreten Anforderungen und Constraints für diesen Anwendungsfall beachtet werden. Das ist so, da verschiedene Eigenschaften der Bibliotheken verschiedene Auswirkung 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 für diese Problemstellung relevanten Funktionen in Annahme dessen, dass die Bibliothek syntaktisch und pragmatisch korrekt [Heinemann, 2021] funktioniert.

#### **Gepflegtheit**

Das Ausmaß, in dem das Projekt aktiv gepflegt 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, mehr als nur ein Contributor, Anzahl der Sterne auf Github (Ausmaß an tieferem, öffentlichem Interesse [Github, 2023]), sowie der Anzahl der Downloads bzw Installationen, falls verfügbar.

#### **Workflow-Eignung**

Die Eignung einer Bibliothek in existierende Workflows und Constraints übernommen zu werden.

## 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]. Die Anforderungen eines Produktes sind in drei Kategorien einzuteilen: Funktionale Anforderungen (FAs), Nichtfunktionale Anforderungen (NFAs), und Constraints [Heinemann, 2021]. Wie oben erwähnt, sind bereits die Constraints und einige funktionale Anforderungen bekannt. Diese sind:

Art	Beschreibung
Constraint	Einbau in brown-field TYPO3
Constraint	Mitarbeiter-UI in TYPO3-Backend
Constraint	Kunden-UI im Frontend
FA	Aus dem Papierformular soll ein Webform werden
FA	Angaben zum Weingut des Weines sollen aus dem Accountdatensatz anstatt aus dem Webform kommen
FA	Registrierte Weinteilnahmen bestehen aus einem Wein und einem Zustand(Ausstehend,Eingegangen).
FA	Beim erstellen einer Weinteilnahme soll ein QR-Code als PDF generiert werden, der den Wein identifiziert
FA	Durch scannen des QR-Codes soll dem Wein der Status “Eingegangen” zugewiesen werden

Tabelle 4.1.: Initial bekannte Anforderungen

# 5. Konzeption und Entwicklung

## 6. Ergebnisse

## 7. Ausblick

# Literaturverzeichnis

- [Abraham, 2005] Abraham, M. (2005). *Sustainability Science and Engineering: Defining Principles*. ISSN. Elsevier Science.
- [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.
- [Dougherty and Dunne, 2012] Dougherty, D. and Dunne, D. D. (2012). Digital science and knowledge boundaries in complex innovation. *Organization Science*, 23(5):1467–1484.
- [Ebert, 2019] Ebert, C. (2019). Vorwort zu systematisches re.
- [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.
- [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.
- [Li et al., 2016] Li, F., Nucciarelli, A., Roden, S., and 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 and Picot, 2015] Loebbecke, C. and 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.
- [npm, inc, 2020] npm, inc (2020). QR code/2d barcode generator. <https://www.npmjs.com/package/qrcode>. Zugriff: Januar 2023.
- [Pagani and Pardo, 2017] Pagani, M. and 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., and 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.
- [Rochko, 2022] Rochko, E. (2022). A QR code generator and reader with a user friendly API. PHP 7.4+. <https://github.com/chillerlan/php-qrcode>. Zugriff: Januar 2023.
- [Tabrizi et al., 2019] Tabrizi, B., Lam, E., Girard, K., and Irvin, V. (2019). Digital transformation is not about technology. *Harvard business review*, 13(March):1–6.
- [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 (ICCSCI 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., and Haenlein, M. (2021). Digital transformation: A multidisciplinary reflection and research agenda. *Journal of Business Research*, 122:889–901.

[W3Schools, 2023] W3Schools (2023). Sass Introduction. [https://www.w3schools.com/sass/sass\\_intro.php](https://www.w3schools.com/sass/sass_intro.php). Zugriff: Januar 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. 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?

Von welchem Endgerät wird gescannt?

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

Wie viel soll der Kunde beim Versand selbst machen? Muss er es selbst mit einem Empfänger versehen, oder gibt es diesen als PDF zum Ausdruck zum aufkleben?

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 "eingetragen" 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 Einlieferungsnummer 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 Einlieferungsnummer zwei mal. Das verkürzt und vereinfacht die Entwicklung, den entstehenden Code, und die Nutzererfahrung auf Seiten von Weinland Mosel.