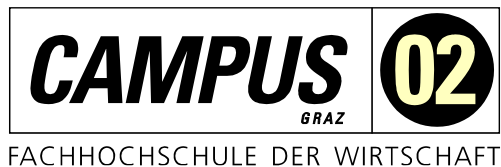


# MASTERARBEIT

## KONZEPTION UND ENTWICKLUNG EINES WEBBASIERTEN PORTALS ZUR HYPOTHEKARKREDITVERMITTLUNG

ausgeführt am



Studiengang

Informationstechnologien und Wirtschaftsinformatik

Von: Marco Schweiger

Personenkennzeichen: 2010320035

Graz, am 15. Dezember 2021

.....  
Unterschrift

## **EHRENWÖRTLICHE ERKLÄRUNG**

Ich erkläre ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benützt und die benutzten Quellen wörtlich zitiert sowie inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

.....

Unterschrift

## **DANKSAGUNG**

An dieser Stelle möchte ich mich bei allen bedanken, die mich während des Schreibens dieser Arbeit und meines Studiums unterstützt haben.

Besonderer Dank gilt meinem Betreuer Dipl.-Ing. Christoph Pangerl für seine Unterstützung beim Verfassen und Korrigieren dieser Arbeit.

## KURZFASSUNG

Die Digitalisierung schreitet über alle Branchen hinweg kontinuierlich voran. Auch die Tätigkeit der Hypothekarkreditvermittlung ist von diesem Trend nicht ausgenommen. Dennoch wird ein Großteil der Informationen, die für eine Kreditvermittlung notwendig sind, noch handschriftlich erfasst und die erforderlichen Dokumente werden per E-Mail ausgetauscht.

Im Zuge dieser Arbeit soll herausgefunden werden, wie der gesamte Prozess der Hypothekarkreditvermittlung sowohl auf Seiten der Kund/innen als auch auf Seiten der Kreditvermittler/innen mit Hilfe eines webbasierten Portals bestmöglich unterstützt werden kann. Das Ziel ist dabei nicht, das persönliche Beratungsgespräch zu ersetzen, sondern die Abwicklung digital zu unterstützen.

Ausgehend von der Analyse des Prozesses werden die Anforderungen an die Anwendung festgelegt. Anschließend wird ein Prototyp konzeptioniert und implementiert, der diese Anforderungen bestmöglich erfüllen soll. Der Fokus liegt dabei auf eine möglichst einfache Bedienung und größtmögliche Zeitersparnis für alle Beteiligten.

Der Prototyp ermöglicht, sämtliche Daten, die im Zuge einer Hypothekarkreditvermittlung erforderlich sind, direkt im Webportal einzugeben. Alle Dokumente können über die Anwendung hochgeladen oder mittels Kameraaufnahme hinzugefügt werden. Kreditvermittler/innen haben im Portal die Möglichkeit, Anfragen an Banken für Finanzierungsangebote zu stellen. Dadurch entfällt das zeitaufwändige, manuelle Zusammenstellen der Unterlagen.

Die von den Banken übermittelten Angebote können den Kund/innen in einem übersichtlichen Vergleich zur Verfügung gestellt werden. Das erhöht die Transparenz und erleichtert die Entscheidung für die bestmögliche Finanzierung.

Der Prototyp erfüllt alle gestellten Anforderungen und könnte zukünftig zu einer vollwertigen Plattform ausgebaut werden.

## **ABSTRACT**

Digitization is advancing continuously across all industries, including mortgage brokerage. Nevertheless, a significant portion of the information required for the loan brokerage is still recorded manually and the necessary documents are then sent by e-mail.

The goal of this work was to figure out how the entire process of mortgage loan brokerage could best be supported by a web-based platform for both customers and financial intermediaries. The aim was not to replace the personal consultation, but rather to support the process digitally.

First, the process was analyzed to identify the requirements for the application. Subsequently, a prototype that best meets these requirements was designed and implemented. The focus was on creating an easy-to-use application that enables all parties to save as much time as possible.

The portal enables the entry of all data required for a mortgage arrangement. All necessary documents can be uploaded within the application or by using the camera, and brokers can submit requests for offers to banks directly within the web platform, thereby eliminating the need for time-consuming manual document sorting. The resulting offers from the banks can then be made available to customers in a transparent manner, which makes it easier for customers to compare offers and select the best financing solution for their needs. Thus, the prototype meets all the identified requirements and could potentially be expanded to a full-featured platform in the future.

# INHALTSVERZEICHNIS

<b>1</b>	<b>EINLEITUNG</b> .....	<b>1</b>
1.1	Zielsetzung .....	1
1.2	Aufbau .....	2
<b>2</b>	<b>HYPOTHEKARKREDITVERMITTLUNG</b> .....	<b>3</b>
2.1	Marktsituation (Nachfrage nach Kreditvermittlungsleistungen) .....	3
2.1.1	Ablauf einer Hypothekarkreditvermittlung .....	5
2.2	Erwartete Vorteile durch das Webportal .....	8
<b>3</b>	<b>TECHNISCHE UND RECHTLICHE GRUNDLAGEN</b> .....	<b>10</b>
3.1	Webanwendungen .....	10
3.2	WebSocket .....	10
3.3	JSON .....	11
3.4	Vue.js .....	12
3.4.1	Aufbau einer Single-File-Vue.js-Komponente .....	12
3.4.2	Virtueller DOM .....	14
3.4.3	Direktiven .....	15
3.5	Datenschutz .....	17
<b>4</b>	<b>ANFORDERUNGEN</b> .....	<b>19</b>
4.1	Funktionale Anforderungen .....	19
4.2	Nicht-funktionale Anforderungen bzw. Qualitätsziele .....	26
<b>5</b>	<b>KONZEPTENTWICKLUNG</b> .....	<b>32</b>
5.1	Anwendungsarchitektur .....	32
5.1.1	Systemarchitektur .....	32
5.1.2	Datenbankstruktur .....	34
5.2	Überblick über die verwendeten Technologien .....	35

<b>6</b>	<b>IMPLEMENTIERUNG</b>	<b>38</b>
6.1	Zentraler Data-Store	38
6.2	Registrierung	41
6.3	Übersicht Kundenportal	43
6.4	Dateneingabe	45
6.5	Datei-Upload	49
6.6	Eine/n zweite/n Kreditnehmer/in hinzufügen	51
6.7	Dokumentenvorschau	52
6.8	Dokumente unterschreiben	53
6.9	PDF-Viewer und PDF-Template-Designer	54
6.10	Übersicht für Kreditvermittler/innen	57
6.11	Anfragen an Banken stellen	60
6.12	Angebotsvergleich	63
6.13	Chat-Funktion	65
6.14	Vermittlung abschließen und Daten löschen	65
6.15	Fazit der Implementierung des Prototyps	66
<b>7</b>	<b>ZUSAMMENFASSUNG</b>	<b>67</b>
7.1	Ausblick	68
	<b>ABBILDUNGSVERZEICHNIS</b>	<b>69</b>
	<b>TABELLENVERZEICHNIS</b>	<b>70</b>
	<b>LISTINGS</b>	<b>71</b>
	<b>LITERATURVERZEICHNIS</b>	<b>72</b>

# 1 EINLEITUNG

Immer mehr Dienstleistungen werden von ursprünglich reinen Offline-Dienstleistungen in digitale Online-Services transformiert. Die Corona-Krise wirkt als zusätzlicher Treiber dieser Verlagerung ins Internet. Auch die Kreditvermittlung ist von diesem Prozess nicht ausgenommen.

Großteils werden die notwendigen Informationen für die Kreditvergabe noch handschriftlich im Zuge des Beratungsgesprächs erfasst und die Dokumente per Mail versendet. Dabei müssen Kund/innen sämtliche Dokumente ausdrucken, ausfüllen, einscannen und per E-Mail retournieren. Das Nachfragen an den Kunden hinsichtlich noch fehlender Dokumente und auch der Versand der Kreditangebote, erfolgen ebenfalls per E-Mail. Dieser Prozess ist sowohl für Kund/innen als auch Kreditvermittler/innen unübersichtlich und nicht komfortabel.

Bestehende kommerzielle Plattformen wie zum Beispiel profin<sup>1</sup>, die zur Unterstützung des Kreditvermittlungsprozesses zur Verfügung stehen, fokussieren sich hauptsächlich auf die Vereinfachung des Prozesses auf Seiten der Kreditvermittler/innen und weniger auf das Generieren eines Mehrwerts auf Seiten der Kund/innen. Kreditvermittler/innen, die diese Portale nutzen, können auf die Weiterentwicklung solcher Angebote sehr wenig bis keinen Einfluss nehmen. Deshalb hat sich die Firma simplifin gegen die Nutzung eines bestehenden und für die Entwicklung eines eigenen Portals mit besonderem Fokus auf die Kundenbedürfnisse und der Vereinfachung der Abwicklung für den/die Kunden/Kundin entschieden.

Im Zuge dieser Arbeit wird die Entwicklung dieses Portals von der Anforderungsanalyse über das Konzipieren der Anwendung bis hin zur Implementierung dokumentiert.

Ziel der Arbeit und der Entwicklung des Prototyps ist die Beantwortung der nachfolgenden Forschungsfrage:

**Wie lässt sich der Prozess der Hypothekarkreditvermittlung durch ein Webportal sowohl für Kund/innen als auch Kreditvermittler/innen digital unterstützen?**

## 1.1 Zielsetzung

Ziel dieser Arbeit ist, herauszufinden, wie der Prozess der Hypothekarkreditvermittlung bei der Firma simplifin mit Hilfe einer Webanwendung bestmöglich digital unterstützt werden kann. Dabei soll anhand eines Prototyps erarbeitet werden, wie die notwendigen Funktionen abgebildet werden können. Priorität haben die einfache Bedienung für Kund/innen und Kreditvermittler/innen

---

<sup>1</sup> Profin ist eine Marke der Infina Credit Broker GmbH: <https://www.profin.at>



und die Verlagerung des Dokumentenaustauschs von der Übermittlung per E-Mail hin zur Nutzung des Webportals.

## **1.2 Aufbau**

Zu Beginn der Arbeit wird ein Überblick über die Marktsituation und den Ablauf von Hypothekarkreditvermittlungen gegeben. Basierend auf internen Workshops bei der Firma simplifin werden anschließend die funktionalen Anforderungen und Qualitätsziele des Kreditvermittlungsportals ausgearbeitet.

Mittels Literaturrecherche werden die wichtigsten theoretischen Grundlagen und verwendeten Technologien der Webanwendung erörtert.

Auf Basis der erarbeiteten Anforderungen wird ein Anwendungskonzept erstellt. Dabei werden zuerst die Anwendungsarchitektur und Datenbankstruktur erläutert und anschließend die wichtigsten verwendeten Technologien aufgelistet.

Ein großer Teil der Arbeit befasst sich mit der Implementierung des Prototyps. Im Kapitel 6 werden deshalb die wichtigsten Teile der Anwendung und des User-Interfaces anhand von Screenshots dargestellt und erklärt.

Abschließend werden die Ergebnisse dieser Arbeit zusammenfasst und mögliche Erweiterungs- bzw. Verbesserungsmöglichkeiten aufgezeigt.

## **2 HYPOTHEKARKREDITVERMITTLUNG**

Am Markt gibt es eine Vielzahl an Kreditangeboten, welche für Kund/innen, die in der Regel nur sehr selten – oder eventuell nur einmalig im Zuge einer Immobilienfinanzierung – mit Finanzierungen in Berührung kommen, nur sehr schwer zu überblicken ist. Kreditvermittler/innen holen für ihre Kund/innen mehrere Angebote bei unterschiedlichen Banken ein. Das erleichtert Kreditnehmern/Kreditnehmerinnen den Vergleich der Angebote und ermöglicht ihnen, von besseren Kreditkonditionen zu profitieren.

Beratungen in finanziellen Angelegenheiten bedürfen besonderen Anforderungen und Qualifikationen. Deshalb ist die gewerbliche Vermögensberatung, und damit auch die Personal- und Hypothekarkreditvermittlung, ein reglementiertes Gewerbe. Das bedeutet, dass für die Gewerbeanmeldung bestimmte Zugangsvoraussetzungen erfüllt werden müssen. Diese können eine Befähigungsprüfung, ein facheinschlägiges Studium oder eine ausreichend lange facheinschlägige Tätigkeit sein. Während der Ausübung des Gewerbes sind jährliche Schulungen nachzuweisen. (Eglmayr, et al., 2018)

Die Personal- und Hypothekarkreditvermittlung ist zum Schutz der Kund/innen unter anderem durch das Hypothekar- und Immobilienkreditgesetz (HIKrG), die Standesregeln für Kreditvermittlung und das Fernfinanzdienstleistungsgesetz stark reglementiert. (Fachverband Finanzdienstleistung, 2020) Diese Gesetze schreiben zum Beispiel vor, welche Informationen und in welcher Form den Kund/innen im Zuge einer Kreditvermittlung zu übermitteln sind.

Da es sich bei der gewerblichen Vermögensberatung außerdem um ein Zuverlässigkeitsgewerbe handelt, kann gemäß § 95 Absatz 1 Gewerbeordnung 1994 bei rechtlichen Verstößen die Gewerbeberechtigung entzogen werden.

### **2.1 Marktsituation (Nachfrage nach Kreditvermittlungsleistungen)**

Das anhaltend niedrige Zinsniveau, die Erwartung steigender Immobilienpreise und höhere Ersparnisse aufgrund eingeschränkter Konsummöglichkeiten sowie der Wunsch nach einer Verbesserung der Wohnsituation aufgrund verstärkter Home-Office-Nutzung treiben die Nachfrage nach Wohnimmobilien nach oben. (Oesterreichische Nationalbank, 2021)

Dadurch steigen sowohl die Summe der aushaftenden Wohnbaukredite privater Haushalte als auch die monatlich vergebenen Kredite an private Haushalte für Wohnbauzwecke seit einigen Jahren konstant an (Abbildung 2-1 und Abbildung 2-2).

### Entwicklung der aushaftenden Wohnbaukredite privater Haushalte in Österreich

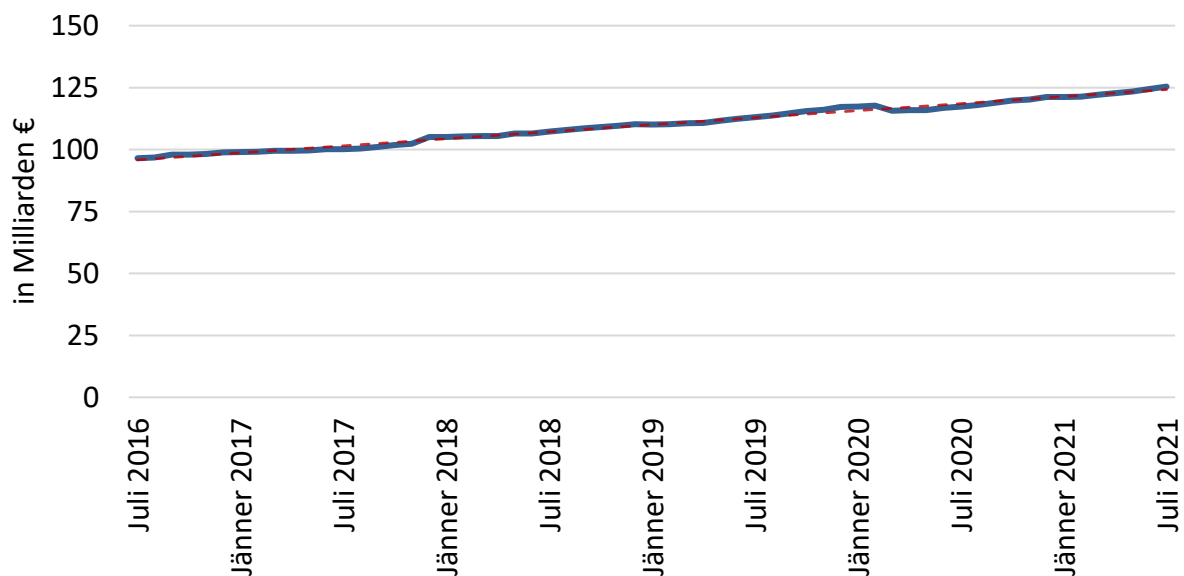


Abbildung 2-1: Entwicklung der aushaftenden Wohnbaukredite privater Haushalte in Österreich (Eigene Abbildung, Datenquelle: Österreichische Nationalbank, 2021)

### Neukreditvergabe an private Haushalte für Wohnbauzwecke in Österreich

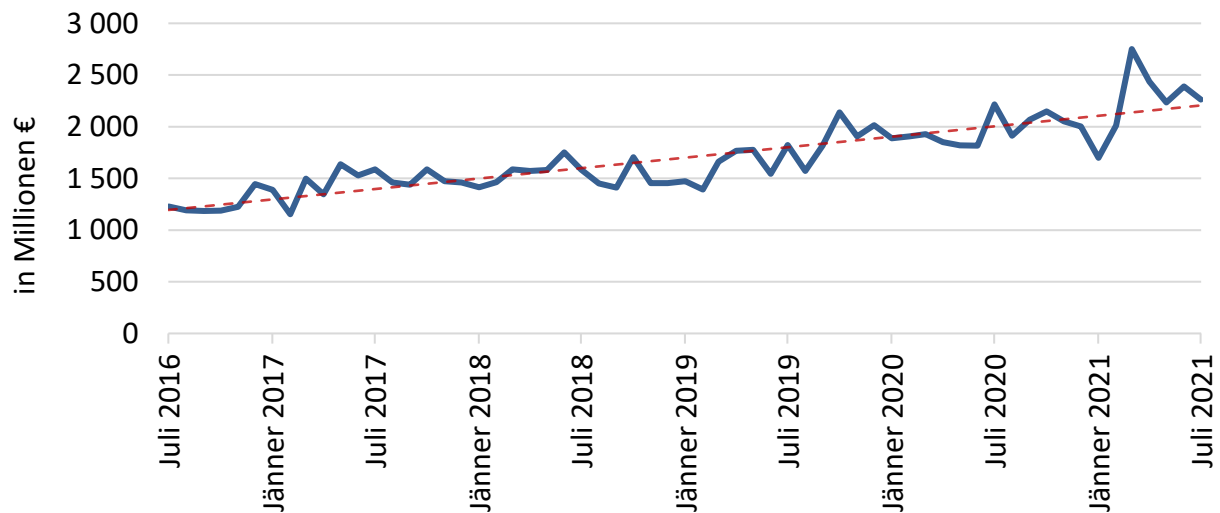


Abbildung 2-2: Neukreditvergabe an private Haushalte für Wohnbauzwecke in Österreich (Eigene Abbildung, Datenquelle: Österreichische Nationalbank, 2021)

Schon der „Digital Banking“-Bericht der bitkom von Juni 2016 (Dietz, 2016) zeigte ein steigendes Interesse an Online-Banking-Produkten. 7 % der Befragten gaben an, bereits einen Kredit online aufgenommen zu haben und weitere 16 % konnten sich vorstellen, das künftig zu tun.

Im „Digital Finance“-Bericht der bitkom von Oktober 2021 wurde diese Frage nicht mehr in gleicher Weise gestellt, jedoch zeigt dieser Bericht, dass die Bereitschaft, Online-Banking zu nutzen, in den letzten Jahren stetig zugenommen hat und mittlerweile bei durchschnittlich 80 %

liegt, alle Altersgruppen miteingerechnet. 39 % der befragten Personen über 65 Jahren geben an, Online-Banking zu nutzen. (Bitkom e.V., 2021)

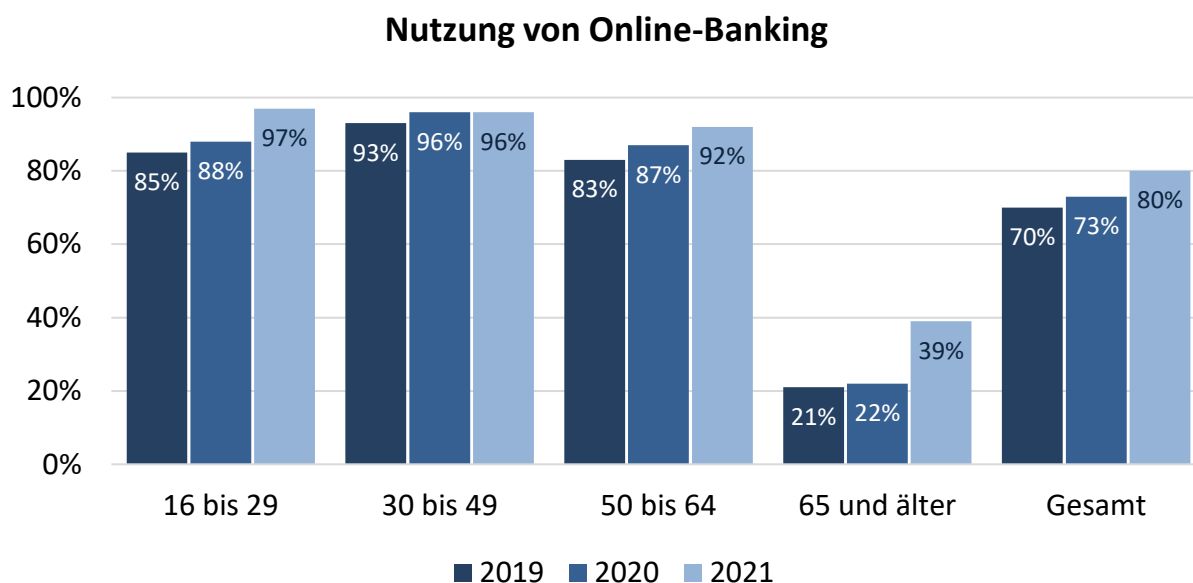


Abbildung 2-3: Nutzung von Online-Banking (Eigene Abbildung, Datenquelle: Bitkom e.V., 2021)

Fürderer (Hellenkamp & Fürderer, 2016) stellt fest:

*„Ein weiterer offensichtlich erkennbarer (Mega)Trend ist, dass Bankkunden zunehmend aufgeklärter und besser informiert werden. Sie folgen heute nicht mehr (ausschließlich) den Informationen und Empfehlungen eines Kundenberaters, sondern generieren eigeninitiativ Informationen aus dem Internet.“*

Der Trend der Digitalisierung ist somit auch im Bereich des Bankgeschäfts und der Kreditvermittlung nicht aufzuhalten. Das führte auch bei die Firma simplifin zur Entscheidung, eine digitale Lösung zu entwickeln.

Aufgrund der starken Nachfrage nach Wohnkrediten im privaten Bereich ist der Markt der Kreditvermittlung in Österreich sehr umkämpft. Deshalb ist es besonders wichtig, Kund/innen bestmöglich zu beraten und diesen die Abwicklung der Kreditvergabe so einfach und zeitsparend wie möglich zu machen. Einige Banken haben das mittlerweile erkannt und bieten eigene Portale zur Abwicklung der Finanzierung an. Diese bankeneigenen Portale wie zum Beispiel die „BAF PartnerFIN App“ der Bank Austria oder e-hyp der Bank99 decken aber selbstredend nur das eigene Portfolio der jeweiligen Bank ab.

### 2.1.1 Ablauf einer Hypothekarkreditvermittlung

Der klassische Kreditvermittlungsprozess umfasst mehrere Prozessschritte, welche nachfolgend am Beispiel der Firma simplifin in vereinfachter Form erläutert werden. Der Fokus der Betrachtung liegt dabei auf dem klassischen Prozess der Kreditvermittlung aus der Perspektive eines/einer Kreditvermittlers/Kreditvermittlerin ohne die unterstützende Verwendung eines Portals.

Abbildung 2-4 gliedert den Prozess in fünf Phasen, die im Folgenden erläutert werden. Auch wenn in der Abbildung die Beratung nur in Phase 2 als Teil der Erstberatung angeführt ist, bedeutet das nicht, dass hier die einzige Beratung stattfindet. Kund/innen werden während der gesamten Kreditvermittlung von ihrem/ihrer Berater/in betreut, auch wenn die umfassendsten Beratungen im Zuge des Erstgesprächs sowie der Besprechung der Angebote erfolgen.

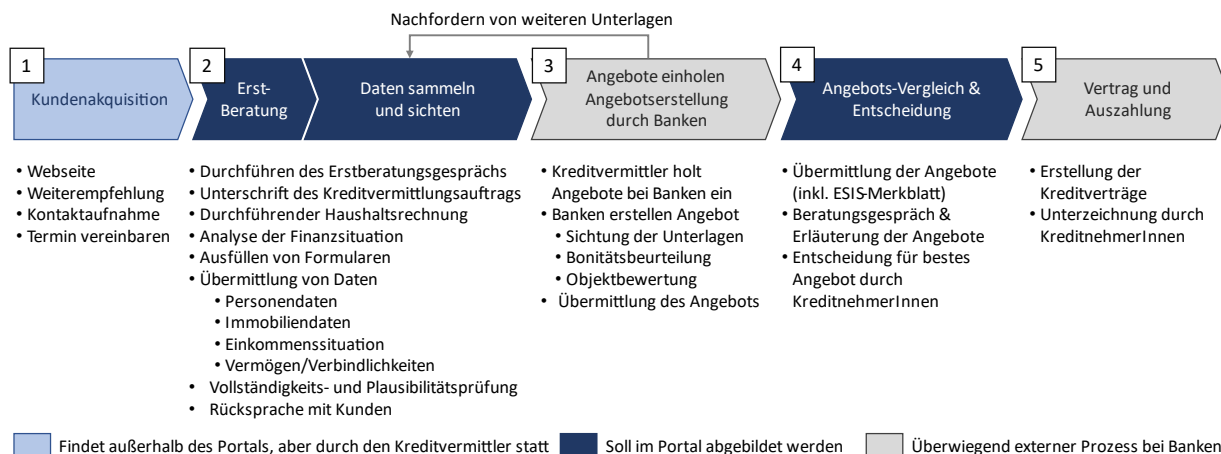


Abbildung 2-4: Ablauf einer Immobilienfinanzierung (In Anlehnung an Eickholt, 2015; Oesterreichische Nationalbank (OeNB), 2004)

## 1. Kundenakquise

Die Kundenakquise hat das Ziel, neue Kund/innen für das Unternehmen zu gewinnen. Die Kontaktaufnahme erfolgt hier durch den/die Kunden/Kundin.

Hellenkamp und Fürderer (2016) stellen fest: „Information geht vor Werbung“. In dieser Phase sind Kund/innen an objektiven Informationen zum Thema interessiert. Qualitativ hochwertige Inhalte, zum Beispiel auf der Webseite, dienen nicht nur als Informationsquelle für den/die Kunden/Kundin, sondern erhöhen gleichzeitig das Google-Ranking. (Hellenkamp & Fürderer, 2016)

Um in der heutigen Zeit erfolgreich Neukund/innen zu gewinnen, ist es nicht mehr ausreichend, guten Content nur auf der eigenen Webseite – und über das Auffinden mit Hilfe von Suchmaschinen – zur Verfügung zu stellen. Stattdessen müssen diese Inhalte den Kund/innen über die verschiedensten Vertriebs- und Kommunikationskanäle zugänglich gemacht werden. Die wichtigsten Plattformen hierfür sind derzeit Facebook, Twitter, Instagram, YouTube, TikTok, LinkedIn und XING. Auch gilt, dass das alleinige Schalten bezahlter Werbung weniger Erfolg bringt als das regelmäßige Posten von qualitativ hochwertigen Inhalten auf diesen Plattformen.

## 2. Erstberatung, Daten sammeln und sichten

Vor Ausübung jeglicher Kreditvermittlungstätigkeit sind Kreditvermittler/innen gesetzlich dazu verpflichtet, die Standardinformationen auszuhändigen. Der Inhalt dieser Standardinformationen<sup>2</sup>

<sup>2</sup> „Zum Inhalt zählen insbesondere folgende Angaben: Identität und Anschrift, GISA-Zahl, die Art

ist in § 4 der Standesregeln für Kreditvermittlung festgehalten und enthält Informationen zum Schutz der Verbraucher/innen.

Im Zuge der Erstberatung lernen Kreditvermittler/innen die Kund/innen und ihre Lebenssituation kennen. Es werden Informationen über die Immobilie eingeholt, eine Haushaltsrechnung durchgeführt, die finanzielle Situation, Vermögen und Verbindlichkeiten der Kunden analysiert sowie Wünsche hinsichtlich der Finanzierung berücksichtigt. Es wird empfohlen das Beratungsgespräch zu dokumentieren.

Meist hatten Kund/innen bis zu diesem Zeitpunkt wenig oder gar keine Erfahrung mit Finanzierungen. Dementsprechend wichtig sind die ausführliche Beratung und Aufklärung über die Möglichkeiten, Varianten und Risiken von Finanzierungen.

Um die Kreditvermittlung durchführen zu können, muss der/die Kunde/Kundin die beteiligten Banken zum Zweck der Kreditvermittlung vom Bankgeheimnis gem. § 38 Abs 2 Z 5 BWG entbinden. Diese Vollmacht wird meist mit der Unterschrift des Kreditvermittlungsauftrags erteilt. Dieser beinhaltet als wesentliche Informationen unter anderem die gewünschte Kredithöhe (Auszahlungsbetrag), höchstmögliche monatliche Kreditbelastung, maximale Provision, den spätesten Zeitpunkt für die Kreditzusage und die Mindest- und Höchstlaufzeiten der Finanzierung.

Im Anschluss an das Beratungsgespräch werden die notwendigen Unterlagen für das Anfragen der Angebote bei den Banken eingeholt. Fehlende Unterlagen werden durch den/die Kunden/Kundin in der Regel per E-Mail nachgereicht.

### **3. Angebote einholen**

Basierend auf den Angaben und Wünschen der Kund/innen werden mehrere Angebote mit zum Beispiel verschiedenen Verzinsungsvarianten (fixer oder variabler Zinssatz) oder unterschiedlichen Laufzeiten bei mehreren Banken eingeholt.

Bei der Übermittlung der Daten an die Banken werden grundsätzlich zwei Vorgehensweisen voneinander unterschieden:

- Übermittlung per E-Mail bei Banken ohne eigene Finanzierungssoftware: Hier werden den Partnerbetreuern/-innen der Banken die erforderlichen Unterlagen per E-Mail übermittelt. Auch die Angebotsrückübermittlung erfolgt hier per E-Mail.
- Banken mit eigenem Finanzierungsportal ohne E-Mail-Übermittlungsmöglichkeit: Einige Banken bieten mittlerweile eigene Finanzierungsportale für ihre eigenen Produkte an. In der Regel wird von diesen die Nutzung des Portals für die Übermittlung der Daten und die Angebotsübermittlung bevorzugt oder sogar gefordert. Des Weiteren ist die Bearbeitungsdauer bei der Nutzung dieser Portale meist geringer, weshalb die Nutzung dieser Portale auch für Kreditvermittler/innen und Kund/innen einen Vorteil bringt.

Banken führen zuerst eine Vollständigkeits- und Konsistenzprüfung der eingereichten Unterlagen anhand der übermittelten Daten durch und fordern gegebenenfalls weitere oder fehlende Dokumente nach. Nach einer Vorprüfung des Kreditantrags werden eine umfassende Bonitätsprüfung, eine Bewertung der Sicherheiten und eine Risikobeurteilung durchgeführt. Verlaufen die Prüfungen positiv, wird im nächsten Schritt ein Angebot erstellt und dem/der Kreditvermittler/in übermittelt. (Oesterreichische Nationalbank (OeNB), 2004)

#### 4. Angebotsvergleich & Entscheidung

Die von den Banken übermittelten Angebote werden dem/der Kunden/Kundin erst nach einer Überprüfung durch den/die Kreditvermittler/in weitergeleitet. Um die Vergleichbarkeit der Angebote für Kund/innen zu erleichtern, schreibt § 8 Hypothekar- und Immobilienkreditgesetz bestimmte vorvertragliche Informationspflichten vor. Konkret müssen Informationen über den Kredit anhand des standardisierten ESIS-Merkblattes vorgelegt werden.

Nach Übermittlung der Unterlagen wird ein Termin zur Besprechung der Angebote und Beratung mit dem/der Kunden/Kundin vereinbart.

Sofern das Vorhaben noch Bestand hat, sowie nach einer eventuellen Anpassung des Angebots und etwas Bedenkzeit entscheidet sich der/die Kunde/Kundin für ein Angebot.

#### 5. Vertrag und Auszahlung

Nach der Zusage beginnt die Bank mit der Erstellung der Vertragsunterlagen. Bei Banken mit Filialnetz wird ein Termin zur Unterzeichnung der Kreditverträge vereinbart. Bei Direktbanken erfolgt die Übermittlung der finalen Vertragsunterlagen per Post.

## 2.2 Erwartete Vorteile durch das Webportal

Das Hauptziel des Webportals ist die Vereinfachung des Ablaufs für Kund/innen. Natürlich sollten der Aufwand und die Durchlaufzeiten auch auf Seiten der Berater/innen reduziert werden.

Die nachfolgende Abbildung 2-5 zeigt die theoretisch erwartete zeitliche Ersparnis, die sich durch das Portal im Verlauf der Kreditvermittlung ergibt.

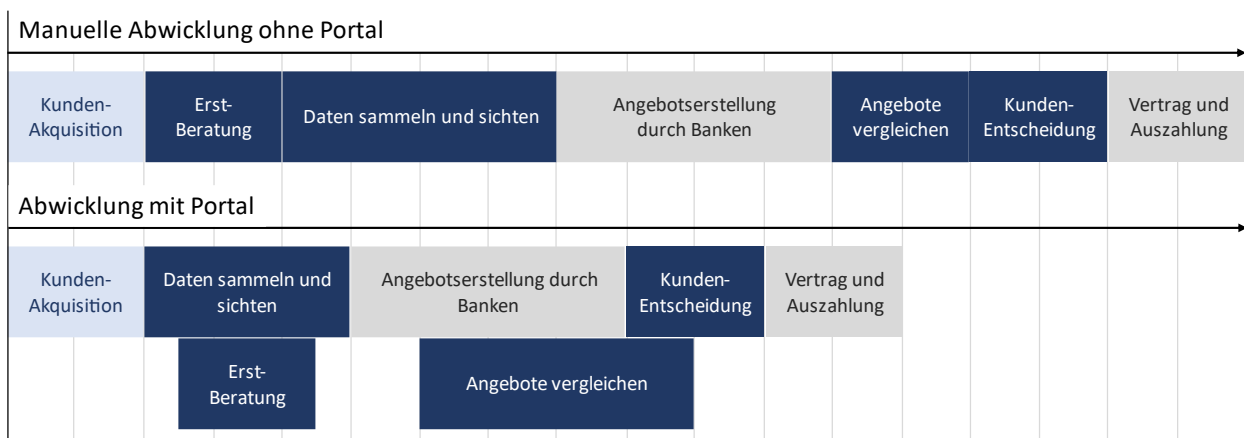


Abbildung 2-5: Zeitersparnis durch das Webportal (Eigene Abbildung)

Bei der herkömmlichen Kreditvermittlung wird in der Regel zuerst das Beratungsgespräch durchgeführt und erst im Zuge dessen oder anschließend mit der Datenübermittlung begonnen. Benutzer/innen, die über die Webseite auf das Webportal gelangen, können die erforderlichen Daten direkt nach der Registrierung eingeben und Dokumente hochladen. Ein weiterer Vorteil ist, dass sich der/die jeweilige Berater/in anhand der Daten bereits im Vorfeld besser auf das Erstgespräch vorbereiten kann. Die Terminvereinbarung für das Erstgespräch erfolgt weiterhin per Telefonat.

Das Einholen der Daten (Daten sammeln und sichten) wird durch die Portal-Unterstützung sowohl für den/die Kunden/Kundin als auch insbesondere für den/die Berater/in erleichtert und verkürzt. Dadurch, dass die Kund/innen die Daten selbstständig im Portal eintragen, wird der Aufwand für den/die Kreditvermittler/in erheblich reduziert. Der zusätzliche Zeitaufwand für die Digitalisierung der Daten, die ansonsten mündlich im Laufe des Erstgesprächs erfasst werden, wird somit auf den/die Kunden/Kundin ausgelagert.

Des Weiteren wird die Übermittlung der Daten an die Banken vereinfacht. Derzeit müssen Kreditvermittler/innen die von den Banken geforderten Unterlagen und Daten per E-Mail übermitteln. Das Portal erstellt diese E-Mails automatisch mit den Unterlagen, die die jeweilige Bank fordert. Diese Vereinfachung in der Übermittlung betrifft jedoch nicht alle Banken. Manche Banken verlangen die Nutzung ihrer Online-Portale zur Übermittlung der Daten und Dokumente und bieten kleineren Firmen wie simplifin keinen Zugriff über eine dokumentierte Schnittstelle.

Ohne Portal wird mit der Übermittlung der Angebote an die Kund/innen normalerweise gewartet, bis mehrere Angebote von unterschiedlichen Banken beim/bei der Kreditvermittler/in eingetroffen und fertig mit den Banken verhandelt sind. Bei der Abwicklung über das Portal werden die erhaltenen Angebote der Banken direkt den Kund/innen zugänglich gemacht, wodurch diese laufend und transparent die Angebote vergleichen können. Zudem werden Änderungen der Angebote aufgrund von Nachverhandlungen der Kreditvermittler/innen mit den Banken transparent und nachvollziehbar im Portal dargestellt.

Natürlich kann die Entscheidung für ein bestimmtes Kreditangebot erst getroffen werden, nachdem alle Angebote vorliegen. Dennoch kann die Kundenentscheidung durch die laufende Bereitstellung der Angebote im Portal ein wenig mit dem Einholen der Angebote parallelisiert werden. Zudem erleichtert ein übersichtlicher Angebotsvergleich direkt im Portal die Entscheidung.

Als Summe aller Maßnahmen verringert sich mit Hilfe des Portals die Durchlaufzeit der Abwicklung.



## 3 TECHNISCHE UND RECHTLICHE GRUNDLAGEN

In diesem Kapitel sollen die theoretischen und technischen Grundlagen der verwendeten Technologien erklärt werden. Des Weiteren werden die rechtlichen Aspekte, die hinsichtlich der Datenschutzgrundverordnung zu berücksichtigen sind, erörtert.

### 3.1 Webanwendungen

Eine Grundvoraussetzung für das Portal ist die Lauffähigkeit auf allen gängigen Plattformen und Systemen. Darüber hinaus sollte die Anwendung über das Internet leicht zugänglich sein, ohne die lokale Installation einer Software zu erfordern. Deshalb wird, wie im Titel der Arbeit definiert, ein webbasiertes Portal und somit eine Webanwendung entwickelt.

Webanwendungen basieren auf den offenen Webtechnologien HTML 5, CSS und JavaScript und können somit mit allen modernen Browsern und damit auf allen gängigen Plattformen ausgeführt werden. HTML (Hypertext Markup Language) ist eine Auszeichnungssprache, die den Inhalt und die Struktur definiert, CSS (Cascading Style Sheets) definiert die Darstellung und den Stil und JavaScript ist zuständig für die dynamischen Elemente und die Anwendungslogik. (Heitkötter, Majchrzak, Wolfgang, & Kuchen, 2012)

Grundsätzlich handelt es sich bei Webanwendungen um herkömmliche Webseiten, die einen Anwendungscharakter aufweisen. Auch wenn sie wie normale Webseiten in einem Browser geladen werden, erfolgen bei Webanwendungen umfangreichere und komplexere Interaktionen als bei Webseiten, die primär zur Darstellung von Informationen dienen. Eine spezielle Form der Webanwendungen sind Single-Page-Applikationen (SPA). Sie bestehen aus einer einzelnen HTML-Datei, deren Inhalte dynamisch nachgeladen werden. Dadurch erfolgt beim Navigieren kein sichtbarer Seitenwechsel wie beim Klick auf einen herkömmlichen Hyperlink, stattdessen werden die Inhalte ohne ein erneutes Laden der Seite dynamisch angezeigt. (Heitkötter, Majchrzak, Wolfgang, & Kuchen, 2012)

Der Nachteil von Webanwendungen gegenüber nativen Anwendungen ist der begrenzte Zugriff auf Hardwarefunktionen. Mittlerweile bietet der HTML-5-Standard aber eine Vielzahl an APIs, unter anderem auch für den Zugriff auf Funktionen wie die Position oder die Kamera. Darüberhinausgehende Funktionen sind für das geplante Portal nicht erforderlich.

### 3.2 WebSocket

Beim HTTP-Protokoll muss eine Kommunikation immer vom Client angestoßen werden. Der Client sendet einen Request an den Server und dieser antwortet mit einem Response. Dadurch kann der Server aber nicht selbstständig die Datenübertragung an einen Client anstoßen. Damit der Client Änderungen der Daten am Server erhält, muss dieser Anfragen kontinuierlich, in einem bestimmten Intervall an den Server senden. (Zammetti, 2019)

Im Gegensatz dazu ermöglicht das WebSocket-Protokoll eine bidirektionale Kommunikation. Das bedeutet, dass der Server auch aktiv Daten an den Client übermitteln kann. Um eine WebSocket-Verbindung herzustellen, wird zuerst ein HTTP-Request mit der Anfrage für eine WebSocket-Verbindung an den Server gesendet. Wenn der Server das WebSocket-Protokoll unterstützt, stimmt er der Verbindung zu, der HTTP-Request wird verworfen, aber die zugrundeliegende TCP/IP-Verbindung wird aufrechterhalten. Über diese Verbindung können nun Client und Server kommunizieren. (Zammetti, 2019)

Das WebSocket-Protokoll bietet deshalb zum Beispiel Vorteile bei einer Chat-Anwendung. Wenn ein/e Chat-Teilnehmer/in eine Nachricht an den Server sendet, könnte diese Nachricht bei der Verwendung des HTTP-Protokolls vom Server nicht aktiv an den/die weiteren Teilnehmer/innen weitergeleitet werden. Alle Teilnehmer/innen müssten alle paar Sekunden am Server nachfragen, ob neue Nachrichten vorhanden sind und diese gegebenenfalls abrufen. Durch eine WebSocket-Verbindung kann die Nachricht vom Server stattdessen aktiv an alle Chat-Teilnehmer/innen übermittelt werden.

### 3.3 JSON

Bei JSON (JavaScript Object Notation) handelt es sich um eine sehr einfache, textbasierte Syntax um strukturierte Daten auszutauschen. Obwohl sich die Notation und auch der Name an den in ECMAScript definierten Javascript Objekten orientiert, verwendet sie viele Konventionen, die auch in anderen Programmiersprachen verwendet werden und ist zudem auch komplett unabhängig von der verwendeten Programmiersprache anwendbar. (Ecma International, 2017)

JSON bietet grundsätzlich zwei Strukturen, um Daten zu definieren (Ecma International, 2017):

- Name / Wert-Paare
- Geordnete Werte-Liste / Array

Beispiel für eine JSON-Datei:

```
{
  "vorname": "Marco",
  "nachname": "Schweiger",
  "alter": 33,
  "hobbies": ["Programmieren", "Schwimmen", "Laufen"],
  "telefon": [
    {"typ": "privat", "nummer": "123456"},
    {"typ": "firma", "nummer": "789012"}
  ]
}
```

Listing 3-1: Beispiel für ein JSON-File

## 3.4 Vue.js

Vue.js ist ein leichtgewichtiges Web-Framework, dessen Popularität in den letzten Jahren immer weiter zugenommen hat. Es handelt sich um ein clientseitiges JavaScript-Webframework, welches die Entwicklung von Single-Page-Anwendungen nach dem Model-View-View-Controller-Modell vereinfacht. Vue.js ist ein sehr leicht zu erlernendes und gut erweiterbares Framework. Es unterstützt Direktiven (v-for, v-if, usw.), auf deren Basis Aktionen, wie Schleifen, if-Bedingungen, usw. im Frontend umgesetzt werden können. Eine der Kernfunktionen des Frameworks ist die Möglichkeit der Datenbindung (Two-Way-Databinding). Das bedeutet, dass Eingaben, die im User-Interface getätigt werden, sofort in die JavaScript-Variablen übernommen werden. Umgekehrt werden dadurch auch Änderungen an den Variablen durch den Programmcode sofort in der Benutzeroberfläche angezeigt. Durch die Strukturierung des Codes in wiederverwendbare Komponenten können benutzerdefinierte Elemente erstellt werden, die in der gesamten Anwendung wiederverwendet werden können. (Steyer, 2019)

Vue.js ist mittlerweile ein sehr umfangreiches Framework. Aufgrund des immensen Funktionsumfangs von Vue.js werden im Folgenden nur die für das Verständnis der Code-Beispiele in dieser Arbeit notwendigen Funktionen beschrieben.

### 3.4.1 Aufbau einer Single-File-Vue.js-Komponente

In Vue.js lassen sich wiederverwendbare Komponenten erstellen. Damit lassen sich bestimmte Elemente, inklusive der Logik, erstellen, die anschließend beliebig oft innerhalb der Anwendung wiederverwendet und platziert werden können. Diese Komponenten lassen sich modularisieren und als Vue-Dateien abspeichern. Diese einzelne Datei (Single-File) enthält sowohl die Ansicht (Template), die Logik (Script) als auch das Design (Style). Anhand des nachfolgenden Code-Beispiels werden die wichtigsten Funktionen von Vue.js, die im Zuge dieser Arbeit verwendet werden, erklärt.

```
<template>
  <div>
    <input v-model="headline" />
    <h1>{{ headline }}</h1>
    <h1>{{ getHeadline }}</h1>
    <h1>{{ headline | capitalize }}</h1>
    <h1>{{ ParameterA }}</h1>
  </div>
</template>

<script>
export default {
  name: "Komponentenname",
  props: ["ParameterA"],
  data() {
    return {
      headline: "Single-File-Komponente",
```

```
    };
  },
  methods: {
    speichereDaten: function () {
      localStorage.setItem("headline", this.headline);
    },
  },
  watch: {
    headline: function () {
      this.speichereDaten();
    },
  },
  computed: {
    getHeadline() {
      return this.headline.toUpperCase();
    },
  },
  filters: {
    capitalize: function (value) {
      return value.toUpperCase();
    },
  },
};
</script>

<style scoped>
h1 {
  color: red;
}
</style>
```

Listing 3-2: Vue.js-Single File Komponente

Die Datei gliedert sich in die folgenden drei Bereiche:

- **Template:** Im Template-Bereich werden die Struktur und der Inhalt der Komponente anhand von normalem HTML definiert. Im h1-Tag wird die Variable *headline* ausgegeben. Die Variable wird dabei mit Hilfe der sogenannten Mustache-Syntax, so wird die Notation mit den doppelten geschweiften Klammern `{{ }}` bezeichnet, ausgegeben. (Steyer, 2019)
- **Script:** definiert die Funktionalität der Komponente
  - **name:** Jede Komponente sollte einen eindeutigen Namen aufweisen.
  - **props:** Mit Hilfe von props können Werte von übergeordneten Komponenten (Parent-Komponenten) an untergeordnete Komponenten (Child-Komponenten) übergeben werden. In Listing 3-2 wird so der Parameter *ParameterA* übergeben und im template-Bereich ausgegeben.
  - **data:** Innerhalb des Data-Bereichs werden die Daten bzw. Variablen, mit denen gearbeitet wird, im JSON-Format angegeben. In Listing 3-2 wird für die Variable

*headline* der Text „Überschrift“ festgelegt und im h1-Tag (template-Bereich) ausgegeben.

- **methods:** Bei den Vue-Methoden handelt es sich um Standard-JavaScript-Funktionen, die im Vue-Objekt definiert werden. Dadurch erkennt das Framework, dass es sich um Vue-Methoden handelt. (Steyer, 2019) Im Beispiel wird beim Aufruf der Methode die Headline in den LocalStorage des Browsers gespeichert. Hier könnten zum Beispiel auch Daten vom einem Webserver geladen werden.
- **computed:** Analog zu den Methoden handelt es sich bei computed properties um JavaScript-Funktionen. Im Gegensatz zu normalen Methoden, die explizit aufgerufen werden müssen, werden diese Funktionen jedes Mal ausgeführt, sobald sich ein Wert, der innerhalb der Funktion verwendet wird, verändert. (Au-Yeung, 2021) Im Beispiel in Listing 3-2 wird, wenn sich die Variable *headline* ändert, der Wert der Variable in Großbuchstaben umgewandelt, zurückgegeben und innerhalb des template-Bereichs als *getHeadline* unmittelbar ausgegeben.
- **watch:** Im watch-Objekt werden die sogenannten Watchers definiert. Mit deren Hilfe können Variablen auf Änderungen überwacht werden und bei Erkennen einer Änderung, Funktionen ausgeführt werden. (Au-Yeung, 2021). Im Gegensatz zu computed-properties benötigen Watcher keinen Rückgabewert und können somit auch asynchrone Funktionen ausführen. (Steyer, 2019) Im Beispiel wird, sobald eine Änderung der reaktiven Variable *headline* erkannt wird, die Methode *speichereDaten* ausgeführt.
- **filters:** Eine weitere nützliche Funktion, insbesondere für Formatierungszwecke, sind Filter. Im Beispiel wird der Filter mit dem Namen *capitalize* angelegt und im template-Bereich, abgetrennt durch das Pipe-Symbol, nach dem Variablennamen angehängt. (Steyer, 2019) Dadurch wird der Wert, ähnlich wie bei der bereits vorgestellten Computed-Property, in Großbuchstaben dargestellt. Der wesentliche Vorteil ist, dass solche Filter sehr einfach auf mehrere Variablen angewendet werden können. Somit lassen sich zum Beispiel Zahlenformatierungen (z.B. genau eine Kommastelle oder ein nachgestelltes €-Symbol) sehr einfach realisieren.
- **Style:** definiert das Design mit Hilfe von CSS. Das Attribut *scoped* definiert, dass die angegebenen CSS-Stile nur innerhalb dieser Komponente verwendet werden. Mit dem angegebenen CSS wird der h1-Tag, der im template-Bereich definiert wurde, in Rot dargestellt.

### 3.4.2 Virtueller DOM

Das DOM (Document Object Model) ist eine Schnittstelle, die jeder Browser bereitstellt und Objekte (XML bzw. HTML) als Baumstruktur darstellt. Über die Schnittstelle lassen sich Manipulationen an den Elementen und der Struktur dieser Baumstruktur vornehmen. Änderungen im DOM sind ressourcenintensiv, da der Browser die Webseite bei jeder Änderung neu rendern muss. (Steyer, 2019)

In Vue.js werden Änderungen, die im Frontend vorgenommen werden sollen, nicht am eigentlichen DOM vorgenommen, sondern an einer Kopie des DOM – dem virtuellen DOM. Bevor Änderungen am eigentlichen DOM vorgenommen werden, wird dieser mit dem virtuellen DOM verglichen. Dadurch lassen sich mehrere Änderungen zusammenfassen und die Anzahl der Manipulationen am realen DOM verringern. Diese Erklärung ist eine vereinfachte Erklärung, im Hintergrund laufen noch weitere Optimierungen, um die Performance zu verbessern. (Steyer, 2019)

### 3.4.3 Direktiven

Vue.js stellt eine Reihe an sogenannten „Direktiven“ bereit. Diese dienen dazu, Manipulationen im Frontend und damit in der Benutzeroberfläche durchzuführen. Die Direktiven werden in Form von HTML-Attributen in das HTML-Template eingebunden. Beispiele für solche Direktiven sind *v-for*, *v-if*, *v-else*, *v-bind*, *v-model*, *v-show*, *v-on* und *v-slot*. (Steyer, 2019).

#### v-if- und v-else-Direktiven

Diese beiden Direktiven entsprechen einer allgemein bekannten if-else-Bedingung und ermöglichen, dass bestimmte Blöcke nur bedingt gerendert/angezeigt werden. Nachfolgender Code-Ausschnitt zeigt ein Beispiel für eine solche Direktive. Nur wenn die Variable *selected* den Wert *true* annimmt, wird *Hallo* angezeigt. Sollte die Variable *false* sein, wird die else-Bedingung erfüllt und *Auf Wiedersehen* angezeigt.

```
<div id="app">
  <div v-if="selected == true">Hallo</div>
  <div v-else>Auf Wiedersehen</div>
</div>
```

Listing 3-3: v-if und v-else Direktive

#### v-for-Direktive

Die v-for-Direktive wird verwendet, um Arrays zu durchlaufen und für jedes Element im Array Daten anzuzeigen. Dafür muss die v-for-Direktive als Attribut zu dem HTML-Element hinzugefügt werden, das wiederholt werden sollte. (Street, Passaglia, & Halliday, 2018)

Im nachfolgenden Beispiel wird das Array *personen* durchlaufen und in einem Listen-Element der Name der Person ausgegeben.

```
<ul>
  <li v-for="person in personen">
    {{ person }}
  </li>
</ul>
...
data: {
  personen: ["Marco", "Caroline", "Max"],
},
```

Listing 3-4: v-for-Direktive (In Anlehnung an Street, Passaglia, & Halliday, 2018)

Im Beispiel wird ein einfaches String-Array verwendet. Es können aber auch komplexe Objekte verwendet werden.

### v-bind-Direktive

Normale Werte innerhalb eines HTML-Tags können vereinfacht durch die Verwendung der sogenannten Moustache-Syntax, so wird die Notation mit den doppelten geschweiften Klammern `{{ }}` bezeichnet, ausgegeben werden (siehe dazu Listing 3-4). Das funktioniert aber nicht, wenn HTML-Attribute dynamisch gesetzt werden sollten. (Steyer, 2019) Hierfür kann die `v-bind`-Direktive verwendet werden.

```
<div id="app">
  
</div>
...
data: {
  imageSource: "https://source.unsplash.com/random/800x600",
},
```

*Listing 3-5: v-bind-Direktive (In Anlehnung an Street, Passaglia, & Halliday, 2018)*

### v-model-Direktive

Die `v-model`-Direktive erlaubt eine sehr einfache bidirektionale Datenbindung (Two-Way-Databinding) zwischen den Daten in der Benutzeroberfläche und dem dahinterliegenden Model (JavaScript der Vue-Instanz). (Steyer, 2019) Damit lässt sich beispielsweise, wie in nachfolgendem Listing angeführt, ohne jegliche Programmierung, der Wert eines Input-Felds live in der Anwendung darstellen. Änderungen in der Input-Box werden sofort ohne ein vollständiges Neu-Rendern der Seite übernommen.

```
<div>
  <input v-model="message" />
  <br />
  {{ message }}
</div>
...
data: {
  message: "Inhalt des Input-Felds",
},
```

*Listing 3-6: v-model-Direktive (In Anlehnung an Steyer, 2019)*

### v-on-Direktive

Für das Eventhandling in Vue.js wird die `v-on`-Direktive verwendet. In Kombination mit `v-on` können die üblichen DOM-Ereignisse wie zum Beispiel `onclick` oder `keyup` verwendet werden.

```
<div>
  <!-- Lange Form mit v-on:click -->
  <button v-on:click="meineMethode()">Klicke mich</button>
  <!-- Kurze Form mit @ anstatt v-on: -->
  <button @click="meineMethode()">Klicke mich</button>
```

```
</div>
...
methods: {
  meineMethode: function () {
    //Führe etwas aus
  },
},
```

Listing 3-7: v-on-Direktive

## 3.5 Datenschutz

Nachdem über die Web-Anwendung personenbezogene Daten erfasst, gespeichert, abgefragt, übermittelt und somit verarbeitet werden, unterliegt diese der Datenschutzgrundverordnung. Die Verordnung trat am 25. Mai 2018 in Kraft und verfolgt das Ziel, einerseits die personenbezogenen Daten natürlicher Personen, andererseits auch Personen bei der Verarbeitung dieser Daten besser zu schützen. Personenbezogene Daten sind jene Informationen, welche eine Person identifizieren oder identifizierbar machen. Dies sind zum Beispiel: Sozialversicherungsnummer, Name, Adresse oder Telefonnummer etc. Die Grundsätze für die Verarbeitung von personenbezogenen Daten gemäß Artikel 5 DSGVO sind folgende:

- **Rechtmäßigkeit:** Die Verarbeitung ist nur zulässig, wenn die betroffene Person ihre Einwilligung gegeben hat.
- **Transparenz:** Die Verarbeitungsweise muss für die betroffenen Personen nachvollziehbar sein. Informationen zur Verarbeitung müssen den Personen in transparenter, verständlicher und leicht zugänglicher Form, in einfacher Sprache zur Verfügung gestellt werden.
- **Zweckbindung:** Die Daten dürfen nur für festgelegte, eindeutige und legitime Zwecke erhoben und verarbeitet werden.
- **Datenminimierung:** Die Daten sollen dem Zweck der Verarbeitung entsprechen und auf das notwendige Maß beschränkt sein.
- **Richtigkeit:** Die Daten müssen sachlich richtig und aktuell sein. Unrichtige Daten sind unverzüglich richtigzustellen oder zu löschen.
- **Speicherbegrenzung:** Die Dauer der Speicherung von Daten in einer Form, die die Identifizierung von Personen ermöglicht, soll auf den notwendigen Verarbeitungszweck reduziert sein.
- **Integrität und Vertraulichkeit:** Die personenbezogenen Daten sind so zu verarbeiten, dass sie vor unbefugtem Zugriff, unabsichtlichem Verlust oder Schädigung geschützt sind.
- **Rechenschaftspflicht:** Die Einhaltung der Grundsätze muss vom/von der Verantwortlichen nachgewiesen werden.



Kreditnehmer/innen, die ihre personenbezogenen Daten über das Web-Portal bekannt geben, haben lt. Artikel 15 ff DSGVO ein Auskunftsrecht über die angegebenen Daten, deren genauen Verarbeitungszweck, die weiteren Empfänger/innen der Daten, die geplante Dauer der Speicherung etc. Außerdem haben die betroffenen Personen das Recht auf Berichtigung der Daten, auf Einschränkung der Verarbeitung, auf Widerspruch zur Verarbeitung und auf die Löschung der Daten.

## 4 ANFORDERUNGEN

Zur Vereinfachung des gesamten Ablaufs der Kreditvermittlung (Dokumentaustausch, Informationsbereitstellung, Angebotsvergleich usw.) für Kund/innen, Berater/innen und Kreditvermittler/innen soll der Prozess in einem Online-Portal abgebildet werden. Im Fokus stehen dabei der Komfort und die Transparenz für Kund/innen sowie die Reduktion des Zeitaufwands sowohl für Kund/innen als auch Berater/innen.

Die persönliche Beratung stellt einen sehr wichtigen Teil in der Kreditvermittlung dar. Deshalb soll das Portal die persönliche Beratung zwar unterstützen, aber auf keinen Fall ersetzen.

Im Folgenden werden die Anforderungen an das Kreditvermittlungsportal definiert. Die Definitionen wurden im Zuge von internen Workshops der Firma simplifin erarbeitet. Dabei wurden der, in Kapitel 2.1.1 beschriebene Prozess, die Unterlagen und Daten, welche die Banken fordern, sowie bereits durchgeführte Kreditabwicklungen detailliert analysiert. Bei den Anforderungen wird zwischen funktionalen und nicht-funktionalen Zielen (Qualitätszielen) differenziert.

### 4.1 Funktionale Anforderungen

Die nachfolgenden funktionalen Anforderungen definieren die Funktionen, die die Anwendung bieten soll.

#### 1. Allgemeine Anforderungen

1.1	<b>Rollen</b>	Im Prototyp sollen die beiden Rollen „Kunde/Kundin“ und „Admin“ implementiert werden. Admins sind hier zugleich Kreditvermittler/innen bzw. Berater/innen. Das System soll aber so konzipiert werden, dass zukünftig weitere Rollen, zum Beispiel für externe Berater/innen, implementiert werden können.
1.2	<b>Mehrere Kreditnehmer/innen</b>	Viele Immobilien werden nicht allein, sondern zu zweit gekauft. Deshalb gibt es bei vielen Hypothekarkrediten mehr als einen/eine Kreditnehmer/in. Das Portal muss aus diesem Grund mindestens zwei Kreditnehmer/innen unterstützen.
1.3	<b>Unterstützung der gängigsten Endgeräte</b>	Die Anwendung muss auf allen gängigen Endgeräten (Desktop-PCs, Laptop, Tablet, Smartphones) genutzt werden können.

## 2. Authentifizierung

2.1	<b>Registrierung</b>	Die Kund/innen können sich beim Portal registrieren und direkt mit der Abwicklung beginnen. Die Registrierung muss in die bestehende Homepage eingebunden werden. Im Zuge der Registrierung müssen die rechtlichen Anforderungen hinsichtlich des Datenschutzes, der Übermittlung der Standardinformationen etc. erfüllt werden. Eine doppelte Registrierung mit gleicher E-Mail-Adresse darf nicht möglich sein.
2.2	<b>Login</b>	Der Login muss anhand der E-Mail-Adresse und einem sicheren Passwort möglich sein. Hinweis: Im Zuge des Prototyps wird noch keine 2-Faktor-Authentifizierung implementiert. Das System muss aber so gestaltet werden, dass diese leicht hinzugefügt werden kann.
2.3	<b>Passwort ändern</b>	Benutzer/innen müssen eine sichere Möglichkeit haben, ihr Passwort zu ändern.
2.4	<b>Passwort vergessen</b>	Über die Passwort-Vergessen-Funktion können die Accountinhaber/innen das Passwort zurücksetzen. Der Ablauf muss gängige Sicherheitskonzepte berücksichtigen.

## 3. Dateneingabe

3.1	<b>Dateneingabe - Allgemein</b>	Benutzer/innen können direkt nach der Registrierung mit der Dateneingabe beginnen. Dabei sollen alle Daten, die für die Kreditanträge bei den Banken notwendig sind, eingegeben werden können. Die Eingabe soll in folgende Bereiche gegliedert werden: <ul style="list-style-type: none"> <li>• Persönliche Daten</li> <li>• Berufliche Daten</li> <li>• Daten zur Immobilie</li> <li>• Haushaltsrechnung</li> <li>• Angaben zur Finanzierung</li> <li>• Vermögen und Verbindlichkeiten</li> </ul>
3.2	<b>Dateneingabe - Haushaltsrechnung</b>	Die Haushaltsrechnung soll direkt im Portal durchgeführt werden können und resultierende Werte wie die Summe der Einnahmen, Ausgaben und frei verfügbares Einkommen automatisch berechnet und angezeigt werden.

<b>3.3</b>	<b>Dateneingabe - Immobilie</b>	Die notwendigen und umfangreichen Immobiliendaten sollen direkt im Portal eingegeben werden, wobei Kund/innen bei der Eingabe durch vorgegebene Auswahlmöglichkeiten bestmöglich unterstützt werden sollen.
<b>3.4</b>	<b>Dateneingabe – Unterscheidung der Immobilienart</b>	Bei der Eingabe der Daten zur Immobilie muss zwischen Grundstück, Wohnung (Eigentumswohnung und Vorsorgewohnung) und Haus (Einfamilienhaus, Mehrfamilienhaus, Reihenhaus, Doppelhaushälfte) unterschieden werden können. Je nach Art der Immobilie werden nur die notwendigen auszufüllenden Felder angezeigt.
<b>3.5</b>	<b>Dateneingabe – Daten zur Finanzierung</b>	Kund/innen können die Projektkosten und das eingebrachte Eigenkapital übersichtlich eingeben. Die Kaufnebenkosten sollen automatisch aus den Projektkosten berechnet werden, zusätzlich aber auch manuell geändert werden können.
<b>3.6</b>	<b>Finanzierungsübersicht</b>	Den Kund/innen soll eine erste Finanzierungsübersicht mit den erwarteten Kosten (Projektkosten, Kaufnebenkosten, Finanzierungsnebenkosten), dem Auszahlungsbetrag und dem Kreditbetrag angezeigt werden.
<b>3.7</b>	<b>Ergänzungen durch Berater/innen</b>	Die Berater/innen haben die Möglichkeit, Kundendaten einzugeben und zu ergänzen.
<b>3.8</b>	<b>Hinweis über Vollständigkeit der Daten</b>	Der/die persönliche Betreuer/in des/der Kunden/Kundin kontrolliert die eingegebenen Daten. Sobald die Daten überprüft und vollständig sind, wird das im Portal visualisiert.

#### 4. Dokumente

<b>4.1</b>	<b>Standardinformationen</b>	Die gesetzlich vorgeschriebenen Standardinformationen müssen dem/der Kunden/Kundin zur Verfügung gestellt werden und der Erhalt vor Beginn der Kreditvermittlung durch diesen/diese bestätigt werden.
<b>4.2</b>	<b>Kreditvermittlungsauftrag</b>	Vor Beginn der Kreditvermittlung müssen die Kund/innen den Kreditvermittlungsauftrag unterzeichnen.
<b>4.3</b>	<b>Hochladen von Dokumenten</b>	Banken fordern Belege für die angegebenen Daten. Darunter fallen zum Beispiel die Lohn- bzw. Gehaltszettel der letzten drei Monate oder Ausweisdokumente. Das Portal muss den

		Kund/innen die Möglichkeit bieten, diese Dokumente hochzuladen.
<b>4.4</b>	<b>Fotografieren von Dokumenten</b>	<p>Die erforderlichen Dokumente sollen nicht nur durch Auswahl von lokal gespeicherten Dateien hochgeladen werden können. Auf mobilen Endgeräten muss es zusätzlich die Möglichkeit geben, direkt aus dem Portal heraus, Dokumente zu fotografieren und hochzuladen.</p> <p>Hinweis: Im Prototyp kann auf die Umwandlung der fotografierten Dokumente in das PDF-Format verzichtet werden.</p>
<b>4.5</b>	<b>Anzeige der hochgeladenen Dokumente</b>	Alle Dokumente (PDF und Bilder) müssen direkt in der Anwendung angezeigt werden können. Wichtig ist diese Funktion für den/die Berater/in. Diese/r kann sich damit direkt im Portal schnell und einfach eine Übersicht über die übermittelnden Dokumente verschaffen und überprüfen, ob die notwendigen Informationen enthalten sind. Der Umweg, die Dateien zuerst herunterzuladen, um diese anschließend lokal anzuzeigen, ist nicht erwünscht.
<b>4.6</b>	<b>Automatisches Generieren bzw. Ausfüllen von Dokumenten</b>	Die Daten der Kund/innen müssen den Banken in geeigneter Form übermittelt werden. Dafür müssen PDFs aus den eingegebenen Daten anhand von Vorlagen automatisiert erzeugt werden können.
<b>4.7</b>	<b>Unterschrift der Dokumente</b>	Die Daten der automatisch generierten Dokumente müssen vom/von der Kunden/Kundin noch vor der Übermittlung an die Banken bestätigt werden. Dies sollte auf mobilen Endgeräten bzw. Geräten mit Touchscreen direkt am Bildschirm mit Stift oder Finger möglich sein. Zusätzlich muss die Möglichkeit bestehen, die Dokumente herunterzuladen, per Hand zu unterschreiben und wieder hochzuladen.
<b>4.8</b>	<b>Dokumentenvorlagen erstellen</b>	Wie unter Anforderung 4.6 beschrieben, müssen Dokumente automatisch anhand der eingegebenen Daten generiert werden und auch unterschrieben werden können. Dafür müssen im Vorfeld Dokumentenvorlagen erstellt werden. In den Vorlagen muss definiert werden, wo welche Informationen automatisch im Dokument platziert werden und wo am Dokument Unterschriften zu tätigen sind.

<b>4.9</b>	<b>Dokumente - Löschen</b>	Hochgeladene Dokumente müssen gelöscht werden können. Dokumente, die bereits an Banken übermittelt wurden, können während einer laufenden Abwicklung nicht gelöscht werden. So ist gewährleistet, dass sowohl der/die Kunde/Kundin als auch der/die Berater/in denselben Datenstand haben. Könnten Benutzer/innen deren Dokumente dennoch löschen, müssten die Daten auf Seiten des/der Beraters/Beraterin aus Gründen der Rückverfolgbarkeit weiterhin verfügbar bleiben. Das würde zu Intransparenz gegenüber den Kund/innen führen. Natürlich können Kund/innen auf Anfrage bzw. nach Abschluss oder bei Abbruch der Kreditvermittlung sämtliche Daten löschen (siehe dazu Anforderung 9.3). Auch Administrator/innen haben jederzeit die Möglichkeit, Dokumente zu löschen.
------------	----------------------------	---

## 5. Auftragsverwaltung

<b>5.1</b>	<b>Kundenübersicht</b>	In einer Übersicht müssen die wichtigsten Daten jedes/jeder Kunden/Kundin und der aktuelle Status der Kreditvermittlung dargestellt werden.
<b>5.2</b>	<b>Status der Abwicklung</b>	Der Fortschritt der Kreditvermittlung und der Vollständigkeitsgrad der eingegebenen Daten sollten automatisch ermittelt und dargestellt werden.
<b>5.3</b>	<b>Filtermöglichkeit</b>	Die Kundenübersicht muss nach dem derzeitigen Status/Fortschritt der Kreditvermittlung gefiltert werden können.
<b>5.4</b>	<b>Zuweisen eines/einer Beraters/Beraterin</b>	Simplifin hat mehrere Berater/innen, wobei jede/r nur eine begrenzte Anzahl an Kund/innen gleichzeitig betreut. Deshalb wird jedem/jeder Kunden/Kundin ein/e Berater/in zugewiesen. Für den Fall einer Urlaubsvertretung müssen einem/einer Kunden/Kundin mehrere Berater/innen zugewiesen werden können.
<b>5.5</b>	<b>Anmerkungen</b>	Im Laufe einer Kreditabwicklung erfolgt ein ständiger Austausch mit den Kund/innen in Form von persönlichen Terminen und Telefonaten. Die Informationen dieser Termine

		sollten zur Einsicht für die Berater/innen im Portal vermerkt werden können.
<b>5.6</b>	<b>Nur für Administrator/innen</b>	Der Zugang zur Auftragsverwaltung darf nur für Admins möglich sein.

## 6. Anfragen an Banken

<b>6.1</b>	<b>Anlegen einer Anfrage</b>	Admins können Anfragen anlegen. Jede Anfrage im Portal entspricht einer Kreditanfrage an eine Bank.
<b>6.2</b>	<b>Mehrere Bank-Anfragen je Kunde/Kundin</b>	Nicht jede Bank eignet sich für die Ansprüche des/der jeweiligen Kunden/Kundin. Der/die Kreditvermittler/in legt fest, welche Banken eine Anfrage erhalten. Je Kunde/Kundin sollten mehrere Banken definiert werden können.
<b>6.3</b>	<b>Unterschiedliche Dokumente je Bank</b>	Die notwendigen Unterlagen und Informationen gleichen sich zum Großteil bei den Banken. Dennoch gibt es geringe Unterschiede hinsichtlich der geforderten Unterlagen und der zu verwendenden Dokumentenvorlagen. Deshalb müssen für jede Anfrage die erforderlichen bzw. zu übermittelnden Dokumente individuell festgelegt werden können. Das Erstellen der Dokumente mit den Kundendaten muss, wie unter Anforderung 4.6 definiert, automatisch erfolgen.
<b>6.4</b>	<b>Kontaktadresse je Anfrage</b>	Jede Anfrage wird an eine andere Bank adressiert. Deshalb müssen die Kontaktdaten des/der jeweiligen Partnerbetreuers/-betreuerin der Bank hinterlegt werden können (E-Mail, Name).
<b>6.5</b>	<b>Übermittlung an Banken</b>	Die in der Anfrage definierten Unterlagen müssen direkt im Portal per E-Mail an die Bank (den/die Betreuer/in bei der Bank) übermittelt werden können.
<b>6.6</b>	<b>Verlauf Anfragen an Banken</b>	Der E-Mail-Verlauf der gesendeten Anfragen an die Banken muss inklusive der übermittelten Daten und Dateien im Portal ersichtlich sein.
<b>6.7</b>	<b>Vorlagen für Anfragen</b>	Es gibt einen Pool an Banken, mit denen zusammengearbeitet wird. Für diese Banken sollten Vorlagen gespeichert werden können. Anhand dieser Vorlagen werden die notwendigen Dokumente, wie unter Anforderung 6.3, beschrieben, den Kund/innen automatisch zur Verfügung gestellt und müssen somit nicht jedes Mal von

		Neuem ausgewählt werden. Zudem werden die Kontaktdaten der Bank in der Vorlage gespeichert und können bei der nächsten Anfrage wiederverwendet werden.
<b>6.8</b>	<b>Anfragen löschen</b>	Anfragen müssen im Falle eines Fehlers durch den/die Berater/in beim Anlegen gelöscht werden können.
<b>6.9</b>	<b>Anfragen nur für Berater/innen</b>	Anfragen können nur von Beratern/Beraterinnen (bzw. im Prototyp gleichen diese den Admins) angelegt, geändert und gelöscht werden.

## 7. Angebote der Banken

<b>7.1</b>	<b>Jeder Anfrage können Angebote zugeordnet werden</b>	Je Anfrage müssen die erhaltenen Angebote der jeweiligen Bank angelegt werden können. Dabei sollen die wichtigsten Daten des Angebots als Werte eingegeben werden können und Dokumente zu jedem Angebot (betrifft insbesondere das gesetzlich vorgeschriebene ESIS-Merkblatt) angehängt werden können.
<b>7.2</b>	<b>Mehrere Angebote je Anfrage</b>	<p>Je Anfrage können mehrere Angebote angelegt werden. Das ist aus den folgenden Gründen notwendig:</p> <ul style="list-style-type: none"> <li>• Mehrere Finanzierungsvarianten: Je Bank (und damit je Anfrage) können verschiedene Finanzierungsvarianten angefragt werden, zum Beispiel eine Variante mit variabler Verzinsung und eine Variante mit fixer Verzinsung oder unterschiedlichen Laufzeiten.</li> <li>• Nachverhandlungen: Bei Nachbesserungen der Angebote aufgrund von Nachverhandlungen mit den Banken erhält man neue Angebote.</li> </ul>
<b>7.3</b>	<b>Freigabe der Angebote</b>	Das Angebot muss vom/von der Kreditvermittler/in manuell freigegeben werden, bevor es den Kund/innen im Angebotsvergleich zur Verfügung gestellt wird. Dadurch können Angebote, die aufgrund von Nachverhandlungen, veraltet sind, unsichtbar gemacht werden.
<b>7.4</b>	<b>Angebotsvergleich</b>	<p>Die wichtigsten Informationen der Angebote müssen in einem Angebotsvergleich übersichtlich dargestellt werden. Dieser Vergleich muss sowohl den Kund/innen als auch den Beratern/Beraterinnen zugänglich sein.</p> <p>Im Angebotsvergleich dürfen nur durch den/die Berater/in freigegebene Angebote angezeigt werden.</p>



## 8. Chat-Funktion

8.1	<b>Chat-Funktion</b>	Es soll eine Chat-Funktion implementiert werden, damit sich Kund/innen bei Fragen mit ihrem/ihrer Betreuer/in austauschen können. Im Chat müssen neben Texten auch Dateien ausgetauscht werden können. Die Benutzererfahrung soll gängigen Chats wie zum Beispiel WhatsApp gleichen.
8.2	<b>Benachrichtigung für Berater/innen</b>	Die Berater/innen müssen über neue Chat-Nachrichten in der Kundenübersicht benachrichtigt werden. Dabei dürfen nur dem/der jeweiligen Kunden/Kundin zugewiesene Berater/innen über neue Nachrichten informiert werden.

## 9. Rechtlich & Datenschutz

9.1	<b>Einhaltung der rechtlichen Rahmenbedingungen</b>	Einhaltung der rechtlichen Anforderungen und Informationspflichten laut Hypothekar- und Immobilienkreditgesetz (HlKrG), Verbraucherkreditgesetz (VKrG), Standesregeln für Kreditvermittlung, Fernfinanzdienstleistungsgesetz und Datenschutzgrundverordnung.
9.2	<b>Impressum und Datenschutzerklärung</b>	Das Impressum und die Datenschutzerklärung müssen im gesamten Portal jederzeit erreichbar/aufrufbar sein.
9.3	<b>Löschen aller Daten</b>	Nach Abschluss der Kreditvermittlung, nach Aufforderung durch den/die Kunden/Kundin oder nach einer Inaktivität des/der Kunden/Kundin von mehr als 60 Tagen sollen die gesamten Daten aus sicherheits- und datenschutzrechtlichen Gründen gelöscht werden. Das Löschen nach einer längeren Inaktivität kann manuell erfolgen.

## 4.2 Nicht-funktionale Anforderungen bzw. Qualitätsziele

Die ISO/IEC 25010:2011-03 „Software-Engineering - Qualitätskriterien und Bewertung von Softwareprodukten (SQuaRE) - Qualitätsmodell und Leitlinien“ definiert ein Qualitätsmodell für Softwareprodukte, welche Qualitätseigenschaften in acht Gruppen mit weiteren Unterpunkten unterteilt.

Nachfolgende Auflistung basiert auf der ISO/IEC 25010:2011-03<sup>3</sup> und beschreibt die darin definierten Qualitätsziele. Die letzte Spalte „Prio“ gibt an, wie wichtig dieses Qualitätsziel im Hinblick auf den Prototypen des Webportals ist (1 = sehr wichtig bis 5 = unwichtig).

Qualitätsziel	Beschreibung	Prio
<b>Funktionale Eignung</b>	Gibt an, wie gut das System die definierten Funktionen abbildet, wenn es unter festgelegten Bedingungen benutzt wird. (ISO/IEC 25010, 2011)	<b>1</b>
Vollständigkeit	Grad, mit dem die spezifizierten Aufgaben und Nutzerziele abgedeckt werden. (ISO/IEC 25010, 2011)	1
Korrektheit	Es müssen die richtigen Ergebnisse mit der erforderlichen Genauigkeit geliefert werden (ISO/IEC 25010, 2011)	1
Angemessenheit	Die Funktionen müssen zu den definierten Anforderungen passen. Es sollten nur die wirklich notwendigen Funktionen angezeigt werden, um eine bestimmte Aufgabe zu erledigen. (ISO/IEC 25010, 2011)	3
<b>Leistungseffizienz</b>	Die Anwendung muss unter festgelegten Bedingungen mit den eingesetzten Ressourcen ein angemessenes Leistungsniveau erreichen. (Balzert, 2009)	<b>3</b>
Zeitverhalten	Die Antwort- und Verarbeitungszeiten sowie die Durchsatzraten müssen in einem angemessenen Rahmen bleiben. (ISO/IEC 25010, 2011)	3
Ressourcenverbrauch	Grad, zu dem die vom System genutzten Ressourcen den Anforderungen entsprechen. Laut ISO/IEC 25010:2011-3 zählen hierzu auch Humanressourcen.	2
Kapazität	Grad, zu dem die definierten Leistungsgrenzen (gleichzeitige Benutzer/innen, Bandbreite, Datenbankgröße usw.) des Systems eingehalten werden. (ISO/IEC 25010, 2011)	3
<b>Kompatibilität</b>	Beschreibt die Fähigkeit des Systems, Informationen mit anderen Systemen auszutauschen und/oder die geforderten Funktionen auszuführen, während es die	<b>5</b>

<sup>3</sup> Es gibt keine offizielle deutsche Version der ISO/IEC 25010:2011-03, deshalb wurden die Begriffe und Beschreibungen teilweise aus der englischen Version übersetzt.

	gleiche Hardware- oder Softwareumgebung mit anderen Systemen teilt. (ISO/IEC 25010, 2011)	
Koexistenz	Die Software kann gemeinsam mit anderen Produkten in einer gemeinsamen Umgebung betrieben werden, ohne dabei negative Auswirkungen auf die anderen Produkte zu haben. (ISO/IEC 25010, 2011)	5
Interoperabilität	Beschreibt die Fähigkeit der Software, mit weiteren vorgegebenen Systemen zusammenzuarbeiten. (Balzert, 2009)	4
<b>Benutzbarkeit</b>	Laut Balzert (2009) beschreibt die Benutzbarkeit die Fähigkeit der Software, von Benutzern/Benutzerinnen verstanden und verwendet zu werden und erlernbar sowie attraktiv zu sein.	<b>1</b>
Angemessenheit Erkennbarkeit	Benutzer/innen müssen erkennen können, ob die Software für ihre gewünschten Aufgaben angemessen ist bzw. für ihre Anforderungen verwendet werden kann. (Balzert, 2009)	2
Erlernbarkeit	Eigenschaft der Software, die Benutzer/innen ermöglicht, die effektive, effiziente, risikofreie und zufriedenstellende Benutzung der Anwendung zu erlernen. (ISO/IEC 25010, 2011)	2
Bedienbarkeit	Die Anwendung hat Eigenschaften, die sicherstellen, dass diese einfach auszuführen und zu bedienen ist. (ISO/IEC 25010, 2011)	1
Schutz vor Benutzerfehlern	Die Software schützt den/die Benutzer/in vor Bedienungsfehlern. (ISO/IEC 25010, 2011)	1
Ästhetik der Benutzer- oberfläche	Die Benutzeroberfläche wird so umgesetzt, dass die Interaktion für den/die Benutzer/in angenehm und zufriedenstellend ist, zum Beispiel durch die Farbgebung oder die Art des grafischen Designs. (ISO/IEC 25010, 2011)	2
Barrierefreiheit	Barrierefreiheit bedeutet, dass die Anwendung auch für Menschen mit körperlichen Einschränkungen oder geringeren technischen Möglichkeiten bedienbar ist. Körperliche Einschränkungen betreffen insbesondere Sehbehinderungen, Taubheit oder Störungen der Feinmotorik. (Walter, 2008)	5

<b>Zuverlässigkeit</b>	Das System kann die definierten Funktionen für eine bestimmte Zeit auf einem bestimmten Leistungsniveau gewährleisten. (ISO/IEC 25010, 2011)	<b>2</b>
Reifegrad	Beschreibt den Fähigkeitsgrad der Software trotz Auftretens von Fehlern nicht zu versagen. (Balzert, 2009)	2
Verfügbarkeit	Grad, zu dem das System betriebsbereit und zugänglich ist, sofern eine Benutzung erforderlich ist. (ISO/IEC 25010, 2011)	2
Fehlertoleranz	Die Software bleibt auch bei Fehlbedienung oder bei Softwarefehlern lauffähig. (Balzert, 2009)	2
Wiederherstellbarkeit	Die Software ist im Falle eines Fehlers schnell wieder lauffähig und die Daten können wiederhergestellt werden. (Balzert, 2009)	2
<b>Sicherheit</b>	Die Anwendung schützt die Informationen und Daten der Benutzer/innen und stellt sicher, dass Benutzer/innen nur auf Daten, für die sie die Berechtigung haben, zugreifen können. (ISO/IEC 25010, 2011)	<b>1</b>
Vertraulichkeit	Es ist sichergestellt, dass die Daten nur jenen Personen zugänglich sind, die für den Zugriff autorisiert sind. (ISO/IEC 25010, 2011)	1
Integrität	Der unbefugte Zugriff auf oder die Änderung von Computerprogrammen und Daten wird verhindert. (ISO/IEC 25010, 2011)	1
Nachweisbarkeit	Das Stattfinden von Aktionen oder Ereignissen muss zweifelsfrei nachgewiesen werden können. (ISO/IEC 25010, 2011)	2
Verantwortlichkeit	Die Aktionen eines/einer Benutzers/Benutzerin müssen diesem/dieser eindeutig zugeordnet werden können. (ISO/IEC 25010, 2011)	2
Authentizität	Die vorgegebene Identität einer Person oder Ressource entspricht deren wahrer Identität. (ISO/IEC 25010, 2011)	1
<b>Wartbarkeit</b>	Beschreibt die Fähigkeit einer Software, offen für Änderungen (Verbesserungen, Anpassungen, Korrekturen, Änderungen der Umgebung) zu sein. (Balzert, 2009)	<b>4</b>

Modularität	Das System besteht aus möglichst unabhängigen Komponenten, die ohne große Auswirkungen auf andere Komponenten ausgetauscht werden können. (ISO/IEC 25010, 2011)	3
Wiederverwendbarkeit	Das System oder Teile des Systems kann bzw. können in anderen Systemen wiederverwendet werden. (ISO/IEC 25010, 2011)	4
Analysierbarkeit	Zum einen sollten Fehler und deren Ursachen leicht diagnostiziert werden können und zum anderen sollten die Auswirkungen von geplanten Änderungen gut analysierbar sein. (ISO/IEC 25010, 2011)	4
Veränderbarkeit	Änderungen an Teilen der Software verursachen keine unerwünschten Fehler bzw. verringern nicht die bestehende Softwarequalität. (ISO/IEC 25010, 2011)	3
Testbarkeit	Beschreibt die Fähigkeit, Testkriterien für das System, die Software oder Komponenten zu definieren und Tests durchzuführen, um damit festzustellen, ob die definierten Kriterien erfüllt werden. (ISO/IEC 25010, 2011)	3
<b>Portabilität</b>	Das System kann von einer Software-, Hardware-, Betriebs-, oder Nutzungsumgebung in eine andere transferiert werden. (ISO/IEC 25010, 2011)	<b>4</b>
Anpassbarkeit	Beschreibt die Möglichkeit, das System an eine andere Hardware-, Software-, Betriebs-, oder Nutzungsumgebung anzupassen. Darunter fällt auch die Skalierbarkeit der Software bei höherer Nutzung oder auch die Individualisierbarkeit der Software durch den/die Kunden/Kundin. (ISO/IEC 25010, 2011)	3
Installierbarkeit	Gibt an, wie aufwändig die Anwendung in einer bestimmten Umgebung installiert oder deinstalliert werden kann. (ISO/IEC 25010, 2011)	3
Austauschbarkeit	Definiert, wie leicht die Software ein vergleichbares Produkt in der gleichen Umgebung ersetzen kann. (ISO/IEC 25010, 2011)	5

Tabelle 4-1: Qualitätsziele von Softwareprodukten (In Anlehnung an ISO/IEC 25010, 2011)

Aus Tabelle 4-1 geht hervor, dass die drei wichtigsten Qualitätsziele die funktionale Eignung, die Benutzbarkeit und die Sicherheit sind.

Die **Funktionalität** hat, da es sich um einen Prototyp handelt, naturgemäß eine sehr hohe Priorität.

Bei den Daten, die von den Kund/innen zur Verfügung gestellt werden, handelt es sich um personenbezogene (Bank-)Daten. Die **Sicherheit** der Anwendung hat deshalb höchste Priorität.

Nur wenn die Anwendung einfach zu bedienen ist und ein gutes Benutzererlebnis bietet, werden Kund/innen bereit sein, die Software zu verwenden. Die Benutzer/innen sind teilweise wenig IT-affin und werden nicht für die Bedienung der Software geschult. Deshalb muss das System leicht erlernbar sein und Kund/innen müssen vor Fehleingaben bzw. Fehlbedienung geschützt werden. Die **Benutzbarkeit** wird aus diesen Gründen als sehr wichtig eingestuft.

Das System sollte auch in der Prototypenphase jederzeit verfügbar sein, weshalb die **Zuverlässigkeit** direkt nach den drei vorhin genannten Qualitätszielen folgt.

Im Zuge der Prototypenentwicklung muss kein Fokus auf die **Wartbarkeit** gelegt werden. Die Entwicklung des Prototyps erfolgt nur durch eine/n einzelne/n Entwickler/in, der/die die Anwendung und deren interne Abhängigkeiten sehr gut kennt. Deshalb können auch die Auswirkungen von Änderungen gut abgeschätzt werden. Dennoch sollten die einzelnen Teile der Anwendung möglichst modular gestaltet werden, um Änderungen schnell durchführen zu können. Bei der **Testbarkeit** soll der Fokus auf sicherheitskritische Aspekte wie die Authentifizierung und das Daten- und Dokumentenhandling gelegt werden.

Da es sich im ersten Schritt um einen funktionalen Prototyp handelt, wird weniger Priorität auf die **Leistungseffizienz, Kompatibilität** und **Portabilität** des Systems gelegt. Die Benutzeranzahl wird zu Beginn noch sehr niedrig sein, weshalb auch mit stark begrenzten Ressourcen und nur geringen Optimierungen der Anwendung keine Leistungsentpässe zu erwarten sind. Der Prototyp muss mit keinen anderen Systemen außer einem Registrierungsformular auf der Webseite interagieren. Deshalb spielt die **Kompatibilität** für den Prototypen so gut wie keine Rolle. Hinsichtlich der **Portabilität** sollte die zukünftige Skalierbarkeit des Systems (Anpassbarkeit) berücksichtigt werden.

## 5 KONZEPTENTWICKLUNG

Auf Basis der Anforderungen wurde ein Konzept für das Webportal entwickelt. Ein sehr zeitintensiver Teil der Konzeptentwicklung war die Erstellung von Mockups für alle Teile der Anwendung. Mit Hilfe der Mockups kann sichergestellt werden, dass alle geforderten Funktionen von der Anwendung abgebildet werden und bestmöglich nutzbar sind. Der Fokus der Mockups liegt auf einer guten Benutzererfahrung und somit dem Layout und weniger auf einem perfekten Design.

In Kapitel 6 werden die wichtigsten Teile der Anwendung anhand von Screenshots erklärt. Nachdem diese Screenshots somit alle Informationen der Mockups sowie zusätzliche Details zeigen, wird auf die Darstellung und Auflistung der erstellten Mockups verzichtet.

### 5.1 Anwendungsarchitektur

Als Grundlage für die gewählte Anwendungsarchitektur dienen die Anforderungen, eine möglichst gute Skalierbarkeit bei gleichzeitig geringen Anlaufkosten sowie die Erfahrung des Entwicklers mit den gewählten Technologien. Ein weiterer Aspekt sind datenschutzrechtliche Anforderungen hinsichtlich der Datenspeicherung ausschließlich in der Europäischen Union.

#### 5.1.1 Systemarchitektur

Als Programmiersprache kommt sowohl client- als auch serverseitig JavaScript zum Einsatz. Dabei werden clientseitig die Frameworks Vue.js und Quasar und serverseitig das auf Node.js basierende Feathers.js Framework verwendet.

Die gesamte Anwendung wird in der Microsoft Azure Cloud veröffentlicht. Dabei werden sämtliche Daten auf Microsoft-Servern in Deutschland (Frankfurt am Main) gespeichert.

Für das Frontend und Backend wird hierfür Azure App Service<sup>4</sup> verwendet. Zur Speicherung der Kundendaten wird Azure Cosmos DB genutzt. Dabei handelt es sich um eine NoSQL-Datenbank, die eine zu MongoDB kompatible Schnittstelle bietet. Das erleichtert die Anbindung der Datenbank, da für Feathers.js ein Datenbankadapter für MongoDB zur Verfügung steht.<sup>5</sup>

Die Dateien und Dokumente werden nicht als Binärdaten (BSON) in der MongoDB-basierten Datenbank gespeichert, sondern in einem separaten Azure Blob Storage<sup>6</sup>. Die Gründe dafür sind die bessere Kontrolle über die Zugriffsberechtigungen auf den Blob Storage sowie die bei BSON auf typischerweise 4 MB limitierte Dateigröße. Für jede/n Kundin/Kunden wird ein separater Container mit der eindeutigen Benutzer-ID angelegt. So werden Dateien von Kund/innen immer

---

<sup>4</sup> Für weitere Informationen zu Azure App Service siehe Kapitel 5.2

<sup>5</sup> Datenbankadapter für die Anbindung einer MongoDB in Feathers.js: <https://github.com/feathersjs-ecosystem/feathers-mongodb>

<sup>6</sup> Für weitere Informationen zu Azure Blob Storage siehe Kapitel 5.2

nur in ihren eigenen Container gespeichert, um die Kund/innen klar zu separieren und damit die Sicherheit zu erhöhen.

**Blobcontainer**

Container nach Präfix durchsuchen  Nur aktive Container anzeigen

Alle 11 Elemente werden angezeigt.

Name	Letzte Änderung	Öffentliche Zugriffsebe...	Leasezustand
Slogs	13.3.2021, 20:08:59	Privat	Available
60575ac465953e002...	21.3.2021, 15:58:23	Privat	Available
606eb17e9bf6a1002...	8.4.2021, 20:43:34	Privat	Available
607336709bf6a1002...	11.4.2021, 19:49:50	Privat	Available

Abbildung 5-1: Azure Blob Storage – Separate Container je Kunde/Kundin (Eigene Abbildung)

Die Benutzeroberflächen und Funktionen für Administrator/innen bzw. Kreditvermittler/innen und Kund/innen unterscheiden sich nur wenig. Deshalb wird für beide Benutzergruppen ein gemeinsames Frontend verwendet. Die Kommunikation erfolgt dabei über https. Die Registrierung der Kund/innen erfolgt über ein Kontaktformular, das in die bestehende Webseite integriert wird und über eine Rest-API an das Backend des Web-Ports angebunden ist. Zur Unterstützung bei der Fehleranalyse und dem Monitoring aller Services der Anwendung kommt Microsoft Application Insights<sup>7</sup> zum Einsatz.

Die nachfolgende Abbildung 5-2 zeigt eine vereinfachte Übersicht der Systemarchitektur.

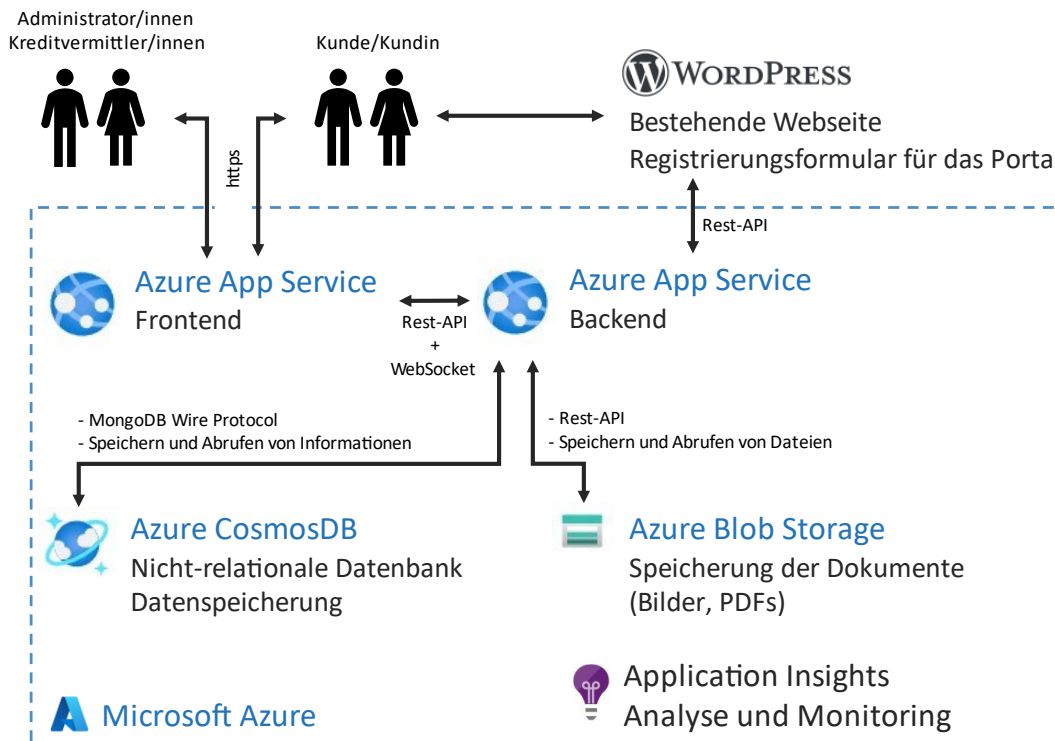


Abbildung 5-2: Systemarchitektur (Eigene Abbildung)

<sup>7</sup> Für weitere Infos zu Microsoft Application Insights siehe Kapitel 5.2



### 5.1.2 Datenbankstruktur

Zur Speicherung der Daten wird die nicht-relationale Microsoft Azure Cosmos DB verwendet. Die Daten sind auf sechs Collections aufgeteilt (siehe nachfolgende Abbildung 5-3).

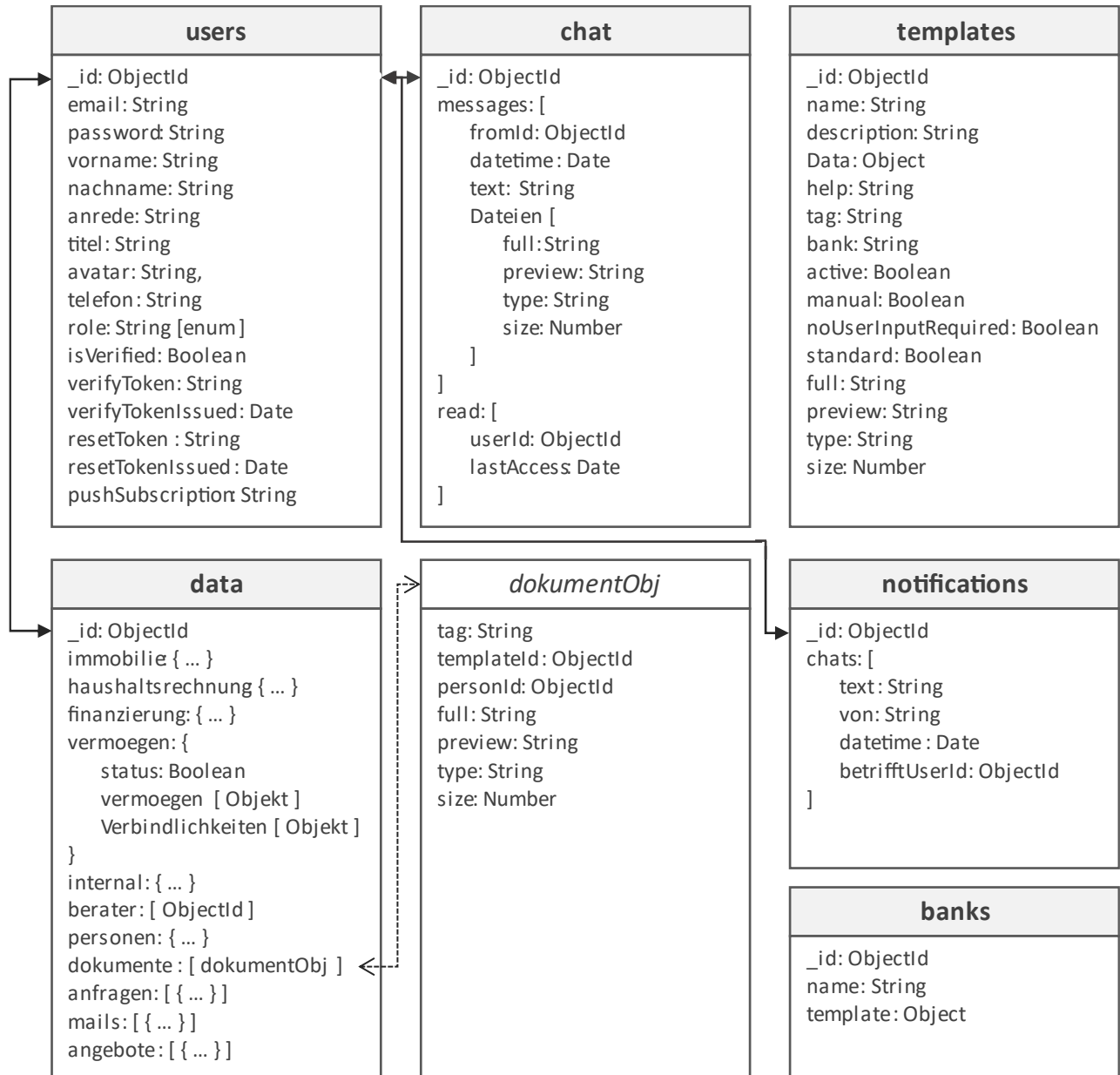


Abbildung 5-3: Datenbankstruktur (Eigene Abbildung)

In der **users**-Collection werden die Kund/innen nach der Registrierung angelegt. Diese Collection wird zudem für die Speicherung der Authentifizierungsdaten (Passwort als Hash, Account-Verifizierungstoken, Token für das Zurücksetzen des Passworts) verwendet. Die Benutzer-ID wird bei der Registrierung vergeben. In den weiteren Collections, in denen benutzerbezogene Daten gespeichert werden, wird jeweils ein Dokument mit dieser Benutzer-ID angelegt. Somit muss in jeder Collection nur nach der Benutzer-ID gefiltert werden, um sämtliche Daten eines/einer Kunden/Kundin zu finden. Zusätzlich sind die Daten der Kund/innen dadurch, obwohl

sie in der gleichen Datenbank und Collection gespeichert sind, stark voneinander separiert, was wiederum die Sicherheit erhöht.

Der Großteil der Daten wird in der **data**-Collection gespeichert. Jede/r Kunde/Kundin erhält dort ein Dokument mit der gleichen ID wie die Benutzer-ID in der users-Collection. Sämtliche Daten werden innerhalb dieses einen Dokuments gespeichert. Die Daten in diesem Dokument werden unter anderem nach den Hauptbereichen (Immobilie, Haushaltsrechnung, Finanzierung, Vermögen (und Verbindlichkeiten), Anfragen und Angebote) der Anwendung strukturiert. Die Objekte dieser Bereiche beinhalten jeweils alle Daten aller Eingabefelder dieses Anwendungsbereichs. Aufgrund der Vielzahl an Parametern/Eingabefeldern jedes Bereichs wird auf die Auflistung der einzelnen Werte in Abbildung 5-3 verzichtet. Jedes Dokument, das die Kund/innen hochladen oder unterzeichnen, wird im *dokumente*-Array der data-Collection gespeichert. Das in Abbildung 5-3 dargestellte *dokumentObj* zeigt den Aufbau eines solchen Objekts. Es handelt sich dabei um keine eigene Collection, sondern die Objektdefinition innerhalb des *dokumente*-Arrays der data-Collection.

In der Collection **chat** werden sämtliche Chat-Nachrichten gespeichert. Wie auch in der data-Collection wird für jede/n Kunden/Kundin ein einzelnes Dokument mit der Benutzer-ID angelegt. Alle Chat-Nachrichten des/der jeweiligen Kunden/Kundin werden in seinem/ihrem eigenen Dokument im message-Array gespeichert. Im read-Array wird für jede/n Benutzer/in gespeichert, wann der letzte Zugriff erfolgte. Damit lässt sich feststellen, ob ein/e bestimmte/r Benutzer/in die gesendete Nachricht bereits gelesen hat.

Kreditvermittler/innen und Kund/innen erhalten in der Anwendung Benachrichtigungen über neue Chat-Nachrichten. Zum Speichern dieser Benachrichtigungen wird die **notifications**-Collection verwendet.

Im Zuge einer Kreditanfrage an eine Bank durch eine/n Kreditvermittler/in muss eine Vielzahl an Parametern festgelegt werden. Damit diese Parameter nicht jedes Mal erneut eingegeben werden müssen, können Vorlagen für die Anfragen an Banken gespeichert werden. Diese Vorlagen werden in der **banks**-Collection gespeichert.

Das Webportal bietet die Möglichkeit, PDF-Formulare zu erstellen, die automatisch mit Kundendaten befüllt werden und als PDF gespeichert werden. Die Vorlagen für diese PDF-Formulare werden in der **templates**-Collection gespeichert.

## 5.2 Überblick über die verwendeten Technologien

Die nachfolgende Auflistung gibt einen Überblick über die wichtigsten verwendeten Technologien, Frameworks und Libraries des Prototyps:

### Frontend:

- **Vue.js:** Vue.js ist ein leichtgewichtiges Framework, um interaktive Benutzeroberflächen zu realisieren. Weitere Informationen siehe Kapitel 3.4.

- **Quasar:** Quasar ist ein auf Vue.js basierendes Frontend Framework. Damit lassen sich sehr schnell Benutzeroberflächen mit Hilfe von zahlreichen vorgefertigten Komponenten wie Textfelder, Buttons, Cards, Listen usw. gestalten. Dabei stellt Quasar aber nicht nur eine sehr umfangreiche UI-Komponenten-Bibliothek bereit, sondern bietet darüber hinaus weitere Funktionen wie zum Beispiel Single Page Apps, Unterstützung bei der Umsetzung von Progressive-Web-Apps oder Multi-Plattform-Desktop-Apps durch die Verwendung von Electron. (Stoenescu, 2021)
- **PDF.js<sup>8</sup>:** PDF.js ist „eine allgemeine, auf Webstandards basierende Plattform zum Parsen und Rendern von PDFs“, die unter der Apache-Open-Source Lizenz veröffentlicht wird. (Mozilla , 2021) PDF.js wird zum Anzeigen der PDFs in der Webanwendung verwendet.
- **Fabric.js<sup>9</sup>:** Bei Fabric.js handelt es sich um eine Canvas-Bibliothek, die das Arbeiten mit dem Canvas-Element erleichtert. Sie stellt ein Objekt-Modell und Funktionen für das Hinzufügen und Manipulieren von Textboxen und Formen bereit. Fabric.js bietet auch die Möglichkeit der Serialisierung und Deserialisierung von Objekten, die auf dem Canvas platziert wurden, als JSON. (Fabric.js Javascript Canvas Library, 2021) Diese Library wird für den Vorlagen-Editor und für das Befüllen der Vorlagen mit den Daten der Kund/innen sowie für das Darstellen der Unterschrift am PDF verwendet.
- **jsPDF<sup>10</sup>:** Dabei handelt es sich um eine clientseitige Library, um PDF-Dateien zu erzeugen. (Parallax Agency Ltd, 2021) Im Portal wird sie dazu verwendet, um die Dokumente, nachdem diese automatisch mit Kundendaten befüllt wurden, als PDF zu speichern.
- **deep-object-diff<sup>11</sup>:** Mit Hilfe der Library deep-object-diff lassen sich die Unterschiede zwischen zwei JavaScript-Objekten ermitteln. Die Library wird verwendet, um vor dem Speichern festzustellen, welche Daten sich geändert haben. Dadurch werden nur jene Daten an den Server übermittelt, die sich geändert haben.

### Backend:

- **Node.js:** Node.js ermöglicht die serverseitige Ausführung von JavaScript. Als Basis dient die gleiche JavaScript-Engine, die auch der Google Chrome Browser verwendet. (Liel, 2019) Die von Natur aus asynchrone Ausführung des Codes durch JavaScript gemeinsam mit dem leichtgewichtigen Design von Node.js erlaubt die Programmierung von stark skalierbaren Anwendungen. (Shute, 2019) Durch die Verwendung von Node.js am Server kann sowohl clientseitig als auch serverseitig, und somit in der gesamten Anwendung, JavaScript als Programmiersprache verwendet werden.

---

<sup>8</sup> PDF.js: <https://mozilla.github.io/pdf.js/>

<sup>9</sup> Fabric.js: <http://fabricjs.com>

<sup>10</sup> jsPDF: <https://github.com/parallax/jsPDF>

<sup>11</sup> deep-object-diff: <https://github.com/mattphillips/deep-object-diff>

- **FeathersJS:** FeathersJS ist ein leichtgewichtiges Web-Framework, um sehr schnell und einfach ein voll funktionsfähiges Backend inklusive Datenbankanbindung zu erstellen. Es unterstützt eine Vielzahl an unterschiedlichen Datenbanken und bietet sowohl Socket.io als auch eine Rest-API als Schnittstellen an. (Luecke, 2021)
- **Nodemailer:** Bei Nodemailer handelt es sich um ein Modul für Node.js-Anwendungen, mit dem E-Mail versendet werden können, das unter der MIT-Lizenz veröffentlicht wird. Es erlaubt das Senden von HTML- und Rein-Text-Nachrichten und bietet Unterstützung für Datei-Anhänge. (Reinman, 2021) Mit Nodemailer werden sämtliche Mails des Webportals (z.B. Registrierung, Passwortänderung, Mails an Banken usw.) versendet.
- **Microsoft Azure Cosmos DB:** Dabei handelt es sich um eine, in der Microsoft Azure Cloud gehostete, sehr schnelle nicht-relationale NoSQL-Datenbank. Microsoft garantiert Antwortzeiten in weniger als 10 ms und eine Verfügbarkeit von 99,999 %. Die Datenbank lässt sich leicht skalieren und bietet eine zu MongoDB kompatible Open-Source-API. (Microsoft Corporation, 2021). Diese Datenbank wird zur Speicherung der Benutzerdaten, aber nicht zur Speicherung der Dateien verwendet.
- **Microsoft Azure Blob Storage:** Azure Blob Storage bietet die Möglichkeit, Objekte bzw. Dateien als Blob (Binary Large Objects) in der Azure Cloud zu speichern. Es bietet Zugriff über http(s) bzw. eine Rest-API. (Microsoft Corporation, 2021) Der Blob Storage wird für das Speichern der Dokumente der Kund/innen und Vorlagen verwendet.
- **Microsoft Azure App Service:** Mit Microsoft Azure App Service lassen sich http-basierte Webanwendungen, Rest-APIs und Backends in der Azure Cloud hosten. Dabei wird eine Vielzahl an Programmiersprachen wie zum Beispiel .NET, Java, Ruby, Node.js, Php oder Python unterstützt. Mit Hilfe von Azure App Service lassen sich Anwendungen sehr einfach und automatisch skalieren. (Microsoft Corporation, 2021) Zudem lässt sich der Ort, an dem die Kundendaten gespeichert werden, auswählen. Das ist insbesondere im Hinblick auf die Einhaltung der DSGVO trotz der Nutzung von Cloud-Diensten sehr vorteilhaft.
- **Microsoft Application Insights:** Für das Monitoring der Anwendung wird Application Insights verwendet. Es überwacht alle verwendeten Services in der Azure Cloud, erkennt selbstständig Anomalien und protokolliert sowohl client- als auch serverseitige Fehler. Dabei wird eine Vielzahl an Parametern wie auszugsweise Request-Rates, Antwortzeiten, Fehleranzahl und Session-Anzahl kontinuierlich überwacht. Mit Hilfe von Alerts lassen sich benutzerdefinierte Alarmschwellen festlegen und E-Mail-Benachrichtigungen bei Überschreitung dieser Schwellen einrichten. Zur einfachen Ursachenforschung bei Fehlern bietet Application Insights die Möglichkeit, die gesamte Kette der Requests einer Benutzersession nachzuvollziehen. Zudem lässt es sich sehr einfach in die Anwendung integrieren. (Microsoft Corporation, 2021)

## 6 IMPLEMENTIERUNG

Auf Basis der Anforderungen, der Konzeptentwicklung und den Mockups wurde die Anwendung implementiert. In diesem Kapitel werden die wichtigsten Teile des Webportals vorgestellt und erklärt.

### 6.1 Zentraler Data-Store

Wie in Kapitel 3.4.1 erwähnt, können Daten mit Hilfe von Properties (props) von übergeordneten Komponenten (Parent-Components) an untergeordnete Komponenten (Child-Components) übergeben werden. In größeren Anwendungen steigt die Anzahl der Komponenten sehr schnell an. Umso mehr Komponenten in einer Anwendung verwendet werden und umso häufiger verschiedene Komponenten auf die gleichen Daten zugreifen und diese verändern, desto schwieriger wird es, die Änderungen bzw. Daten über die gesamte Anwendung hinweg synchron zu halten.

Um mit dieser Komplexität umgehen zu können, bietet es sich an, alle Daten in einem zentralen Datenspeicher vorzuhalten. Alle Komponenten können direkt auf diesen zentralen Speicher zugreifen. Somit entfällt die Notwendigkeit, sämtliche Daten über mehrere Ebenen bzw. Komponenten hinweg mit Hilfe von Properties weiterzureichen. Es gibt eine gemeinsame, zentrale „Datenwahrheit“.

Im Prototyp wird auf die Verwendung der Vuex-Library<sup>12</sup> zur Umsetzung dieses zentralen Data-Stores verzichtet. Stattdessen wird, als einfachere Variante, eine weitere Vue-Instanz als zentrales Speicherobjekt verwendet. Durch die Verwendung einer Vue-Instanz als Objekt können alle Funktionen wie Reaktivität, Methoden, Computed Properties, Watcher usw. von Vue.js auch innerhalb dieses zentralen Data-Stores verwendet werden. Dieses zentrale Objekt zur Datenspeicherung wird der Vue-Instanz der gesamten Anwendung als Instance-Property<sup>13</sup> hinzugefügt. Dadurch steht das Objekt in allen Komponenten sofort zur Verfügung, ohne dass es jedes Mal importiert bzw. geladen werden muss (siehe nachfolgendes Listing 6-1).

```
//Hinzufügen des Stores zur Vue-Instanz  
import Vue from 'vue';  
import MainStore from './stores/mainStore';  
Vue.prototype.$Store = MainStore;  
  
//Zugriff auf den Store in der Anwendung  
this.$Store.Eigenschaft = 'Max Mustermann'
```

Listing 6-1: Store-Objekt global zur Verfügung stellen

<sup>12</sup> Vuex ist eine State-Management-Library für Vue.js. Es dient als zentraler Datenspeicher mit einer Reihe von Regeln, damit Daten nur auf vorhersagbare Weise verändert werden können. Es ist ähnlich aufgebaut wie das bekanntere Redux. (What is Vuex.js | Vuex, 2021) <https://vuex.vuejs.org>

<sup>13</sup> Vue.js-Instance-Properties: <https://vuejs.org/v2/cookbook/adding-instance-properties.html>

Der Store gliedert sich in mehrere Teile, die in nachfolgender Abbildung 6-1 abgebildet sind.

MainStore Container für die untergeordneten Stores				
<b>UserStore</b> <ul style="list-style-type: none"> <li>– Login</li> <li>– Logout</li> <li>– Benutzerdaten</li> </ul>	<b>AdminStore</b> <ul style="list-style-type: none"> <li>– Erweitere Daten für Admins (z.B. Alle Mitarbeiter auflisten)</li> </ul>	<b>UiStore</b> <ul style="list-style-type: none"> <li>– Dokumenten-vorschau</li> </ul>	<b>NotificationStore</b> <ul style="list-style-type: none"> <li>– Benachrichtigungen (z.B. über neue Chat-Nachrichten, neue Dokumente oder neue Angebote)</li> </ul>	<b>DataStore</b> <ul style="list-style-type: none"> <li>– Alle Kundendaten</li> <li>– Laden und Speichern der Daten</li> <li>– Überwacht die Daten und synchronisiert Änderungen automatisch mit dem Server</li> <li>– Berechnet Werte aus den Kundendaten (z.B. Kaufnebenkosten, Eigenmittel, Aktueller Status der Kreditvermittlung, uvm.)</li> </ul>

Abbildung 6-1: Gliederung des zentralen Datenspeichers/Data-Stores (Eigene Abbildung)

Der MainStore dient hauptsächlich als Container für die untergeordneten Stores. Dadurch wird der Zugriff auf die unterschiedlichen Arten von Daten innerhalb der Anwendung strukturiert und die Übersichtlichkeit der Stores verbessert.

```
import DataStore from './DataStore';
import UiStore from './uiStore';
import UserStore from './userStore';
import NotificationStore from './notificationStore';
import AdminStore from './adminStore';

const MainStore = new Vue({
  data() {
    return {
      Data: DataStore,
      User: UserStore,
      Ui: UiStore,
      Notifications: NotificationStore,
      Admin: AdminStore,
    };
  },
  ...
});

//Zugriff innerhalb der Anwendung
this.$Store.Data.Eigenschaft = 'Max Mustermann'
```

Listing 6-2: MainStore-Gliederung

### UserStore

Im UserStore sind die Benutzerdaten des/der aktuellen Benutzers/Benutzerin wie Name, Telefonnummer, E-Mail-Adresse usw. gespeichert. Zudem stellt er die Authentifizierungsmethoden wie Login und Logout zur Verfügung.

### AdminStore

Der AdminStore stellt erweiterte Funktionen und Daten bereit, wenn es sich beim/bei der angemeldeten Benutzer/in um eine/n Administrator/in handelt. Darunter fällt zum Beispiel das Laden einer Liste aller Kreditvermittler/innen, um diese den Kund/innen zuweisen zu können.

### UiStore

Im UiStore werden Parameter für die Darstellung der Benutzeroberfläche und Funktionen zum Laden und Anzeigen der Dokumente in der Dokumentenvorschau zusammengefasst.

### NotificationStore

Der NotificationStore kümmert sich um das Laden und die Anzeige der für den/die angemeldete/n Benutzer/in verfügbaren Benachrichtigungen wie zum Beispiel neue Chat-Nachrichten, neue Dokumente oder neue Angebote.

### DataStore

Der Hauptteil des zentralen Datenspeichers ist der **DataStore**. Dieser beinhaltet alle Benutzerdaten, die in der Anwendung dargestellt und eingegeben werden. Im Portal werden Werte aus vom/von der Benutzer/in eingegebenen Daten an verschiedenen Stellen berechnet. Der Großteil dieser Werte wird mit Hilfe von Computed-Properties im DataStore berechnet. Damit die Änderungen der Daten sofort mit dem Server synchronisiert werden, werden alle Daten durch einen Vue-Watcher überwacht (siehe Listing 6-3). Sobald die Benutzer/innen Änderungen vornehmen, wird mit Hilfe von *deep-object-diff*<sup>14</sup> ermittelt, welche Daten verändert wurden.

Damit die Speicherung in der serverseitigen Azure Cosmos DB problemlos funktioniert, wird das JavaScript-Objekt mit den geänderten Daten mit Hilfe der *dot-object-Library*<sup>15</sup> von der Klammern-Notation in ein Objekt mit Punkt-Notation umgewandelt. Abschließend werden die veränderten Daten zum Server übermittelt.

```
watch: {
  data: {
    handler: function() {
      this.saveData();
    },
    deep: true,
  },
},
...
methods: {
  async saveData() {
    //Unterschiede zwischen alten Daten (oldData) und geänderten Daten
    let differences = {
      ...addedDiff(oldData, this.data),
      ...updatedDiff(oldData, this.data)
    };
  }
};
```

---

<sup>14</sup> Nähere Informationen siehe Kapitel 5.2

<sup>15</sup> dot-object-Library: <https://github.com/rhalff/dot-object>

```
//Wenn keine Unterschiede --> Abbrechen
//Wenn Kunde/Kundin z.B. den gleichen Wert erneut eingegeben hat
if (!differences || Object.keys(differences).length < 1) return;

//Das JavaScript-Object in eine Dot-Notation für MongoDB umwandeln
const patchObject = dot.dot(differences);

//Übertragen an Server
//...
},
},
```

Listing 6-3: DataStore: Überwachen der Änderungen und Speichern auf Server (Auszug)

Des Weiteren wird der DataStore, durch eine WebSocket-Verbindung mit dem Server über Änderungen der Daten, die durch andere Benutzer/innen am Server vorgenommen werden in Echtzeit informiert. Wenn zum Beispiel ein/e Kreditvermittler/in gleichzeitig die Kundendaten bearbeitet, werden diese Änderungen erkannt, die neuen Daten geladen und dem/der Kunden/Kundin sofort angezeigt.

## 6.2 Registrierung

Die Homepage dient als erste Anlaufstelle für die Kund/innen. Sie gelangen über die direkte Eingabe der URL, den Klick auf Werbungen auf unterschiedlichen Kanälen wie beispielsweise Facebook und Google sowie über die Suche in Suchmaschinen auf die Homepage. Die Hürden für die Kontaktaufnahme müssen so gering wie möglich gehalten werden. Deshalb sollen im Zuge der Registrierung bzw. im Zuge der Anforderung eines individuellen Finanzierungsangebots nur die notwendigen Daten eingegeben werden müssen. Der Fokus liegt im ersten Schritt darauf, die Kontaktdaten für eine persönliche Kontaktaufnahme zu erhalten. Das Portal sollte als Unterstützung dienen/den Prozess unterstützen, aber die persönliche Beratung nicht ersetzen.

Aus diesem Grund ist der Registrierungsprozess mehrstufig aufgebaut.

- 1. Registrierungsformular:** Kund/innen geben die wichtigsten persönlichen und jene für die Finanzierung notwendigen Daten in einem HTML-Formular ein. Das Ziel ist, dass der/die Kunde/Kundin ein Angebot anfordert und somit der persönliche Kontakt hergestellt wird. Deshalb wird an dieser Stelle auf die Eingabe eines Passworts verzichtet.



The image shows two side-by-side form panels. The left panel, titled 'Ihre Finanzierung', contains the following fields: 'Anrede\*' (dropdown menu), 'Vorname\*' (text input), 'Nachname\*' (text input), 'Bundesland\*' (dropdown menu), 'Finanzierungsvorhaben\*' (dropdown menu), 'Eigenkapital' (text input), and 'Kreditbetrag\*' (text input). The right panel, titled 'Ihre Wünsche zur Beratung', contains: 'E-Mail\*' (text input), 'Telefon\*' (text input), 'Wann sind Sie telefonisch am besten erreichbar?' (dropdown menu), a checkbox for 'Ich stimme den Datenschutzbestimmungen zu.', and a green button labeled 'Angebot anfordern'.

Abbildung 6-2: Registrierungsformular (Eigene Abbildung)

- Überprüfen der Daten client- und serverseitig:** Das Formular wird client-seitig überprüft und nur dem Server übermittelt, wenn alle Pflichtfelder ausgefüllt wurden und die Datenschutzbestimmungen akzeptiert wurden. Da clientseitige Überprüfungen durch technisch versierte Anwender/innen leicht umgangen werden können, werden sämtliche Daten am Server einer Plausibilitätskontrolle unterzogen. Da die E-Mail-Adresse für den Login verwendet wird, darf diese nur einmalig verwendet werden. Die Übermittlung der Daten an den Server erfolgt in Form eines JSON-Objekts über eine REST-Schnittstelle.

Um allgemeine Bots, die nicht spezifisch für dieses Registrierungsformular angepasst werden, davon abzuhalten, viele automatische Registrierungen durchzuführen, wird ein einfaches Honeypot-Feld in das Formular eingebaut. Dabei handelt es sich im Grunde um ein Input-Feld innerhalb des Formulars, das für normale Benutzer/innen nicht sichtbar ist und somit auch beim ausgefüllten Formular leer bleibt. Am Server wird nach dem Absenden des Formulars überprüft, ob dieses Honeypot-Feld wie erwartet leer ist. Einfache Bots füllen auch dieses Honeypot-Feld aus und werden somit erkannt.

Auf die Verwendung von Google reCAPTCHA<sup>16</sup> oder ähnliche Services wird, unter anderem aufgrund von datenschutzrechtlichen Bedenken, vorerst verzichtet.

---

<sup>16</sup> reCAPTCHA ist ein Dienst von Google, der Webseiten vor Betrüger/innen und Missbrauch schützt. Die Hauptaufgabe ist, zwischen echten Benutzern/innen (Personen) und Bots (automatisierte Zugriffe) zu unterscheiden und letztere effektiv von Interaktionen auszuschließen. Dabei werden in Abhängigkeit vom erkannten Risiko gegebenenfalls Aufgaben an die Benutzer/innen gestellt. Das reicht vom einfachen Setzen eines Hakens in einer Checkbox bis zur Auswahl von Bildern mit bestimmten Motiven. (Google Ireland Limited, 2021)

3. **Anlegen des/der Benutzers/Benutzerin:** Der/die Benutzer/in wird mit den angegebenen Daten im Portal angelegt und ein Willkommensmail mit einem Hinweis und Link zum Webportal verschickt. Dabei wird ein benutzerspezifischer einmaliger Schlüssel (Verifizierungstoken) erzeugt und zugehörig zur E-Mail-Adresse in der Datenbank gespeichert. Dieser Verifizierungstoken wird der in der URL des Links, den die Kund/innen erhalten, als Parameter übergeben (siehe Listing 6-4).

```
https://portal.simplifin.at/auth/register?type=verify&token=b77e46b9-b216-4548-872d-fd840ae7818c&email=00831177@edu.campus02.at
```

Listing 6-4: Beispiellink mit dem Verifizierungstoken und der E-Mail-Adresse als Parameter

4. **Verifizierung der E-Mail-Adresse und Passwortvergabe:** Mit dem Klick auf den Link in der Willkommens-E-Mail gelangen die Benutzer/innen auf die Seite für die Passwortvergabe. Bei der Passwortvergabe wird die Einhaltung der Sicherheitskriterien sowohl clientseitig – für schnelles Feedback an den/die Benutzer/in – als auch serverseitig überprüft.

The screenshot shows a registration form with two password input fields. The first field is labeled 'Passwort' and the second 'Passwort wiederholen'. Below the fields, there is a list of security criteria:

- Das Passwort muss zwischen 8 und 60 Zeichen lang sein
- Es ist mindestens 1 Zahl notwendig
- Es ist mindestens 1 Kleinbuchstabe notwendig
- Es ist mindestens 1 Großbuchstabe notwendig
- Es ist mindestens 1 Sonderzeichen (!,@,#,\$,%,&,\*,\_ ) notwendig
- Die beiden Passwörter müssen übereinstimmen

At the bottom of the form is a blue button labeled 'ANMELDUNG ABSCHLIESSEN'.

Abbildung 6-3: Passwort-Sicherheitskriterien (Eigene Abbildung)

Serverseitig wird anhand der E-Mail-Adresse, der zugehörige, in der Datenbank gespeicherte Verifizierungstoken ermittelt und mit dem vom Client während der Passwortvergabe übermittelten Verifizierungstoken verglichen. Nur wenn diese beiden Token (Parameter vom Client und serverseitig aus Datenbank) übereinstimmen, kann davon ausgegangen werden, dass der/die Benutzer/in eine gültige E-Mail-Adresse eingegeben hat und es sich um eine zulässige Registrierung handelt.

5. **Weiterleitung zum Portal:** Nachdem die Anmeldung abgeschlossen wurde, wird der/die Benutzer/in zur Übersichtseite des Portals weitergeleitet.

## 6.3 Übersicht Kundenportal

Die Einstiegsseite des Kundenportals gliedert sich in der Desktopversion in zwei Bereiche (siehe Abbildung 6-4). Der Hauptbereich mit der Dateneingabe befindet sich im linken

Bildschirmbereich. Im rechten Teil wird eine Dokumentenvorschau angezeigt. In der mobilen Version wird der Bereich der Dokumentenvorschau ausgeblendet und stattdessen als Pop-Up, bei Auswahl eines Dokuments, realisiert (siehe Abbildung 6-5).

Der Hauptbereich begleitet die Kund/innen Schritt für Schritt bei der Kreditabwicklung:

1. Dateneingabe  
Hier geben die Kund/innen alle notwendigen Informationen, die für die Abwicklung ihrer Finanzierung notwendig sind, ein. Dieser Bereich gliedert sich, zur leichteren und übersichtlicheren Eingabe der Daten, wiederum in mehrere Unterbereiche.
2. Dokumente  
Im Bereich Dokumente befinden sich jene Dokumente, die eine Unterschrift des/der Kunden/Kundin erfordern.
3. Angebote  
Im Bereich Angebote werden die Angebote der Banken zur Verfügung gestellt. Kund/innen erhalten hier die Möglichkeit, die einzelnen Angebote miteinander zu vergleichen.
4. Kontakt  
Der letzte Punkt im Hauptbereich ist die Kontaktfunktion in Form eines Chats. Hier erhalten Kund/innen die Möglichkeit, ihre/n persönliche/n Berater/in direkt zu kontaktieren.

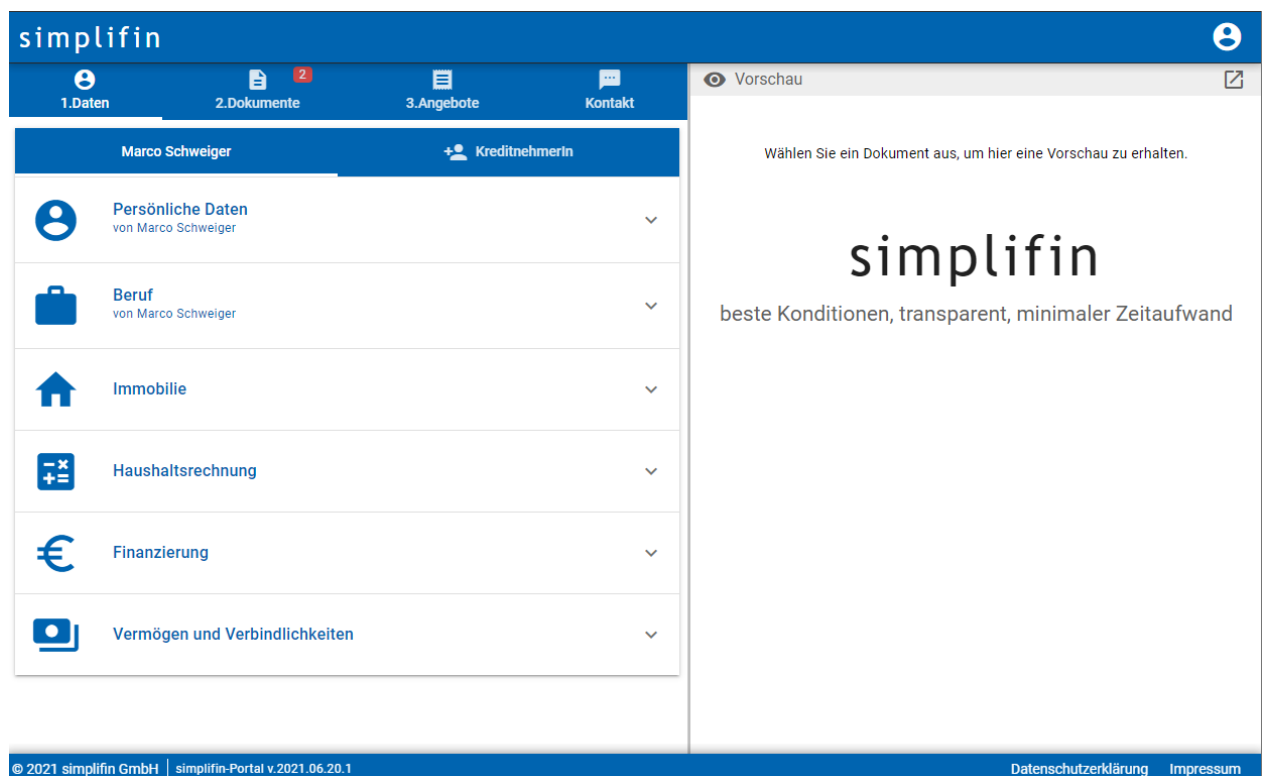


Abbildung 6-4: Portal-Einstiegsseite/Übersicht (Eigene Abbildung)

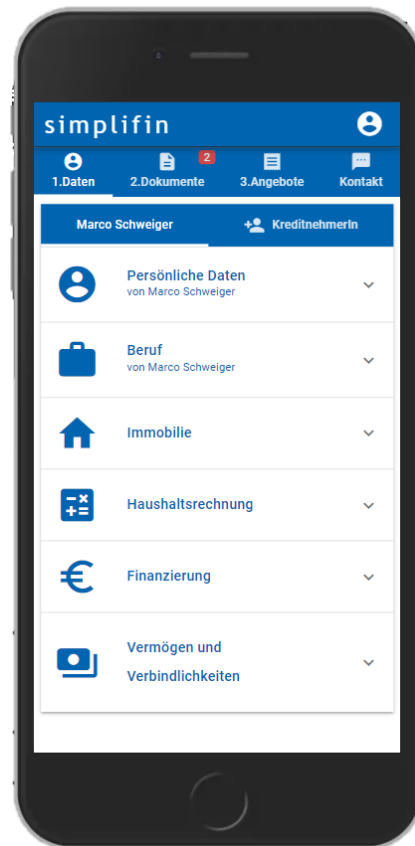


Abbildung 6-5: Portal-Einstiegsseite/Übersicht - Mobile Variante (Eigene Abbildung)

## 6.4 Dateneingabe

Die Dateneingabe gliedert sich, wie in Abbildung 6-4 ersichtlich, in die sechs Hauptbereiche „Persönliche Daten“, „Beruf“, „Immobilie“, „Haushaltsrechnung“, „Finanzierung“ und „Vermögen und Verbindlichkeiten“.

Jeder Bereich ist, zur Verbesserung der Übersichtlichkeit, zusätzlich in kleinere Teilbereiche unterteilt (siehe Abbildung 6-6). Banken verlangen eine Vielzahl an Dokumenten im Zuge einer Finanzierungsabwicklung. Deshalb sind in jedem Bereich einige Dokumente zum Nachweis bzw. zur Bestätigung, der von den Kund/innen gemachten Angaben erforderlich. Kund/innen können diese zum jeweiligen Themenbereich gehörenden Dokumente direkt mittels Smartphone-Kamera oder durch einen Dateiupload hochladen (nähere Informationen zum Datei-Upload finden Sie in Kapitel 6.5).

The image shows a user profile form with two main sections. The left section is titled 'Persönliche Daten' and contains fields for name, title, address, birth date, and family status. The right section is titled 'Miete-Wohnung' and contains fields for rental information, contact details, and document uploads.

**Persönliche Daten**  
 von Marco Schweiger

**Persönliche Daten**

Anrede: Herr

Titel: Dipl.-Ing.

Vorname: Marco

Nachname: Schweiger

Geburtsdatum: 10.05.1988

Geburtsort: Bruck an der Mur

Staatsbürgerschaft: Österreich

Steuerdomizile: Österreich

Ausbildung: Akademische Ausbildung ohne Doktorat mit/ohne MBA

**Familienstand**

Familienstand: ledig

Anzahl Kinder:

Anzahl unterhaltspflichtiger Personen (keine Kinder):

**Miete-Wohnung**

Wohnhaft seit (Monat / Jahr): 11/2013

**Kontakt**

kontaktzeit: Ganztags

E-Mail: 00831177@edu.campus02.at

Hinweis: Eine Änderung dieser E-Mail-Adresse hat keine Auswirkung auf Ihre Login-Daten

Telefonnummer: +436641234567

**Kfz**

Eigenes Kfz vorhanden (kein Firmenauto)

**Dateien**

**Ausweiskopie**  
 Führerschein, Reisepass, Personalausweis

**Meldebestätigung**  
 Auszug aus zentralem Melderegister / Meldezettel

**Heiratsurkunde**

**Scheidungsvergleich / Alimentationsvergleich**  
 Für den Falle einer zurückliegenden Scheidung, benötigen wir bitte Unterlagen aus denen hervorgeht ob sie unterhaltspflichtig sind.

Abbildung 6-6: Dateieingabe und Dateiupload - Persönliche Daten (Eigene Abbildung)

Je nach Art der geforderten Daten, werden dazu passende Eingabefelder verwendet. Zum Beispiel werden Kund/innen vordefinierte Werte in einer Drop-Down-Liste zur Auswahl vorgeschlagen. Beispielsweise kann das Land, die Art der abgeschlossenen Ausbildung oder der Familienstand aus einer Liste ausgewählt werden. Die gewählten Werte werden zusätzlich serverseitig überprüft, um bewusste Falscheingaben durch versierte Benutzer/innen zu verhindern.

## Dynamisches Ein- und Ausblenden von Eingabefeldern

Damit Kreditnehmer/innen nur unbedingt notwendige Daten angeben müssen, werden nicht notwendige Felder dynamisch ein- und ausgeblendet. Beispielsweise werden bestimmte Felder in Abhängigkeit vom Immobilientyp (z.B. Grundstück, Haus, Wohnung) oder davon, ob es sich um ein Bauvorhaben (Neubau) oder den Kauf einer Immobilie handelt, angezeigt oder ausgeblendet.

Dazu wird die Vue-Direktive `v-if` verwendet. Das nachfolgende Listing zeigt auszugsweise das dynamische Ein- und Ausblenden des Feldes für die Fläche der Nebengebäude, welches nur für Häuser angezeigt wird.

```
// Vue Template-Bereich
<InputNumber
  v-if="showIf(['Haus'])"
  v-model.number="data['flaecheNebengebäude']"
  suffix=" m²"
  label="Nebengebäude [m²]"
```

```
</>

// Vue Script-Bereich
methods: {
  /**
   * Gibt zurück, ob entsprechendes Eingabefeld
   * angezeigt werden sollte oder nicht
   * @param {Array.<string>} arrayInput Die Immobilientypen
   * für die das Feld angezeigt werden soll
   */
  showIf(arrayInput) {
    // Wenn kein Wert übergeben wird, nicht anzeigen
    if (!arrayInput) return false;

    // Anzeigen, wenn:
    // --> gewählter Typ im übergebenen Array enthalten
    // --> "Sonstiges" immer anzeigen
    // this.typ enthält den aktuell gewählten Immobilientyp
    if (arrayInput.includes(this.typ) || this.typ === 'Sonstiges') {
      return true;
    }

    return false;
  },
}
```

Listing 6-5: Dynamisches Ein- und Ausblenden der Felder mit Hilfe der vue-Direktive „v-if“

### Live-Berechnung wichtiger Kennzahlen

Wie in den Anforderungen in Kapitel 4.1 angeführt, sollten die wichtigsten Kennzahlen sowohl bei der Haushaltsrechnung als auch in der Finanzierungsübersicht sofort anhand der vom/von der Kunden/Kundin eingegebenen Werte berechnet werden. Die Werte der Finanzierungsübersicht werden durch die Verwendung von Vue-Data-Binding ohne Verzögerung direkt bei Änderung der Werte wie beispielsweise des Kaufpreises oder der Kaufnebenkosten aktualisiert. Kund/innen können die Kaufnebenkosten manuell eingeben oder anhand marktüblicher Werte automatisch berechnen lassen.

€ Finanzierung
^

---

**Projektkosten**

Kaufpreis  
360.000,00 €

---

Grundstückspreis

---

Aufschließungskosten

---

Baukosten

---

Renovierungskosten

---

Küche  
16.000,00 €

---

Einrichtung  
35.000,00 €

---

**Kaufnebenkosten**

⚙️ automatisch berechnen

Gründerwerbssteuer (3,5 %)  
12.600,00 €

---

Grundbucheintragung (1,1 %)  
3.960,00 €

---

Kaufvertrag (ca. 2,4 %)  
8.640,00 €

---

Maklerprovision (ca. 3,6 % inkl. USt.)  
12.960,00 €

**Finanzierungswünsche**

Gewünschte Laufzeit

30 Jahre

5 Jahre

35 Jahre

Art der Verzinsung

variabel  fix  kombiniert

Gewünschte Laufzeit der Fixzins-Periode

20 Jahre

5 Jahre

30 Jahre

Zwischenfinanzierung (wenn notwendig)

---

**Finanzierungsübersicht**

<b>= Projektkosten</b>	<b>411.000,00 €</b>
davon Kaufnebenkostenrelevant	360.000,00 €
<b>+ Kaufnebenkosten</b>	<b>38.160,00 €</b>
<b>= Projektkosten inklusive Kaufnebenkosten</b>	<b>449.160,00 €</b>
<b>- eingebrachte Eigenmittel</b>	<b>100.000,00 €</b>
<b>= notwendiger Auszahlungsbetrag</b>	<b>349.160,00 €</b>
<b>+ Finanzierungsnebenkosten</b>	<b>12.888,94 €</b>
davon Bearbeitungsgebühr	7.240,98 €
davon Grundbucheintragung des Pfandrechts (1,56 %)	5.647,96 €
<b>= Kreditbetrag</b>	<b>362.048,94 €</b>

Abbildung 6-7: Live-Berechnung der wichtigsten Kennzahlen (Eigene Abbildung)

Änderungen der Werte werden durch den zentralen Store mit Hilfe von Computed-Properties<sup>17</sup> (siehe Kapitel 6.1) automatisch erkannt. Daraufhin werden die notwendigen Berechnungen durchgeführt und die neuen Werte zurückgegeben.

```
<Headline title="Finanzierungsübersicht" class="q-mt-lg q-mb-sm" separator />
<div class="row">
  ...
  <div class="col-8 q-pl-lg">
    davon Bearbeitungsgebühr
  </div>
  <div class="col text-right">
    {{ this.$Store.Data.Finanzierung.bearbeitungsgebuehr | asEuro }}
  </div>
  <div class="col-8 q-pl-lg">
    davon Grundbucheintragung <br />des Pfandrechts (1,56 %)
  </div>
  <div class="col text-right">
    {{ this.$Store.Data.Finanzierung.pfandrecht | asEuro }}
  </div>
</div>
```

<sup>17</sup> Computed-Properties sind eine Funktion von Vue.js. Damit werden Werte automatisch berechnet, sobald sich bestimmte Daten ändern. Nähere Informationen finden Sie unter Punkt 3.4.

```

<div class="col-8 text-h6">= Kreditbetrag</div>
<div class="col text-right text-h6">
  {{ this.$Store.Data.kreditbetrag | asEuro }}
</div>
</div>

```

Listing 6-6: Datenbindung an den zentralen Data-Store, um die Daten bei Änderung sofort anzuzeigen

Um die Kund/innen bei der Haushaltsrechnung zu unterstützen, werden die wichtigsten Ausgaben- und Einnahmen-Kategorien aufgelistet. Anhand der Eingaben werden automatisch die wichtigsten Kennzahlen und das, für die Finanzierung besonders relevante, frei verfügbare Einkommen berechnet (siehe Abbildung 6-8).

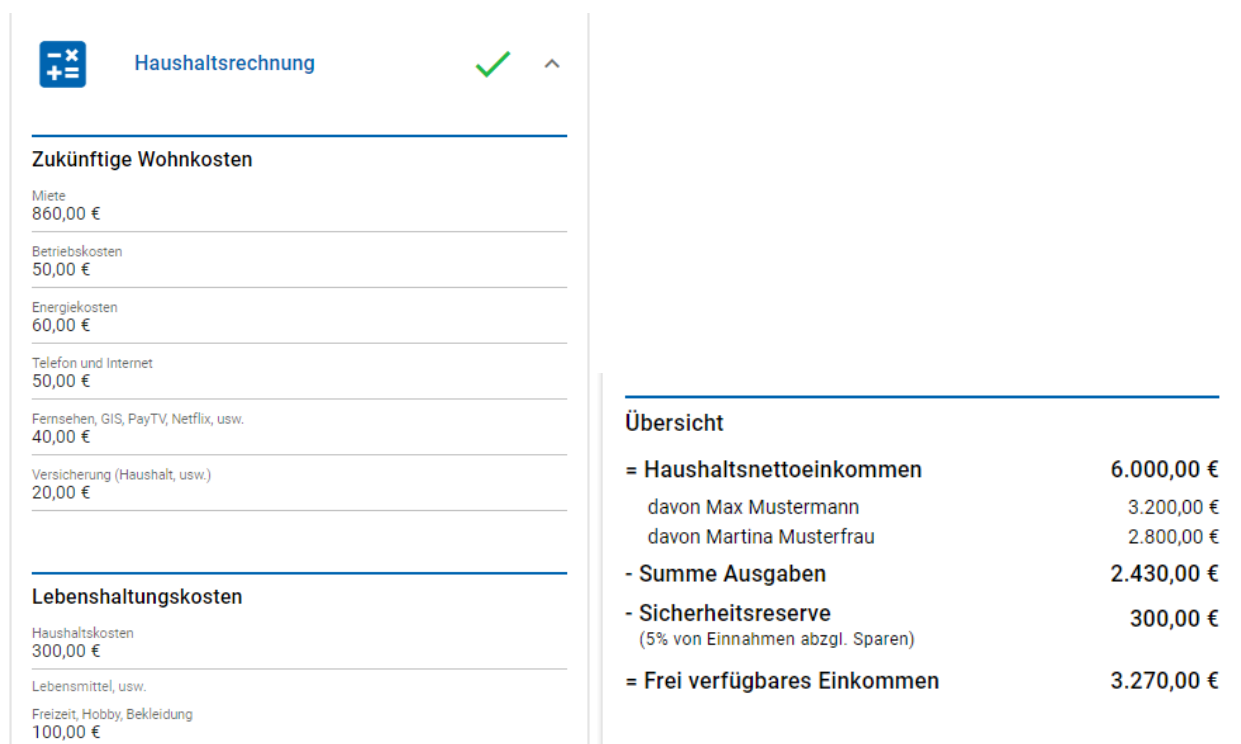
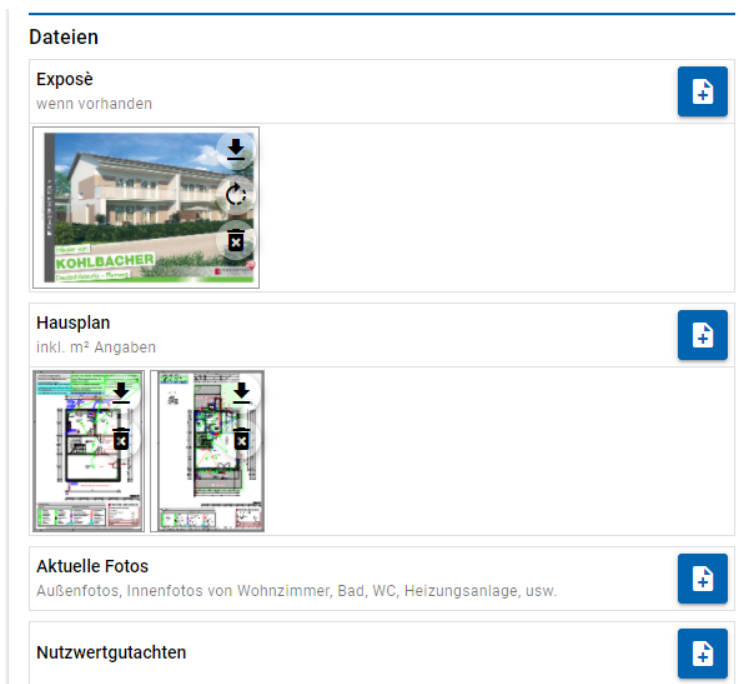


Abbildung 6-8: Haushaltsrechnung (Eigene Abbildung)

## 6.5 Datei-Upload

Banken verlangen Nachweise für die von den Kreditnehmern/-innen gemachten Angaben. Dazu ist es erforderlich, dass Dokumente und Bilder hochgeladen werden können. Hierfür wurde eine Vue-Komponente gestaltet und implementiert (siehe Abbildung 6-5).





Listing 6-7: Files-Komponente

Die Komponente übernimmt mehrere Aufgaben:

1. Upload-Funktion: Auf mobilen Geräten öffnet sich die Kamera, am Desktop der Dateiauswahl-Dialog, um Dateien hinzuzufügen. Je gefordertes Dokument können mehrere Dateien hochgeladen werden. Der/die Benutzer/in wird während dem Upload über eine Fortschrittsanzeige über den Status des Uploads informiert. Die hochgeladenen Dateien werden im Azure Blob Storage abgelegt.
2. Vorschaubild generieren: Nach dem Upload von PDF-Dateien wird mit Hilfe von PDF.js ein Vorschaubild der ersten Seite erzeugt und gespeichert. Bei Bildern wird ebenfalls ein Vorschaubild mit geringerer Auflösung generiert und gespeichert, um die Ladezeiten für die Benutzer/innen zu optimieren.
3. Vorschaubild anzeigen: Sobald Dateien hochgeladen wurden, werden in der Komponente Vorschaubilder der Dateien angezeigt. Nach dem Klick auf ein Vorschaubild, wird die Datei geladen und direkt im Browser angezeigt.
4. Drehen von Bildern: Bei mobilen Geräten werden hauptsächlich Bilder, die mit der Kamera aufgenommen werden, hochgeladen. Um die Ausrichtung der Bilder nachträglich ändern zu können, ist in die Files-Komponente eine Funktion zum Drehen der Bilder integriert.
5. Herunterladen: Kund/innen können die hochgeladenen Dateien lokal herunterladen.
6. Löschen: Solange die jeweilige Datei noch nicht im Zuge einer Kreditanfrage an eine Bank übermittelt wurde, kann die Datei über das Papierkorb-Symbol am Vorschaubild gelöscht werden.

Die Vue-Komponente und das Datenmodell sind so flexibel gestaltet, dass weitere Dokumente sehr einfach ohne Programmierung hinzugefügt werden können. Dazu muss lediglich ein weiterer Files-Tag (siehe Listing 6-8) eingefügt werden. In den Attributen werden der Titel, die

Beschreibung, optional ein Hilfe-Text und der Tag-Name angegeben. Anhand des Tag-Namens werden die jeweils dem/der Kunden/Kundin gehörenden Dokumente automatisch abgespeichert. Bereits vorhandene Dokumente werden ebenfalls anhand des Tag-Namens gefunden und angezeigt.

```
<Files
  :documents="storeData.data.dokumente"
  text="Exposé"
  caption="wenn vorhanden"
  tag="expose"
  :person-id="data._id"
  :user-id="storeData.data._id"
/>
```

Listing 6-8: Files-Komponente

## 6.6 Eine/n zweite/n Kreditnehmer/in hinzufügen

Viele Kredite werden von mehr als einer Person abgeschlossen. Deshalb wurde in den Anforderungen in Kapitel 4.1 definiert, dass mindestens zwei Kreditnehmer/innen unterstützt werden müssen. Um diese Anforderung zu erfüllen, hat man direkt unter der Auswahl der Hauptbereiche die Möglichkeit, eine/n weitere/n Kreditnehmer/in hinzuzufügen. Dabei wird berücksichtigt, dass sich nur die persönlichen und die beruflichen Daten der beiden Personen unterscheiden. Man erkennt die personenbezogenen Daten an den kleinen Ergänzungen unter diesen beiden Menüpunkten (im Beispiel in Abbildung 6-9 „von Marco Schweiger“ und „von Caroline Viertler“). Sämtliche anderen Daten sind für beide Kreditnehmer/innen gleich und müssten deshalb nicht noch einmal eingegeben werden.

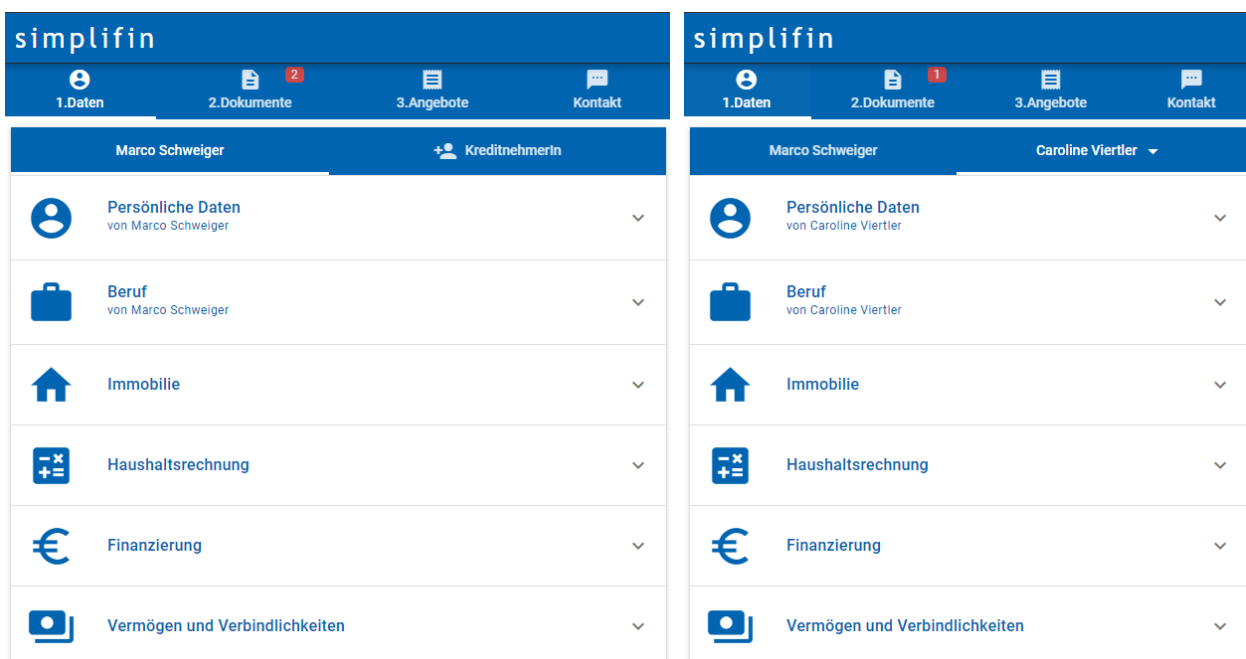


Abbildung 6-9: Links: Erster Kreditnehmer, Rechts: Zweite Kreditnehmerin (Eigene Abbildung)

## 6.7 Dokumentenvorschau

Damit sich Kund/innen und Kreditvermittler/innen einen schnellen Überblick über die Dokumente machen können, ohne diese zuvor herunterladen zu müssen, um diese anschließend lokal in einem entsprechenden Bildbetrachtungs- oder PDF-Programm anzuzeigen, wird eine Anzeigefunktion für PDF und Bild-Dateien direkt in das Web-Portal integriert.

Zu diesem Zweck wurde eine Vue-Komponente entwickelt, die je nach Dateityp – Bild oder PDF – entweder ein Standard-HTML-Image-Element oder einen selbstentwickelten, auf PDF.js und Fabric.js basierenden PDF-Viewer verwendet (siehe Kapitel 6.9 für weitere Informationen zum PDF-Viewer). Die Komponente bzw. Vorschau wird auf Desktop-Geräten seitlich neben dem Eingabebereich und auf mobilen Geräten als Pop-Up angezeigt.

Der PDF-Viewer (siehe Abbildung 6-10) bietet bekannte Funktionen wie Zoom und Zoom-To-Fit<sup>18</sup> (links unten) sowie Seitensprungfunktionen (rechts-unten).

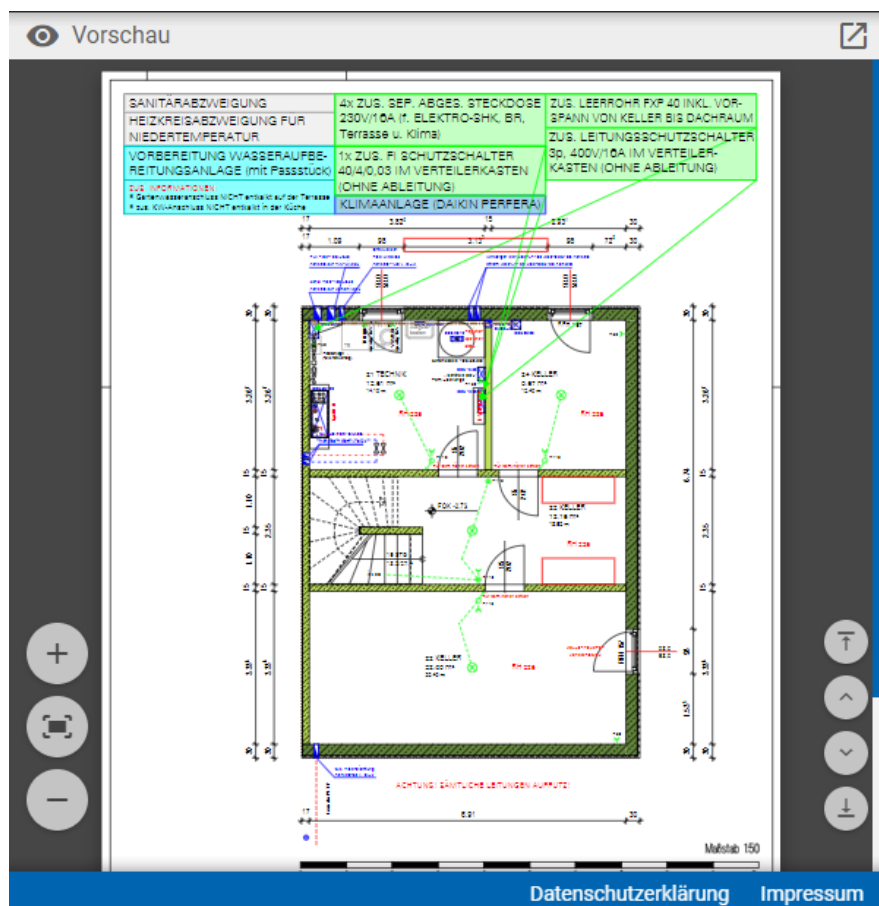


Abbildung 6-10: Dateivorschau - PDF-Viewer (Eigene Abbildung)

<sup>18</sup> Mit Hilfe von Zoom-To-Fit wird die Seite sowohl in der Breite als auch in der Höhe so in den Anzeigebereich eingepasst, dass die gesamte Seite sichtbar ist.

## 6.8 Dokumente unterschreiben

Wie in Kapitel 2.1.1 beschrieben, sind den Kund/innen einige Dokumente im Zuge der Hypothekarkreditvermittlung auszuhändigen. Um den Erhalt und die darin enthaltenen Informationen zu bestätigen, müssen diese vom/von der Kunden/Kundin unterzeichnet werden. Derartige Dokumente werden für Kund/innen im Bereich „Dokumente“ aufgelistet (siehe Abbildung 6-11). Administrator/innen können Dokumente als Standard-Dokumente definieren. Bestimmte Dokumente wie zum Beispiel der Kreditvermittlungsauftrag und die Standardinformationen, werden den Kund/innen direkt nach der Registrierung angezeigt. Weitere Dateien werden den Kund/innen zur Verfügung gestellt, sobald die Dateneingabe abgeschlossen wurde. Vor der Übermittlung der Daten an die Banken müssen diese Dokumente unterzeichnet werden, um die Richtigkeit der eingegebenen Daten zu bestätigen.

Kund/innen können die Dokumente downloaden, unterzeichnen und wieder hochladen oder sie, als einfachere Möglichkeit, direkt in der Anwendung unterzeichnen. Nach dem Klick auf das Stift-Symbol (siehe Abbildung 6-11) wird das Dokument mit den erforderlichen Daten automatisch erzeugt und in der Dokumentenvorschau angezeigt.

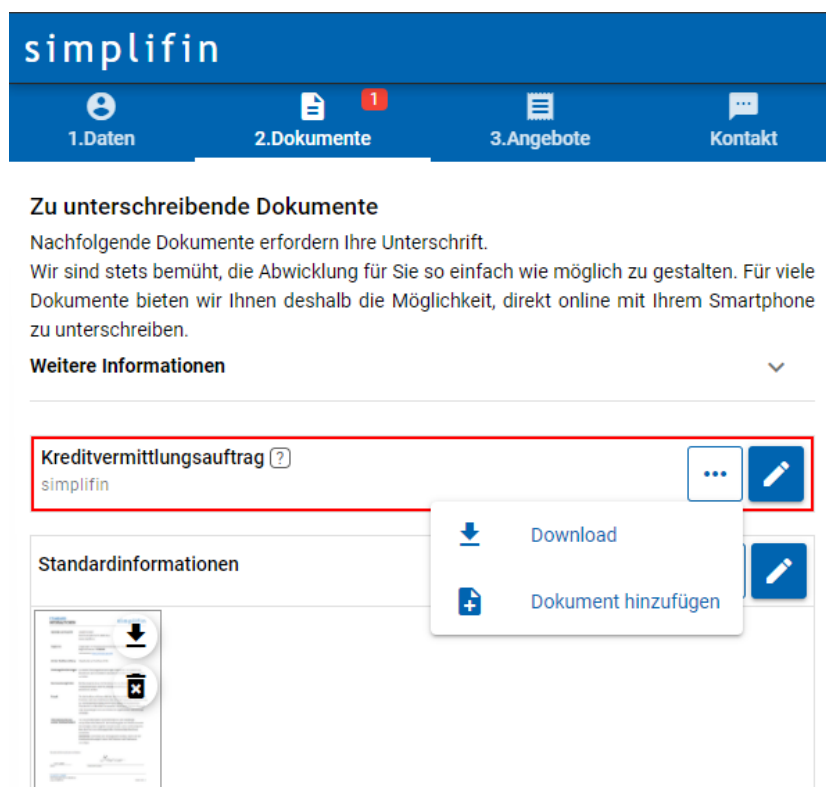


Abbildung 6-11: Hauptbereich – 2. Dokumente (Eigene Abbildung)

Wie in Abbildung 6-12 ersichtlich, sind die Stellen, an denen eine Unterschrift durch den/die Kreditnehmer/in erforderlich ist, in der Dokumentenansicht deutlich gekennzeichnet. Alle Kreditnehmer/innen müssen die Dokumente unterzeichnen, deshalb werden je nachdem, ob der Kredit von einem Kreditnehmer oder zwei Kreditnehmern/-innen beantragt wird, unterschiedliche Dokumentenvorlagen mit der passenden Anzahl an Unterschriftsfeldern angezeigt.



Abbildung 6-12: Unterschriftsfelder (eigene Abbildung)

Nach dem Klick auf „Hier unterschreiben“ wird das Unterschriftsfeld geöffnet (siehe Abbildung 6-13). Die Unterschrift ist zwar grundsätzlich auch mittels Maus am Desktop-PC möglich, idealerweise wird dazu aber ein (mobiles) Endgerät mit Touchscreen verwendet.

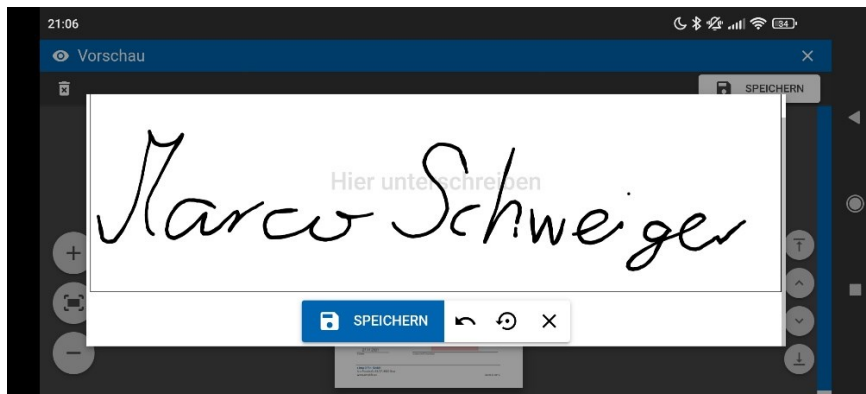


Abbildung 6-13: Unterschriftsfunktion auf mobilem Endgerät (Eigene Abbildung)

Sobald die Unterschrift und das Dokument gespeichert sind, wird eine PDF-Datei erzeugt, die vom/von der Kunden/Kundin nicht mehr verändert werden kann. Somit sind die Unterschrift und die Informationen im Dokument fest miteinander verbunden.

Im nachfolgenden Kapitel 6.9 befinden sich weitere technische Informationen zur automatischen Dokumentenerstellung.

## 6.9 PDF-Viewer und PDF-Template-Designer

Administrator/innen haben die Möglichkeit, Dokumentenvorlagen selbstständig, ohne Programmierkenntnisse direkt im Webportal zu erstellen (siehe Abbildung 6-15). Dafür wurde ein PDF-Template-Designer entwickelt. Dieser verwendet grundsätzlich die folgenden drei Elemente:

- **PDF-Datei:** Beim Anlegen eines neuen Template muss eine PDF-Datei ausgewählt und hochgeladen werden. Dieses Basis-PDF muss lokal, zum Beispiel mit Hilfe von Microsoft Word, erstellt werden und dient als Basis bzw. Hintergrund der Vorlage.

- **Text-Feld:** Diese Felder werden später automatisch mit den Kundendaten befüllt. Dazu muss im hinzugefügten Feld eine sehr einfache, selbstentwickelte Syntax<sup>19</sup>, die sich stark an der Datenbankstruktur orientiert, verwendet werden (siehe Abbildung 6-14). Sobald der/die Kunde/Kundin die Vorlage im Kundenportal öffnet, wird diese Syntax automatisch durch den entsprechenden Wert der Kundendaten ersetzt.

Zusätzlich können an das Vue-Pendant angelehnte Pipes verwendet werden. Durch das Anhängen eines Ausdrucks an die Variablendefinition können die Daten als Flächenangabe ( |a ), Prozentwert( |p ), Jahr( |y ), Integer( |i ), Dezimalzahl mit zwei Kommastellen ( |f ), Euro-Wert ( |m ), Checkbox ( |b ) oder Datum ( |d ) formatiert werden (siehe „Grundstücksfläche“ in Abbildung 6-14).

<b>Baufortschritt</b>	Status Immobilie:	<u>{immobilie.baufortschritt}</u>
	Baujahr des Gebäudes:	<u>{immobilie.bauJahr}</u>
	Baubeginn:	<u>{immobilie.bauBeginn}</u>
	Bauende:	<u>{immobilie.bauEnde}</u>
<b>Flächenangaben</b>	Grundstücksfläche	<u>{immobilie.flaecheGrundstueck f} m<sup>2</sup></u>
	Wohnfläche	<u>{immobilie.flaecheWohn f} m<sup>2</sup></u>

Abbildung 6-14: Textfeld-Syntax für die automatische Eingabe (Eigene Abbildung)

Die Schriftgröße des Textfeldes kann beliebig angepasst werden. Die Schriftart ist standardmäßig Arial und lässt sich im Prototyp nicht verändern.

- **Unterschriftsfeld:** Damit Kund/innen Unterschriften im Dokument setzen können, werden Unterschriftsfelder am Dokument positioniert. Beim Klick auf das Feld wird ein Pop-Up geöffnet, in dem Benutzer/innen mit Hilfe der Maus oder des Touchscreens unterzeichnen können. Das Unterschriftsfeld im Pop-Up verwendet die JavaScript Library „Signature Pad“ von Szymon Nowak<sup>20</sup> als Basis.

---

<sup>19</sup> Beim Prototyp muss diese Syntax verwendet werden. Zukünftig ist geplant, diese Funktionalität dahingehend zu erweitern, dass der gewünschte Wert aus einer Liste mit den möglichen Feldern/Kundendaten ausgewählt werden kann.

<sup>20</sup> Signatur Pad von Szymon Nowak: [https://github.com/szimek/signature\\_pad](https://github.com/szimek/signature_pad)

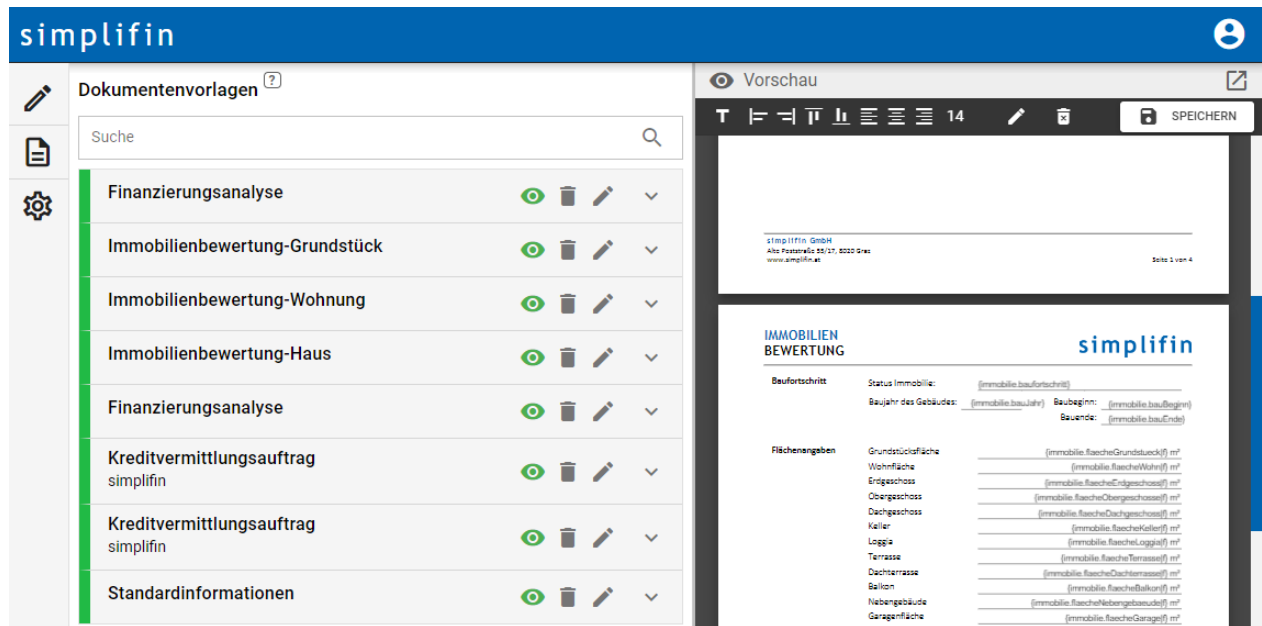


Abbildung 6-15: PDF-Template-Designer (Eigene Abbildung)

Der PDF-Viewer für die Dokumentenanzeige und der PDF-Template-Designer verwenden denselben technischen Unterbau. Dieser ist in mehrere Ebenen und Unterkomponenten gegliedert und ist in Abbildung 6-16 vereinfacht dargestellt. Die oberste Komponente ist der PDF-Viewer, der sich um das Design der PDF-Anzeige und die Darstellung der Steuerelemente kümmert. Der PDF-Viewer beinhaltet die Komponente für das PDF-Dokument. Diese Komponente lädt das gewünschte PDF anhand der URL und ist zuständig für das Speichern des gesamten Dokuments, das sich aus mehreren einzelnen Seiten zusammensetzt. Jede Seite wird jeweils durch eine PDF-Page-Komponente repräsentiert. Diese Komponente ist das Kernstück des PDF-Viewers und des PDF-Template-Designers. In dieser Komponente befinden sich die beiden Schichten für das Darstellen des PDFs (auf PDF.js aufbauender PDF-Layer) und der darüberliegenden Kommentare (auf Fabric.js aufbauender Annotation-Layer, der die Text- und Unterschriftsfelder darstellt).

Die Anmerkungsschicht (Annotation-Layer) und die PDF-Schicht (PDF-Layer) sind jeweils mit Fabric.js initialisierte HTML5-Canvas-Elemente. Das PDF wird mit Hilfe von PDF.js geladen und anschließend im Canvas-Element dargestellt. Eine Ebene direkt über diesem Canvas-Element, befindet sich ein weiteres Canvas-Element, das die Text- und Unterschriftsfelder darstellt. Mit Hilfe der Steuerelemente im oberen Teil des Viewers (siehe Abbildung 6-15) können diese Elemente hinzugefügt und anschließend per Drag and Drop auf der Seite positioniert werden.

Fabric.js bietet die Möglichkeit, alle Elemente, die in einem Canvas dargestellt werden, als JSON zu serialisieren und deserialisieren. Mit Hilfe dieser Funktionalität werden die Daten und Elemente des Annotation-Layers als separate Schicht, zusätzlich zur eigentlichen PDF-Datei, am Server gespeichert. Da die Daten somit getrennt vom eigentlichen PDF gespeichert werden, können die Templates erneut geladen und bearbeitet/angepasst werden.

Erst wenn der/die Kunde/Kundin die Unterschrift getätigt bzw. die Daten bestätigt hat, werden die beiden Schichten zusammengefügt und ein einzelnes PDF erzeugt. Dafür werden alle Seiten des

Dokuments durchlaufen und jeweils die beiden Canvas-Elemente in JPG-Bilder umgewandelt. Diese beiden Bilder werden anschließend mit Hilfe von jsPDF in PDFs umgewandelt und alle Seiten zu einem Dokument kombiniert.

<b>PDF-Viewer</b>	Darstellen der Steuerelemente (Text-, Unterschriftenfeld, Elemente-Ausrichtung, Zoom, Seitensprung)
<b>PDF-Dokument</b>	Laden des PDF anhand einer URL inklusive Ladefortschrittsanzeige Zusammenfügen und Speichern des gesamten PDF mit Hilfe von jsPDF Ermitteln der aktuell sichtbaren Seiten
<b>PDF-Seite</b>	Beinhaltet den Annotation- und den PDF-Layer Rendern des PDF Speichern der Einzelseiten als Bilddateien
<b>Annotation-Layer Anmerkungsschicht Fabric.js</b>	Verwendet Fabric.js zur Canvas-Manipulation Positionieren und Anzeigen der Text- und Unterschriftenfelder Event-Handler für den Klick auf das Unterschriften Feld Automatisches Ausfüllen der Textfelder mit Kundendaten Lädt und Speichert die Text- und Unterschriftenfelder (JSON)
<b>PDF-Layer PDF-Schicht PDF.js</b>	Laden und Anzeigen des PDF-Inhalts mit Hilfe von PDF.js

Abbildung 6-16: Technische Ebenen des PDF-Viewers (Eigene Abbildung)

Eine Herausforderung während der Umsetzung war, dass die Geschwindigkeit des PDF-Viewers sehr langsam und der Speicherverbrauch sehr hoch waren. Größere Dokumente ab ca. 10 Seiten konnten deshalb, insbesondere auf mobilen Endgeräten, nicht angezeigt werden. Um dieses Problem zu lösen, wird anhand der Scroll-Position, dem aktuellen Zoom-Level und der Seitengröße ermittelt, welche Seiten sich gerade im sichtbaren Bereich befinden. Anhand dieser Information werden nur diese sichtbaren Seiten sowie jeweils eine Seite davor und danach geladen und in den Canvas-Elementen gerendert. Dadurch kann der Ressourcenverbrauch verringert werden und die Anzeige funktioniert auch bei Dokumenten mit mehr als 10 Seiten sowohl in Desktop-Browsern als auch auf mobilen Endgeräten problemlos.

## 6.10 Übersicht für Kreditvermittler/innen

Kreditvermittler/innen erhalten eine Übersicht über alle Kund/innen, deren Anträge sich derzeit in Bearbeitung befinden. Dabei werden die wichtigsten Daten wie zum Beispiel die Kontaktdaten, Finanzierungssumme, Loan-to-Value-Faktor (LTV)<sup>21</sup> und der/die zugewiesene Betreuer/in dargestellt (siehe Abbildung 6-17). Anhand der Farbkennzeichnung wird der aktuelle Status der Abwicklung kenntlich gemacht.

<sup>21</sup> Als Loan-To-Value-Faktor bezeichnet man das Verhältnis von Kreditbetrag zu Kaufpreis bzw. zum Wert der Immobilie. Diese Kennzahl stellt ein wichtigen Faktor für die Kreditkonditionen dar.



Dabei wird zwischen folgenden Status unterschieden:

1. Unvollständig: Der/die Kunde/Kundin hat sich gerade erst registriert und es wurden noch keine Daten eingetragen.
2. Zur Überprüfung: Es wurden bereits Daten eingetragen. Der/die Kreditvermittler/in muss die Daten auf Vollständigkeit und Korrektheit überprüfen.
3. Überprüft: Die Daten wurden durch den/die Kreditvermittler/in erfolgreich überprüft. Banken können für Angebote kontaktiert werden.
4. Weitergeleitet: Die Daten wurden bereits an Banken weitergeleitet. Es können auch schon Angebote vorhanden sein.

In den Kundendetails finden sich nähere Informationen zum Status bzw. Fortschritt der Kreditabwicklung (siehe Abbildung 6-18).

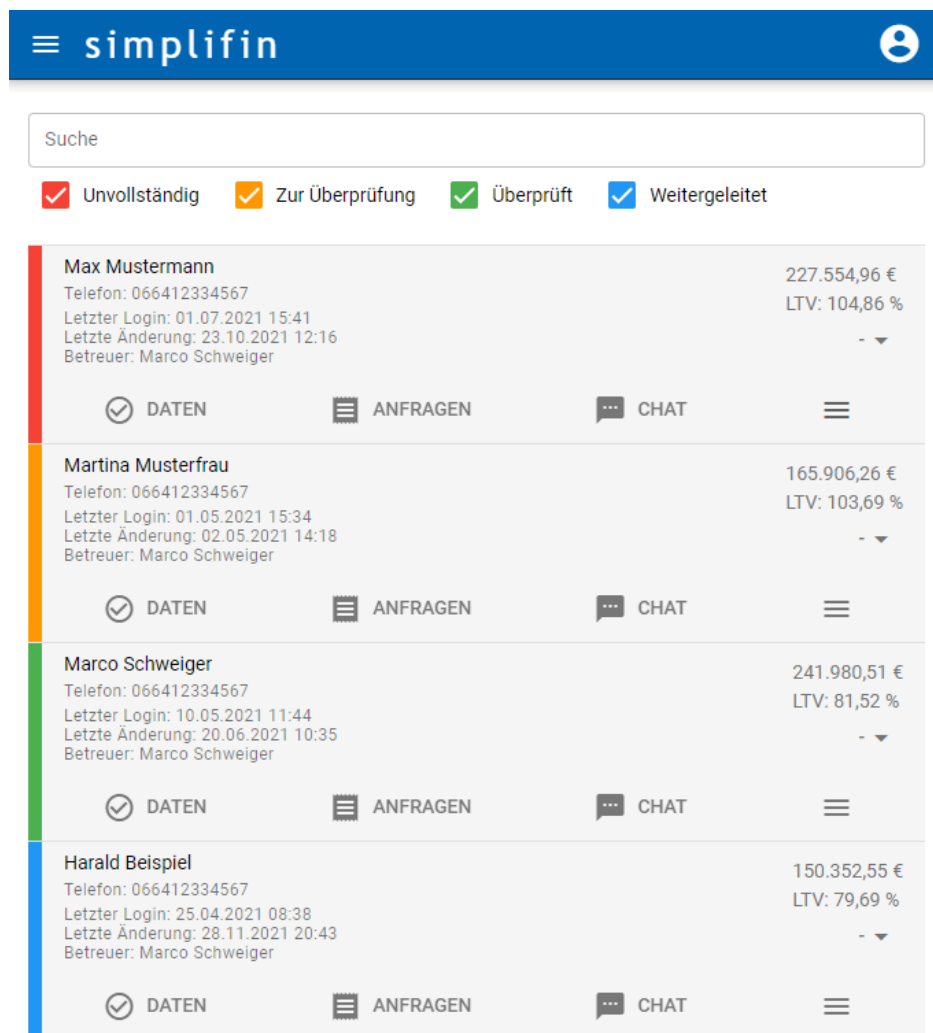


Abbildung 6-17: Übersicht für Kreditvermittler/innen (Eigene Abbildung)

Beim Klick auf das Burger-Menü rechts unter dem LTV-Wert erscheinen weitere Daten zu dem/der Kunden/Kundin (siehe Abbildung 6-18). Hier kann der/die Kreditvermittler/in Anmerkungen zu dem/der Kunden/Kundin machen oder Notizen aus dem letzten Gespräch einfügen und dem/der zuständigen Kreditvermittler/in zuweisen. Im Prototyp sieht jede/r

Kreditvermittler/in alle Kund/innen. In einem zukünftigen Entwicklungsschritt soll das Berechtigungskonzept dahingehend erweitert werden, dass Kreditvermittler/innen nur die ihnen zugewiesenen Kund/innen sehen können. Im aktuellen Zustand dient die Beraterzuweisung nur dazu, die Zuständigkeit kenntlich zu machen.

The screenshot displays the customer details for Max Mustermann. At the top, the name 'Max Mustermann' is shown alongside contact information: 'Telefon: +436641234567', 'Letzter Login: 28.11.2021 20:48', 'Letzte Änderung: 28.11.2021 20:53', and 'Betreuer: Marco Schweiger'. Financial data is presented on the right: '362.048,94 €' and 'LTV: 100,57 %'. Below this is a navigation bar with icons for 'DATEN', 'ANFRAGEN', 'CHAT', and a menu icon. The 'Anmerkungen' section contains an empty text input field. The 'Mitarbeiter zuweisen' section shows 'Marco Schweiger' as the assigned employee. The 'Status: Registriert' section includes a checkbox for 'Kreditvermittlungsauftrag hochgeladen' and a progress bar labeled 'Bereiche: 7/8'. The 'Vermittlung abschließen' section features a toggle switch and a red 'ABSCHLIESSEN' button. The 'Kunden löschen' section has a toggle switch and a red 'LÖSCHEN' button.

Abbildung 6-18: Kundendetails für Kreditvermittler/innen (Eigene Abbildung)

### Überprüfung der Daten

Durch den Klick auf den Menüpunkt „Daten“ gelangen Kreditvermittler/innen zur gleichen Ansicht, die auch den Kund/innen zur Verfügung steht. Zusätzlich können Kreditvermittler/innen aber weitere Funktionen nutzen und weitere Daten einsehen (siehe Abbildung 6-19). Unter anderem werden die Untermenüs der Kunden um eine Checkbox erweitert, mit der Kreditvermittler/innen nach der Überprüfung der Daten die Vollständigkeit bestätigen. Diese Bestätigung wird auch für die Ermittlung des aktuellen Status verwendet.

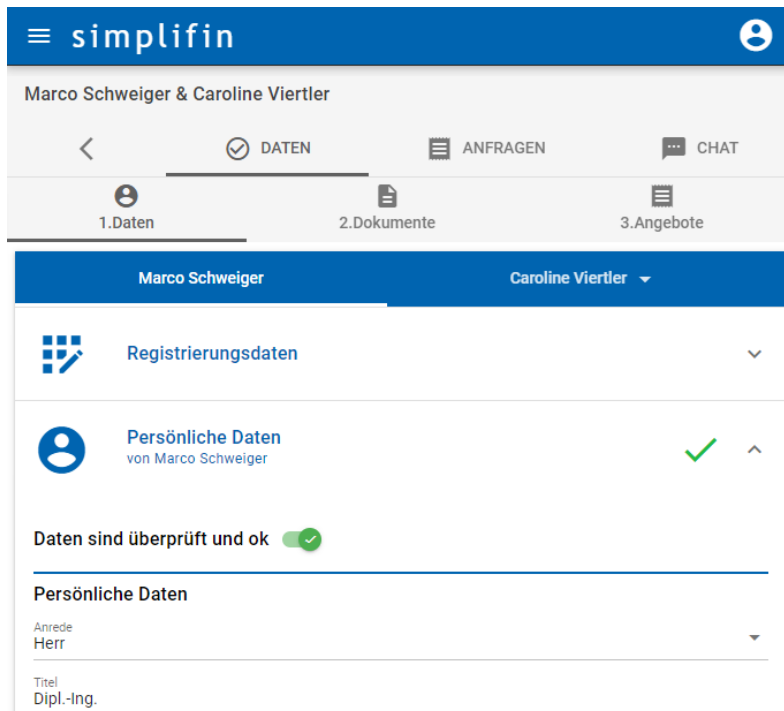


Abbildung 6-19: Datenüberprüfung durch Kreditvermittler/innen (Eigene Abbildung)

## 6.11 Anfragen an Banken stellen

Um die Informationen und Unterlagen zur Angebotseinholung an Banken weiterleiten zu können, wird für jede Bank, von der ein Angebot eingeholt werden soll, im Admin-Bereich im Menüpunkt „Anfragen“ eine Anfrage angelegt (Abbildung 6-20). Jede erstellte Anfrage lässt sich als Vorlage speichern und dadurch bei zukünftigen Anfragen als Basis zur Anfrageerstellung wiederverwenden.

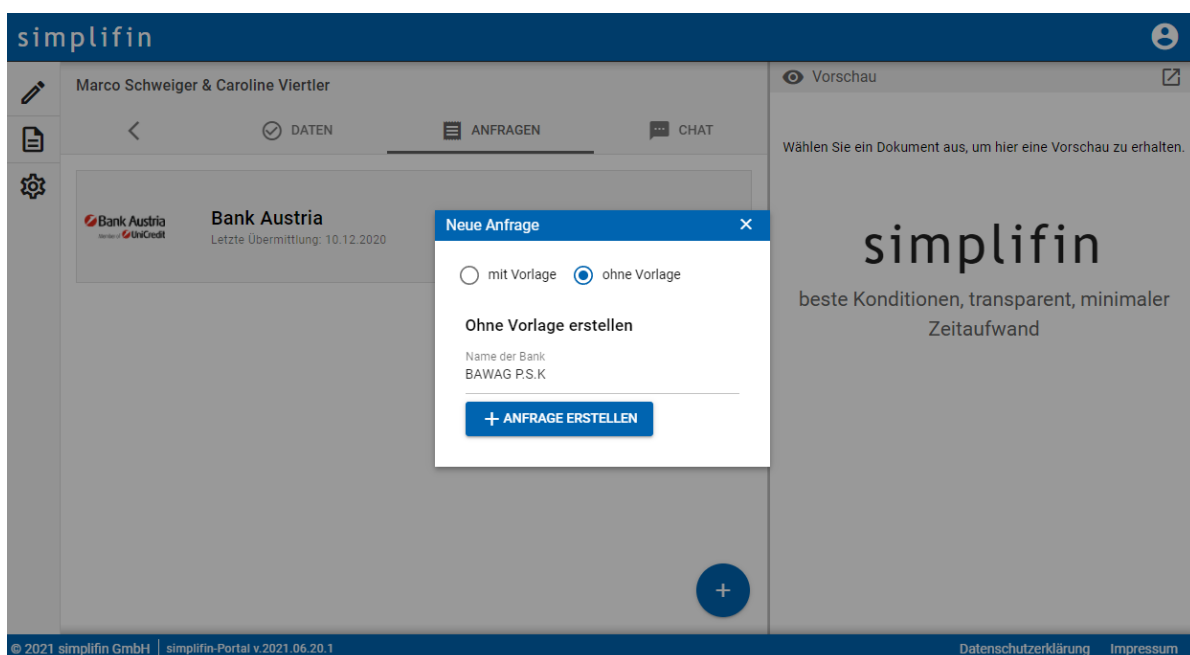


Abbildung 6-20: Anfrage erstellen (Eigene Abbildung)

Grundsätzlich fordern alle Banken die gleichen Dokumente und Nachweise, trotzdem gibt es kleinere Unterschiede oder bankenspezifische Formulare, die bei der Kreditanfrage bereitzustellen sind. Deshalb müssen Kreditvermittler/innen nach dem Hinzufügen einer Anfrage zuerst die erforderlichen Dokumentenvorlagen der jeweiligen Bank auswählen. Dazu werden die notwendigen Vorlagen aus einer Liste mit allen Dokumentenvorlagen selektiert. Eine Filtermöglichkeit erleichtert die Auswahl (siehe Abbildung 6-21). Nach dem Speichern der Anfrage werden die selektierten Dokumente dem/der Kunden/Kundin in seinem/ihrer „Dokumente“-Bereich zur Unterschrift bereitgestellt.

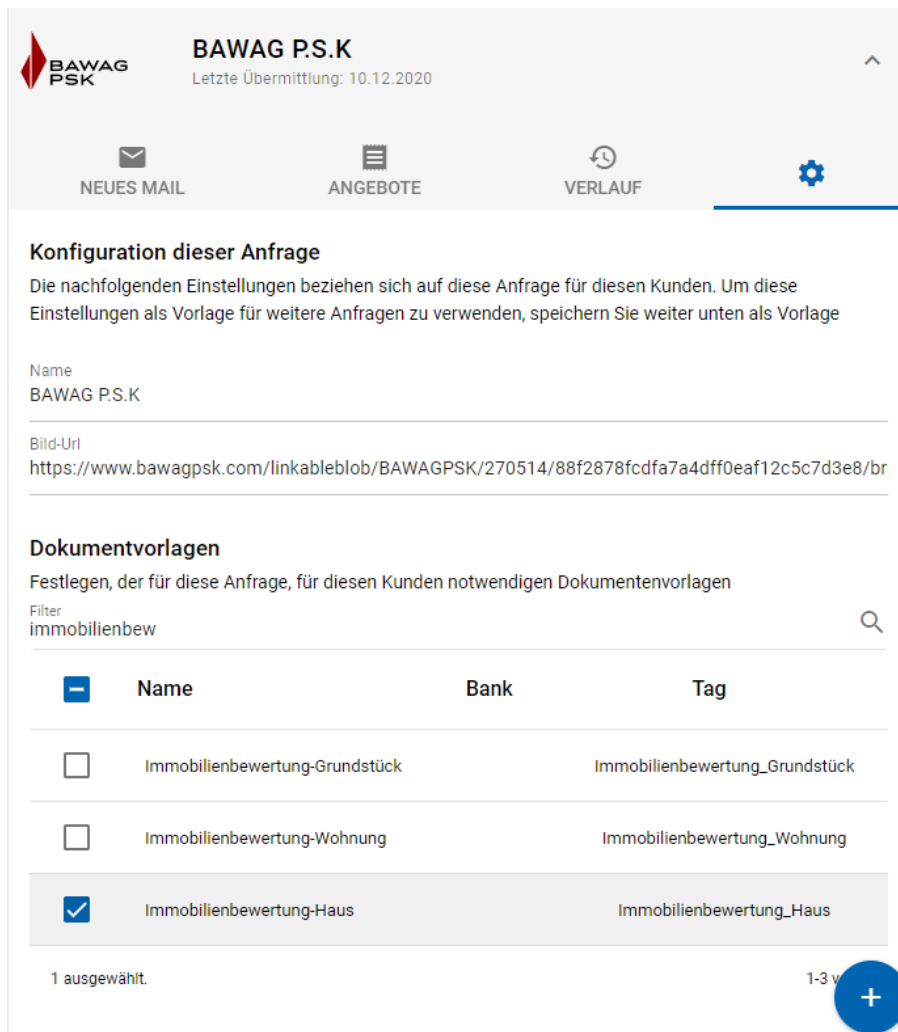


Abbildung 6-21: Anfrage bearbeiten – Dokumentenvorlagen (Eigene Abbildung)

Sobald alle erforderlichen Daten und Dokumente vom/von der Kunden/Kundin bereitgestellt wurden, kann der/die Kreditvermittler/in die Kreditanfragen an die Banken weiterleiten, um Angebote einzuholen. Dafür ist direkt im Webportal ein einfach gehaltener E-Mail-Client integriert (siehe Abbildung 6-22). Damit kann die Kreditanfrage direkt im Portal per E-Mail an den/die Partnerbetreuer/in<sup>22</sup> der Bank weitergeleitet werden. Unter den Standard-Feldern für

<sup>22</sup> Partnerbetreuer/innen sind die Ansprechpartner/innen für Kreditvermittler/innen und sind seitens der Bank für die Kreditabwicklung verantwortlich.

Empfänger/in, CC und den Text werden alle Dokumente aufgelistet, die für die jeweilige Anfrage relevant sind. Um einen Überblick darüber zu behalten, welche Dokumente bereits an die jeweilige Bank versendet wurden, werden Dateien, die in der Vergangenheit bereits an diese Bank verschickt wurden, in der Auflistung grün markiert (siehe Abbildung 6-22). Beim Versenden der Mails wird der E-Mail-Text automatisch in eine Vorlage mit der korrekten firmenmäßigen Signatur eingefügt.

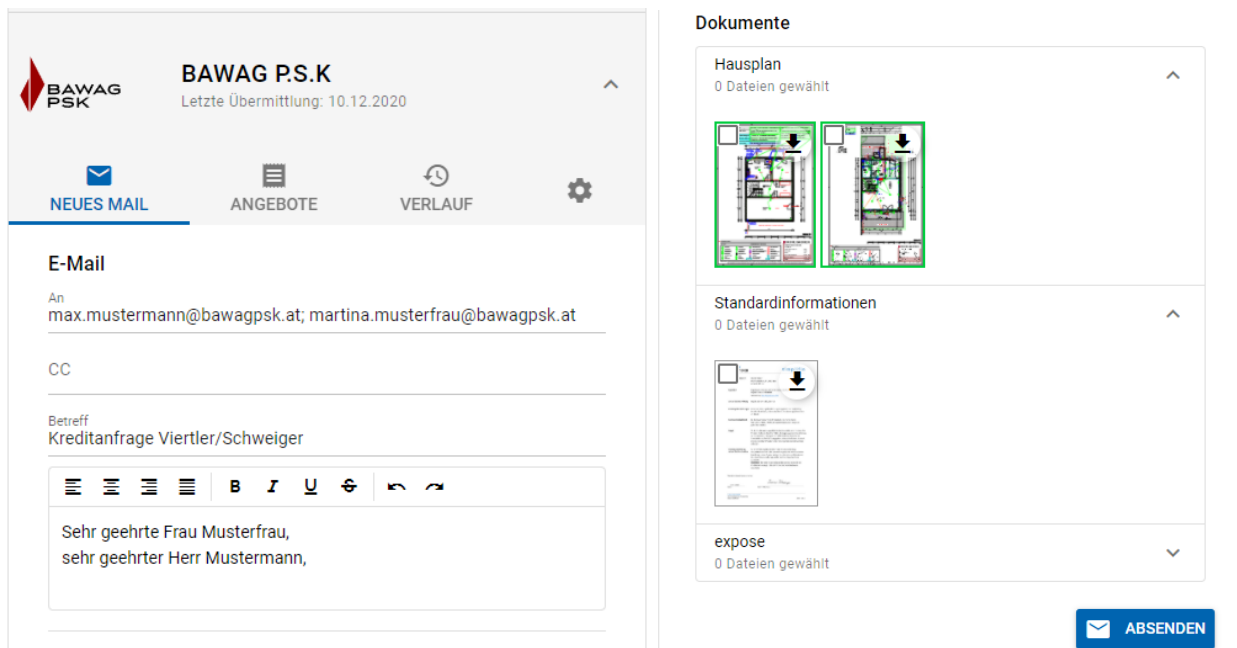


Abbildung 6-22: Kreditanfrage - E-Mail an Bank (Eigene Abbildung)

Sämtliche E-Mails, die über das Webportal versendet werden, können in einem Verlauf, zugehörig zur jeweiligen Anfrage/Bank, eingesehen werden (siehe Abbildung 6-23). Empfangene Mails werden beim Prototyp nicht im Portal angezeigt. Für zukünftige Versionen ist im ersten Schritt geplant, die E-Mail-Antworten der Banken manuell einzupflegen, damit der Verlauf sowohl gesendete als auch erhaltene Nachrichten enthält.

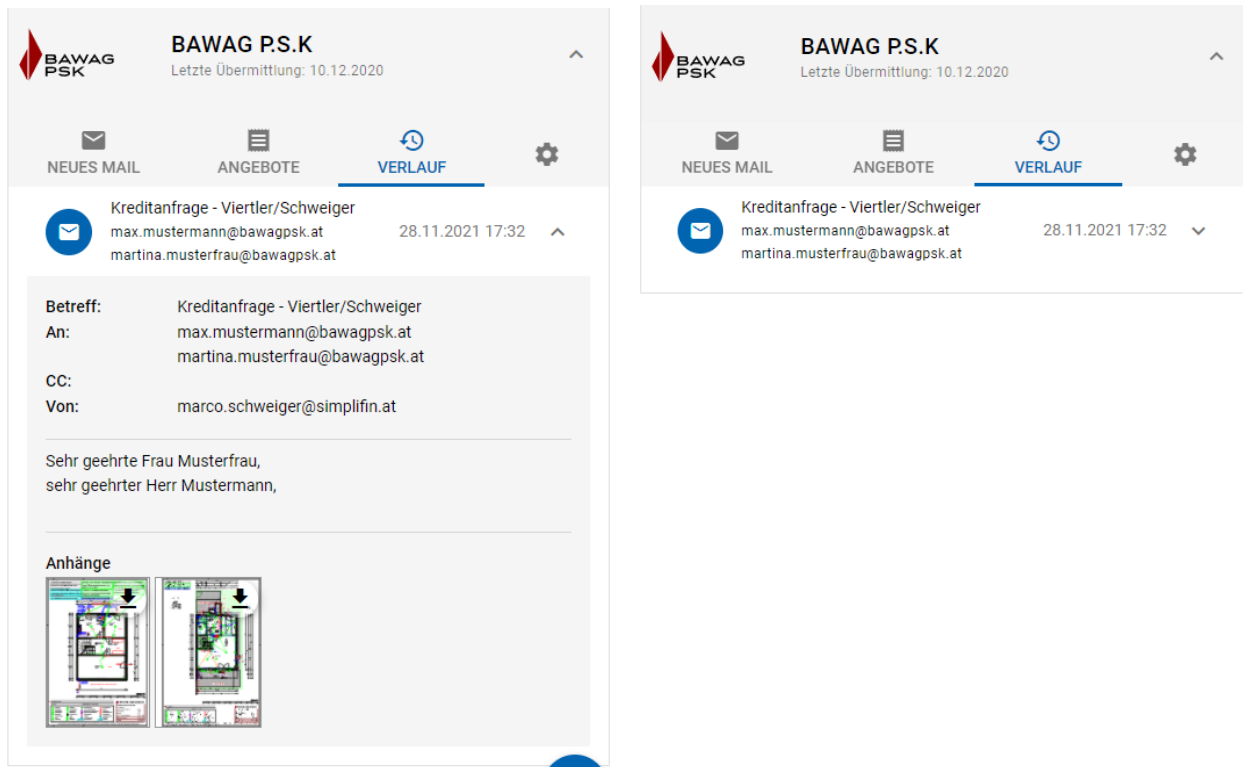


Abbildung 6-23: E-Mail-Verlauf (Eigene Abbildung)

## 6.12 Angebotsvergleich

Nach dem Erhalt der Angebote der Banken müssen die Kreditvermittler/innen die wichtigsten Daten zur Finanzierung, zugehörig zur jeweiligen Anfrage, im Webportal eingeben. Sobald alle Daten vollständig ausgefüllt wurden, wird das Angebot für den/die Kunden/Kundin sichtbar geschaltet (siehe rotes „X“ und grüner Haken in der Abbildung 6-24).

Oft werden je Institut mehrere unterschiedliche Finanzierungsvarianten (verschiedene Laufzeiten, Aufteilung der Kreditsumme in einen fix- und einen variabel-verzinsten Teil usw.) angefragt oder Nachverhandlungen des Angebots durchgeführt. Deshalb können auch mehrere Finanzierungsvarianten bzw. Angebote je Anfrage und Kreditinstitut eingegeben werden.



Abbildung 6-24: Angebot einpflegen (Eigene Abbildung)

Eine besonders wichtige Funktion für Kund/innen ist der Angebotsvergleich, der in Abbildung 6-25 dargestellt ist. In diesem Bereich werden die wichtigsten Daten sämtlicher Angebote übersichtlich dargestellt. Die Angebote können gefiltert und sehr einfach miteinander verglichen werden. Zudem kann das ESIS-Merkblatt bzw. können andere Dokumente des jeweiligen Finanzierungsangebots, für den/die Kunden/Kundin sichtbar, an das Angebot angehängt werden.

Jede Finanzierung ist individuell auf die Bedürfnisse der Kund/innen zugeschnitten. Deshalb erfolgt in diesem Vergleich keine Wertung oder Markierung des laut Effektivzinssatzes besten Angebots. Das passende Angebot wird gemeinsam mit den Kund/innen in einem Beratungsgespräch ermittelt.

The screenshot shows the 'simplifin' interface with a navigation bar (1. Daten, 2. Dokumente, 3. Angebote, Kontakt) and a filter bar (Bank Austria, Raiffeisen, Ing Diba). Below is a comparison table:

	Bank Austria	Raiffeisen	Ing Diba 30 Jahre (20 Fix)	Ing Diba 35 Jahre (20 Fix)
Kreditbetrag	316.000,00 €	319.000,00 €	314.500,00 €	314.500,00 €
Auszahlungsbetrag	308.622,74 €	312.047,50 €	309.691,20 €	309.691,20 €
Gesamtbetrag	381.994,82 €	381.203,49 €	375.915,68 €	386.904,10 €
Gesamtkosten	73.372,08 €	69.155,99 €	66.224,48 €	77.212,90 €
Laufzeit	360 Monate	360 Monate	360 Monate	420 Monate
davon fix verzinst	240 Monate	240 Monate	240 Monate	240 Monate
Sollzinssatz	1,25 %	1,125% u. 0,875%	1,50% u. 0,50%	1,50% u. 0,50%
Effektivzinssatz	1,50 %	1,41% (1,50% u. 1,20%)	1,36% (1,72% u. 0,59%)	1,34% (1,69% u. 0,58%)
Monatsrate	1061,43 €	1065,54 €	1047,63 €	923,73 €
Gültig bis	12.12.2021	18.12.2021	16.12.2021	16.12.2021
Sonstiges		Split: Var: 100.000 (0,875%) Fix: 219.000 (1,125%)	Split: Var: 100.000 (0,50%) Fix: 214.500 (1,50%)	Split: Var: 100.000 (0,50%) Fix: 214.500 (1,50%)
Bemerkungen				
Dateien				

Abbildung 6-25: Angebotsvergleich für Kund/innen (Eigene Abbildung)

## 6.13 Chat-Funktion

Zusätzlich zum telefonischen und persönlichen Kontakt haben Kund/innen die Möglichkeit ihre/n persönliche/n Berater/in direkt über das Portal mit Hilfe einer Chat-Funktion zu kontaktieren. Der Chat ist, was Funktionalität und grundsätzliches Design betrifft, an gängige Messenger wie zum Beispiel WhatsApp angelehnt (siehe Abbildung 6-26). Darüber hinaus ist es möglich, Dateien über die Chat-Funktion auszutauschen.

Der Übertragung der Chatnachrichten erfolgt über eine WebSocket-Verbindung. Dadurch werden Nachrichten, die ein/e Kreditvermittler/in an den Server sendet, sofort an den Client bzw. den/die Kunden/Kundin übertragen und angezeigt.



Abbildung 6-26: Chat-Funktion auf mobilem Endgerät (Eigene Abbildung)

## 6.14 Vermittlung abschließen und Daten löschen

Nach Abschluss bzw. Beendigung der Kreditvermittlung werden die Daten und Dokumente der Kund/innen aus dem Online-Portal gelöscht. Auch wenn größter Wert auf die Sicherheit der Anwendung und Infrastruktur gelegt wird, verringert diese Maßnahme das Risiko von



Datendiebstahl. Zudem ist es nach der Beendigung der Kreditvermittlung nicht mehr erforderlich, dass die Daten online und jederzeit zugänglich sind. Deshalb werden nur die rechtlich unbedingt notwendigen Daten und Unterlagen nach Abschluss der Kreditvermittlung manuell aus dem Portal heruntergeladen und lokal sowie verschlüsselt abgelegt. Anschließend wählt der/die zuständige Kreditvermittler/in aus, dass der Auftrag abgeschlossen werden soll. Um diese Funktion nicht versehentlich auszulösen, muss hierfür zuerst die Checkbox aktiviert werden, erst danach kann der Button „Abschließen“ gewählt werden (siehe Abbildung 6-18 in Kapitel 6.10). Nach einer nochmaligen Bestätigung in einem Pop-Up erhält der/die Kunde/Kundin eine E-Mail mit dem Hinweis, dass seine/ihre Daten innerhalb von 14 Tagen gelöscht werden. Diese Frist gibt Kund/innen die Möglichkeit, ihre Daten herunterzuladen, bevor diese endgültig gelöscht werden.

Den gleichen Ablauf bietet die zweite Funktionalität „Kunden löschen“, nur entfällt hier die Benachrichtigung der Kund/innen. Diese Funktion wird normalerweise verwendet, wenn sich Kunden registriert haben, aber noch nie im Portal aktiv waren.

## **6.15 Fazit der Implementierung des Prototyps**

In diesem Kapitel wurden die wichtigsten Elemente und Bereiche der Anwendung vorgestellt, mit denen das Ziel dieser Arbeit – die digitale Unterstützung des Hypothekarkreditvermittlungsprozesses – erreicht wird. Das entwickelte Webportal kann alle, für den Prototyp definierten, Anforderungen erfüllen. Anhand des Prototyps konnte die Abwicklung einiger Finanzierungen getestet werden. Die integrierten Funktionen decken den notwendigen Funktionsumfang ab und erleichtern die Durchführung der Kreditvermittlung, sowohl für den/die Kunden/Kundin als auch den/die Kreditvermittler/in.

Derzeit können nur Finanzierungsanfragen an Banken, die Anfragen per E-Mail akzeptieren, direkt über das Portal abgewickelt werden. Kreditanfragen an Banken, die die Nutzung ihrer eigenen Software verlangen, können mit dem Prototyp zwar vollständig erfasst, aber nicht automatisiert versendet werden. Für weitere, für die Zukunft wünschenswerte Funktionen siehe Kapitel 7.1.

## 7 ZUSAMMENFASSUNG

Diese Arbeit befasst sich mit der Fragestellung *„Wie lässt sich der Prozess der Hypothekarkreditvermittlung durch ein Webportal sowohl für Kund/innen als auch Kreditvermittler/innen digital unterstützen?“*

Zur Beantwortung dieser Frage wurde ein umfangreicher Prototyp in Form einer Webanwendung sowohl konzeptioniert als auch implementiert. Zunächst wurde der Prozess der Hypothekarkreditvermittlung analysiert und aufbauend darauf wurden die funktionalen Anforderungen und Qualitätsziele definiert. Unter Berücksichtigung dieser Anforderungen wurde daraufhin eine webbasierte Anwendung mit besonderem Fokus auf die leichte Bedienung und Zeitersparnis sowohl für Kund/innen als auch Kreditvermittler/innen entwickelt.

Über das Webportal können sämtliche, für die Finanzierungsabwicklung notwendige, Daten eingegeben werden. Kund/innen werden beim Ausfüllen bestmöglich unterstützt. Beispielsweise werden Kund/innen durch die Haushaltsrechnung geführt, wobei die wichtigsten Kennzahlen automatisch berechnet und angezeigt werden. Während der Eingabe der Projekt- und Finanzierungsdaten werden ebenso notwendige Kennzahlen wie Auszahlungsbetrag, Finanzierungsnebenkosten usw. berechnet.

Um die Daten für Banken in einer übersichtlichen Form zur Verfügung zu stellen, werden aus den Benutzereingaben automatisch PDF-Dokumente mit den erforderlichen Daten generiert. Diese Daten müssen vom/von der Kunden/Kundin vor der Übermittlung an die Banken mittels Unterschrift bestätigt werden. Auf Geräten mit Touchscreen kann diese direkt in der Anwendung gesetzt werden. Banken fordern zusätzlich Belege für die eingegebenen Daten wie zum Beispiel Einkommensnachweise oder Kontoauszüge. Diese Dokumente können direkt über das Portal hochgeladen oder auf mobilen Geräten mit Hilfe der Kamera aufgenommen werden.

Kreditvermittler/innen haben im Portal eine Übersicht über alle Kunden und können jederzeit den Status der Abwicklung einsehen. Kreditanfragen an die Banken können direkt über das Portal mit allen erforderlichen Dokumenten per E-Mail gesendet werden. Durch den Entfall des Zusammensammelns der Dokumente und des manuellen Versendens der Anfragen kann viel Zeit gespart werden. Nach Erhalt der Angebote von den Banken werden diese den Kund/innen direkt im Portal zur Verfügung gestellt.

Der Prototyp erfüllt sämtliche in Kapitel 4 festgelegte Anforderungen. Zudem verringert das Portal den Zeitaufwand für die Kreditabwicklung sowohl auf Seiten der Kund/innen als auch auf Seiten der Kreditvermittler/innen. Durch das Webportal haben alle Beteiligten jederzeit einen Überblick über den aktuellen Status der Kreditabwicklung. Die übersichtliche Aufbereitung der Angebote erhöht die Transparenz und erleichtert Kund/innen die Entscheidung für die bestmögliche Finanzierung.

## 7.1 Ausblick

Der Prototyp ermöglicht, Kreditanfragen an Banken, welche Anfragen per E-Mail akzeptieren, auf einfache Weise zu stellen. Manche Banken verlangen die Nutzung ihrer eigenen Software und stellen derzeit keine Schnittstellen (APIs) für die automatische Eingabe von Kundendaten zur Verfügung. Das bedeutet, dass diese Anfragen derzeit händisch ausgefüllt werden müssen. Zukünftige Arbeiten könnten eruieren, wie das Ausfüllen der Felder dieser bankenspezifischen Anwendungen automatisiert werden könnte und, ob seitens der Banken in absehbarer Zeit Schnittstellen angeboten werden.

Obwohl das Webportal mit besonderem Fokus auf einfache Bedienung entwickelt wurde, könnte eine geführte Einführung – nach dem ersten Login – den Kund/innen die Orientierung und den Einstieg erleichtern.

Hinsichtlich der Sicherheit sollte in Zukunft eine 2-Faktor-Authentifizierung in die Anwendung integriert werden.

Im Prototyp wird nicht zwischen Administrator/innen und Kreditvermittler/innen unterschieden, weshalb jede/r Administrator/in alle Kund/innen sehen kann. Durch Ausbau der Benutzerrollen und Berechtigungen könnte das Portal zukünftig auch externen Kreditvermittlern/-innen zur Verfügung gestellt werden.

Eine sehr interessante Erweiterungsmöglichkeit wären Prognosemöglichkeiten für zu erwartende Kreditkonditionen. Dafür könnten historische Finanzierungsdaten analysiert und kundenindividuelle Prognosen im Portal erstellt werden.

Nach Abschluss des Kreditvertrags endet heute meist die Kundenberatung. Zukünftig könnten die Finanzierungsbedingungen der Kund/innen gespeichert und laufend überprüft werden, ob sich durch die derzeitige Marktsituation Einsparungen durch eine Umschuldung erzielen ließen. Somit könnte man die Kund/innen bei Überschreitung einer bestimmten Schwelle direkt kontaktieren, um eine Umschuldung zu empfehlen.

## ABBILDUNGSVERZEICHNIS

Abbildung 2-1: Entwicklung der aushaftenden Wohnbaukredite privater Haushalte in Österreich (Eigene Abbildung, Datenquelle: Österreichische Nationalbank, 2021).....	4
Abbildung 2-2: Neukreditvergabe an private Haushalte für Wohnbauzwecke in Österreich (Eigene Abbildung, Datenquelle: Österreichische Nationalbank, 2021).....	4
Abbildung 2-3: Nutzung von Online-Banking (Eigene Abbildung, Datenquelle: Bitkom e.V., 2021).....	5
Abbildung 2-4: Ablauf einer Immobilienfinanzierung (In Anlehnung an Eickholt, 2015; Oesterreichische Nationalbank (OeNB), 2004) .....	6
Abbildung 2-5: Zeitersparnis durch das Webportal (Eigene Abbildung) .....	8
Abbildung 5-1: Azure Blob Storage – Separate Container je Kunde/Kundin (Eigene Abbildung) .....	33
Abbildung 5-2: Systemarchitektur (Eigene Abbildung).....	33
Abbildung 5-3: Datenbankstruktur (Eigene Abbildung) .....	34
Abbildung 6-1: Gliederung des zentralen Datenspeichers/Data-Stores (Eigene Abbildung).....	39
Abbildung 6-2: Registrierungsformular (Eigene Abbildung) .....	42
Abbildung 6-3: Passwort-Sicherheitskriterien (Eigene Abbildung).....	43
Abbildung 6-4: Portal-Einstiegsseite/Übersicht (Eigene Abbildung) .....	44
Abbildung 6-5: Portal-Einstiegsseite/Übersicht - Mobile Variante (Eigene Abbildung) .....	45
Abbildung 6-6: Dateieingabe und Dateupload - Persönliche Daten (Eigene Abbildung).....	46
Abbildung 6-7: Live-Berechnung der wichtigsten Kennzahlen (Eigene Abbildung) .....	48
Abbildung 6-8: Haushaltsrechnung (Eigene Abbildung) .....	49
Abbildung 6-9: Links: Erster Kreditnehmer, Rechts: Zweite Kreditnehmerin (Eigene Abbildung) .....	51
Abbildung 6-10: Dateivorschau - PDF-Viewer (Eigene Abbildung).....	52
Abbildung 6-11: Hauptbereich – 2. Dokumente (Eigene Abbildung) .....	53
Abbildung 6-12: Unterschriftsfelder (eigene Abbildung).....	54
Abbildung 6-13: Unterschriftsfunktion auf mobilem Endgerät (Eigene Abbildung) .....	54
Abbildung 6-14: Textfeld-Syntax für die automatische Eingabe (Eigene Abbildung) .....	55
Abbildung 6-15: PDF-Template-Designer (Eigene Abbildung) .....	56
Abbildung 6-16: Technische Ebenen des PDF-Viewers (Eigene Abbildung) .....	57
Abbildung 6-17: Übersicht für Kreditvermittler/innen (Eigene Abbildung).....	58
Abbildung 6-18: Kundendetails für Kreditvermittler/innen (Eigene Abbildung) .....	59
Abbildung 6-19: Datenüberprüfung durch Kreditvermittler/innen (Eigene Abbildung) .....	60
Abbildung 6-20: Anfrage erstellen (Eigene Abbildung) .....	60
Abbildung 6-21: Anfrage bearbeiten – Dokumentenvorlagen (Eigene Abbildung) .....	61
Abbildung 6-22: Kreditanfrage - E-Mail an Bank (Eigene Abbildung) .....	62
Abbildung 6-23: E-Mail-Verlauf (Eigene Abbildung).....	63
Abbildung 6-24: Angebot einpflegen (Eigene Abbildung) .....	63
Abbildung 6-25: Angebotsvergleich für Kund/innen (Eigene Abbildung) .....	64
Abbildung 6-26: Chat-Funktion auf mobilem Endgerät (Eigene Abbildung) .....	65

## **TABELLENVERZEICHNIS**

Tabelle 4-1: Qualitätsziele von Softwareprodukten (In Anlehnung an ISO/IEC 25010, 2011) ..... 30

---

## LISTINGS

Listing 3-1: Beispiel für ein JSON-File.....	11
Listing 3-2: Vue.js-Single File Komponente .....	13
Listing 3-3: v-if und v-else Direktive .....	15
Listing 3-4: v-for-Direktive (In Anlehnung an Street, Passaglia, & Halliday, 2018) .....	15
Listing 3-5: v-bind-Direktive (In Anlehnung an Street, Passaglia, & Halliday, 2018) .....	16
Listing 3-6: v-model-Direktive (In Anlehnung an Steyer, 2019).....	16
Listing 3-7: v-on-Direktive.....	17
Listing 6-1: Store-Objekt global zur Verfügung stellen .....	38
Listing 6-2: MainStore-Gliederung.....	39
Listing 6-3: DataStore: Überwachen der Änderungen und Speichern auf Server (Auszug) .....	41
Listing 6-4: Beispiellink mit dem Verifizierungstoken und der E-Mail-Adresse als Parameter.....	43
Listing 6-5: Dynamisches Ein- und Ausblenden der Felder mit Hilfe der vue-Direktive „v-if“ .....	47
Listing 6-6: Datenbindung an den zentralen Data-Store, um die Daten bei Änderung sofort anzeigen ...	49
Listing 6-7: Files-Komponente.....	50
Listing 6-8: Files-Komponente.....	51

## LITERATURVERZEICHNIS

- Au-Yeung, J. (2021). *Vue.js 3 By Example*. Birmingham: Packt Publishing Ltd.
- Balzert, H. (2009). *Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering*. Heidelberg: Springer Spektrum.
- Bitkom e.V. (2021). *Digital Finance 2021 - Die Transformation der Finanzindustrie in Zahlen*. Berlin: Bitkom e.V.
- Dietz, U. (2016). *Digital Banking*. Berlin: bitkom. Von <https://www.bitkom.org/sites/default/files/file/import/Bitkom-Praesentation-Digital-Banking-06-06-2016-final.pdf> abgerufen
- Ecma International. (2017). *Standard ECMA-404 The JSON Data Interchange Syntax*. Geneva: Ecma International.
- Eglmayr, N., Loretz, E., Meixner, A., Necas, C., Posselt, M., & Siemaszko, S. (November 2018). *WKO - Skriptum Gewerbliche Vermögensberatung und Wertpapiervermittler*. Wien: Fachverband Finanzdienstleister.
- Eickholt, J. (2015). *Nutzung von Online-Immobilienfinanzierungen*. Wiesbaden: Springer Gabler.
- Fabric.js Javascript Canvas Library*. (25. 10 2021). Von <http://fabricjs.com> abgerufen
- Fachverband Finanzdienstleistung. (18. 5 2020). *Die Kreditvermittlung*. Von WKO: <https://www.wko.at/branchen/information-consulting/finanzdienstleister/artikel-kreditvermittlung.pdf> abgerufen
- Google Ireland Limited. (15. 11 2021). *reCAPTCHA*. Von *What is Recaptcha?*: <https://www.google.com/recaptcha/about/> abgerufen
- Heitkötter, H., Majchrzak, T., Wolfgang, U., & Kuchen, H. (2012). *Business Apps: Grundlagen und Status quo*. Münster: Förderkreis der Angewandten Informatik an der Westfälischen Wilhelms-Universität Münster e.V.
- Hellenkamp, D., & Förderer, K. (2016). *Handbuch Bankvertrieb - Theorie und Praxis im Zukunftsdialog*. Wiesbaden: Gabler Verlag.
- ISO/IEC 25010. (2011). *ISO/IEC 25010:2011-03 Software-Engineering - Qualitätskriterien und Bewertung von Softwareprodukten (SQuaRE) - Qualitätsmodell und Leitlinien*.
- Liebel, C. (2019). *Progressive Web Apps - Das Praxisbuch*. Bonn: Rheinwerk Verlag GmbH.

Luecke, D. (04. 12 2021). *FeathersJS*. Von <https://docs.feathersjs.com/> abgerufen

Microsoft Corporation. (25. 10 2021). *Azure Cosmos DB: Nicht relationale Datenbank*. Von Microsoft: <https://azure.microsoft.com/de-de/services/cosmos-db/#overview> abgerufen

Microsoft Corporation. (25. 10 2021). *Introduction to Blob (object) storage - Azure Storage | Microsoft Docs*. Von Microsoft: <https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blobs-introduction> abgerufen

Microsoft Corporation. (01. 12 2021). *Overview - Azure App Service | Microsoft Docs*. Von <https://docs.microsoft.com/en-us/azure/app-service/overview> abgerufen

Microsoft Corporation. (1. 12 2021). *What is Applications Insights? - Azure Monitor | Microsoft Docs*. Von <https://docs.microsoft.com/en-us/azure/azure-monitor/app/app-insights-overview> abgerufen

Mozilla . (25. 10 2021). *PDF.js*. Von <https://mozilla.github.io/pdf.js/> abgerufen

Oesterreichische Nationalbank (OeNB). (2004). *Leitfadenreihe zum Kreditrisiko - Kreditvergabeprozess und Kreditrisikomanagement*. Wien: Oesterreichische Nationalbank (OeNB).

Oesterreichische Nationalbank. (2021). *Immobilien aktuell – International Q3/21*. Wien: Oesterreichische Nationalbank.

Österreichische Nationalbank. (17. 10 2021). *Kreditentwicklung privater Haushalte in Österreich sowie im Euroraum gegliedert nach Verwendungszweck*. Von Österreichische Nationalbank: <https://www.oenb.at/isaweb/report.do?lang=DE&report=1.5.7> abgerufen

Österreichische Nationalbank. (17. 10 2021). *Neukreditvergaben an private Haushalte und nichtfinanzielle Unternehmen in Österreich*. Von Österreichische Nationalbank: <https://www.oenb.at/isaweb/report.do?lang=DE&report=1.5.51> abgerufen

Parallax Agency Ltd. (30. 11 2021). *Parallax*. Von jsPDF - HTML5 PDF Generator: <https://parall.ax/products/jspdf> abgerufen

Reinman, A. (30. 11 2021). *Nodemailer*. Von Nodemailer: <https://nodemailer.com/about/> abgerufen

Shute, Z. (2019). *Advanced JavaScript*. Birmingham: Packt Publishing Limited .

Steyer, R. (2019). *Webanwendungen erstellen mit Vue.js*. Wiesbaden : Springer Fachmedien Wiesbaden GmbH.

Stoenescu, R. (30. 11 2021). *Quasar.dev*. Von Why Quasar? | Quasar Framework: <https://quasar.dev/introduction-to-quasar> abgerufen



Street, M., Passaglia, A., & Halliday, P. (2018). *Complete Vue.js 2 Web Development*. Birmingham: Packt Publishing Ltd.

Walter, T. (2008). *Kompendium der Web-Programmierung*. Berlin Heidelberg: Springer-Verlag.

*What is Vuex.js | Vuex*. (02. 12 2021). Von <https://vuex.vuejs.org> abgerufen

Zammetti, F. (2019). *Practical Flutter: Improve your Mobile Development with Google's Latest Open-Source SDK*. New York: Springer Science+Business Media.