



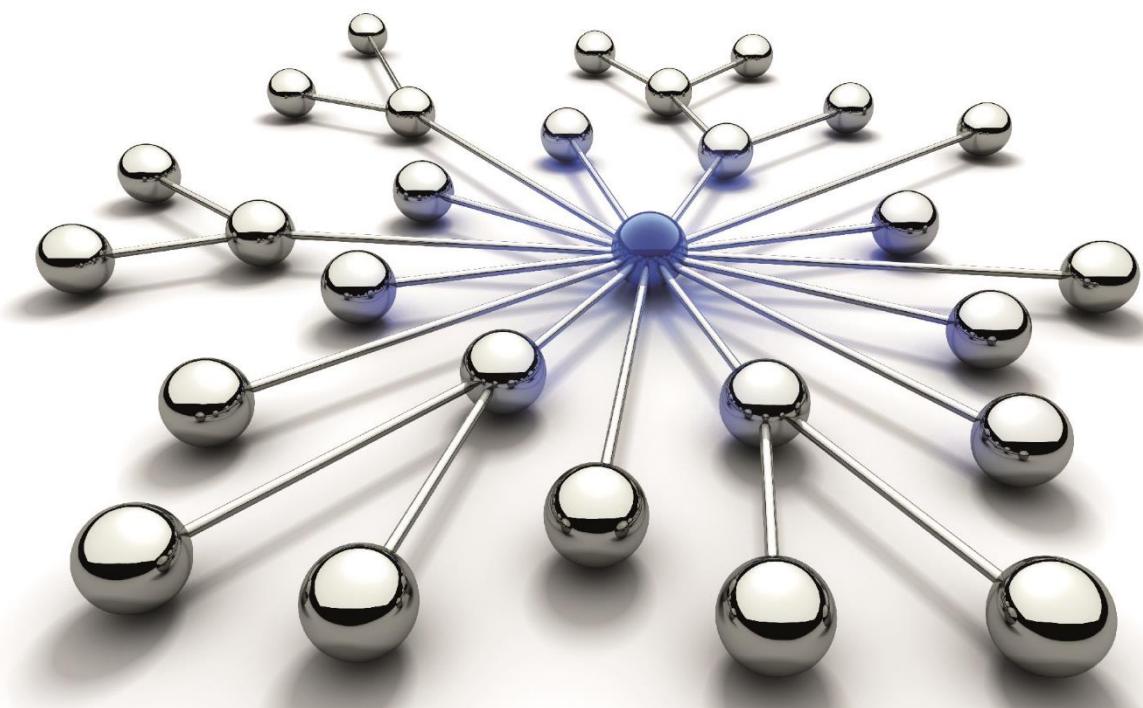
PartnerDashboard

Diplomarbeit

Gernot Krems

Manuel Sander

Johannes Zach





Inhaltsverzeichnis

1.	Eidesstattliche Erklärung.....	12
2.	Kurzfassung	13
3.	Abstract	14
4.	Danksagung	15
5.	Kooperationsvereinbarung mit dem Auftraggeber.....	16
6.	Einleitung.....	19
7.	Grundlagen und Methoden.....	21
7.1.	Vorgehensmodell.....	21
7.2.	Technisches Vorgehen.....	22
8.	Projektantrag.....	24
8.1.	Allgemeine Firmenbeschreibung.....	24
8.2.	Problemstellung.....	25
8.3.	Zielsetzung	25
8.4.	Ablauf.....	26
8.5.	Ergebnis	27
8.6.	Aufgabenstellungen der einzelnen Teammitglieder	28
9.	Projektvorstudie	30
9.1.	Ausgangslage	30
9.1.1.	IST-Prozesse.....	30
9.1.1.1.	Bestellprozess	31
9.1.1.2.	Einrichtungsprozess	31
9.1.1.3.	Verrechnung.....	31
9.1.2.	Anwendungsfälle - Use Case	31
9.1.2.1.	Use Case - Bestellprozess.....	32
9.1.2.2.	Use Case – Einrichtungsprozess.....	32



9.1.2.3. Use Case – Verrechnung	33
9.1.3. Vorabsprachen mit dem Anwender/Auftraggeber	33
9.2. SWOT Analyse.....	33
9.2.1. Stärken (Strengths).....	33
9.2.2. Schwächen (Weaknesses).....	33
9.3. SOLL Zustand	34
9.3.1. Ziele	34
9.4. SOLL - Prozesse	35
9.4.1. Bestellprozess.....	35
9.4.2. Einrichtungsprozess.....	35
9.4.3. Verrechnung.....	35
9.4.4. Domainbestellung	35
9.5. SOLL - Anwendungsfälle - Use Cases	36
9.5.1. Use Case – Bestellprozess.....	36
9.5.2. Use Case – Einrichtungsprozess	36
9.5.3. Use Case – Verrechnung.....	37
9.5.4. Use Case – Domainbestellung	37
9.6. Userstories.....	38
9.7. Variantenentscheide.....	39
9.7.1. Auswahlentscheidung der Programmiersprache.....	39
9.7.2. Auswahlentscheidung der Authentifizierungsmethoden	40
9.8. Requirements on a higher level.....	41
9.8.1. Functional Requirements	41
9.8.2. Non Functional Requirements	41
9.9. Zeitliche Anforderungen.....	41
9.10. Ablaufdiagramm.....	42



10.	Mockups	44
10.1.	Login	44
10.2.	Kundenübersicht, Kunden anlegen	45
10.3.	Kundeninformationen, Produktdetails	45
10.4.	Provisionen, Produkt hinzufügen	46
10.5.	FAQ, News	46
10.6.	Rechnungsübersicht, Benutzerverwaltung	47
11.	Projektplanung	49
11.1.	Meilensteinplan	49
11.2.	Gantt-Diagramm	50
11.3.	Risikoanalyse	50
11.4.	Stakeholder Analyse	51
11.5.	Sprint-Backlog	52
11.6.	Projekt-Retrospektive	53
12.	Verwendete Tools	55
12.1.	Projektplanung	55
12.1.1.	Jira	55
12.1.2.	Gantter for Google Drive	56
12.1.3.	yED	56
12.1.4.	Pencil	56
12.1.5.	Wunderlist	57
12.1.6.	Google Drive	57
12.2.	Projektdokumentation	58
12.2.1.	Microsoft Word	58
12.2.2.	Microsoft Excel	58
12.2.3.	Microsoft PowerPoint	58



12.3.	Entwicklungsumgebung	59
12.3.1.	Atom	59
12.3.2.	Apache	59
12.3.3.	MySQL Workbench	60
12.3.4.	Browser.....	60
12.4.	Versionierungstool.....	61
12.4.1.	Bitbucket.....	61
12.4.2.	Source Tree.....	61
12.5.	Zusätzliche Tools	62
12.5.1.	Skype.....	62
12.5.2.	WhatsApp	62
13.	Technische Realisierung	64
13.1.	Systemarchitektur.....	64
13.1.1.	Entwicklungsumgebung.....	64
13.1.2.	Livesystem	65
13.1.3.	Absicherung mittels Transport Layer Security (TLS).....	66
13.2.	Softwarearchitektur.....	67
13.3.	Klassendiagramm	69
13.3.1.	Namespace „Data“	69
13.3.1.1.	Value Object	69
13.4.	Datenbankdesign	73
13.5.	Beschreibung der Funktionalität.....	74
13.5.1.	Authentifizierung	74
13.5.2.	Kundenmodul	74
13.5.3.	Bestellmodul	75
13.5.4.	Rechnungsmodul	75



13.5.5.	Administrationsmodul	75
14.	Qualitätsmanagement.....	77
14.1.	Programmierrichtlinien.....	77
14.1.1.	Eingesetzte PSRs	77
14.1.2.	Verwendete Design Patterns.....	78
14.2.	Testdurchführung	78
14.2.1.	Komponententest.....	78
14.2.2.	Integrationstest	78
14.2.3.	Systemtest	78
14.3.	Teststrategie	79
14.4.	Komponententests.....	79
14.5.	Integrationstests	84
14.6.	Systemtests	86
14.6.1.	Userstory 1.....	86
14.6.1.1.	Logischer Testfall	86
14.6.1.2.	Konkreter Testfall	86
14.6.1.3.	Testergebnis	86
14.6.2.	Userstory 2.....	86
14.6.2.1.	Logischer Testfall	86
14.6.2.2.	Konkreter Testfall	86
14.6.2.3.	Testergebnis	87
14.6.3.	Userstory 3.....	87
14.6.3.1.	Logischer Testfall	87
14.6.3.2.	Konkreter Testfall	87
14.6.3.3.	Testergebnis	87
14.6.4.	Userstory 4.....	87



14.6.4.1. Logischer Testfall	87
14.6.4.2. Konkreter Testfall	87
14.6.4.3. Testergebnis	88
14.6.5. Userstory 5.....	88
14.6.5.1. Logischer Testfall.....	88
14.6.5.2. Konkreter Testfall	88
14.6.5.3. Testergebnis	88
14.6.6. Userstory 6.....	88
14.6.6.1. Logischer Testfall.....	88
14.6.6.2. Konkreter Testfall	88
14.6.6.3. Testergebnis	89
14.6.7. Userstory 7.....	89
14.6.7.1. Logischer Testfall.....	89
14.6.7.2. Konkreter Testfall	89
14.6.7.3. Testergebnis	89
14.6.8. Userstory 8.....	89
14.6.8.1. Logischer Testfall.....	89
14.6.8.2. Konkreter Testfall	89
14.6.8.3. Testergebnis	90
14.6.9. Userstory 9.....	90
14.6.9.1. Logischer Testfall.....	90
14.6.9.2. Konkreter Testfall	90
14.6.9.3. Testergebnis	90
14.6.10. Userstory 10.....	90
14.6.10.1. Logischer Testfall.....	90
14.6.10.2. Konkreter Testfall	90



14.6.10.3. Testergebnis	90
14.6.11. Testfall: Berechtigungsverwaltung / Partner	91
14.6.11.1. Logischer Testfall	91
14.6.11.2. Konkreter Testfall	91
14.6.11.3. Testergebnis	91
14.6.12. Testfall: Berechtigungsverwaltung - Administrator	91
14.6.12.1. Logischer Testfall	91
14.6.12.2. Konkreter Testfall	92
14.6.12.3. Testergebnis	92
14.6.13. Testfall: Verfügbarkeitscheck	92
14.6.13.1. Logischer Testfall	92
14.6.13.2. Konkreter Testfall	92
14.6.13.3. Testergebnis	92
14.6.14. Testfall: Rechnungsübersicht.....	93
14.6.14.1. Logischer Testfall	93
14.6.14.2. Konkreter Testfall	93
14.6.14.3. Testergebnis	93
14.6.15. Testfall: Produkt kündigen.....	93
14.6.15.1. Logischer Testfall	93
14.6.15.2. Konkreter Testfall	93
14.6.15.3. Testergebnis	93
15. Literatur- und Quellenhinweise	95
16. Abbildungsverzeichnis.....	97
17. Tabellenverzeichnis.....	104
18. Zusammenfassung.....	108
19. Ausblick	110



20. Persönliches Fazit	112
21. Anhang.....	114
21.1. Begleitprotokoll.....	114
21.2. Abnahme Design	115
21.3. Projektantrag	116
21.4. Benutzerhandbuch.....	118
21.4.1. Login.....	118
21.4.2. Two-Factor Authentication.....	118
21.4.3. Startseite.....	119
21.4.4. Navigation.....	119
21.4.5. Dashboard.....	120
21.4.5.1. Kundenübersicht	121
21.4.5.1.1. Neuer Kunde.....	122
21.4.5.1.2. Produktinformationen.....	122
21.4.5.1.3. Kundeninformationen	123
21.4.5.2. Rechnungsübersicht	124
21.4.5.3. Produktübersicht.....	126
21.4.6. News	127
21.4.7. FAQ	127
21.5. Administrationshandbuch.....	129
21.5.1. Basiskonfiguration	129
21.5.2. Dashboard.....	129
21.5.3. Benutzerverwaltung	130
21.5.4. Berechtigungsverwaltung.....	132
21.5.5. Request Engine	133
21.5.6. Requestanfragen	134
21.5.7. News Verwaltung.....	135



21.5.8.	FAQ Verwaltung.....	136
21.6.	Stundennachweis.....	139
21.6.1.	Gernot Krems.....	139
21.6.1.1.	Gesamtübersicht	139
21.6.1.2.	PTD.....	139
21.6.1.3.	Stunden pro Monat	140
21.6.2.	Manuel Sander.....	141
21.6.2.1.	Gesamtübersicht	141
21.6.2.2.	PTD.....	141
21.6.2.3.	Stunden pro Monat	142
21.6.3.	Johannes Zach.....	143
21.6.3.1.	Gesamtübersicht	143
21.6.3.2.	PTD.....	143
21.6.3.3.	Stunden pro Monat	144
21.7.	Sprintstatusberichte.....	146
21.7.1.	Sprint 1.....	146
21.7.2.	Sprint 2.....	148
21.7.3.	Sprint 3.....	150
21.7.4.	Sprint 4.....	152
21.7.5.	Sprint 5.....	154
21.7.6.	Sprint 6.....	156
21.7.7.	Sprint 7.....	158
21.8.	Meeting Protokolle	161
21.8.1.	Kickoff-Meeting	161
21.8.1.1.	Thema 1 - Projektantrag.....	162
21.8.1.2.	Thema 2 - Vorstudie	162



21.8.1.3. Thema 3 - Mockups	162
21.8.1.4. Thema 4 - Datenbank/allgemeine Punkte.....	163
21.8.2. Meeting Schnittstelle.....	164
21.8.2.1. Thema 1 – Schnittstelle Datenbank	165
21.9. Ordnerstruktur.....	167



1. Eidesstattliche Erklärung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Diplomarbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch bei keiner anderen Prüferin/ keinem anderen Prüfer als Prüfungsleistung eingereicht.

Mir ist bekannt, dass Zu widerhandeln mit der Note „nicht genügend“ (ohne Möglichkeit einer Nachbesserung oder Wiederholung) geahndet wird und weitere rechtliche Schritte nach sich ziehen kann.

Diese Arbeit wurde in gedruckter Version bei der zuständigen Prüferin/dem zuständigen Prüfer hinterlegt.

Wolfgang Geissler

Mark Jahn

Johannes Zadach



2. Kurzfassung

Das Unternehmen perfect:net bietet im österreichischen und deutschen Markt individuell angepasstes Webhosting an. Neben den obligaten Webhosting-Paketen für Standard-CMS (Joomla, Typo3, WordPress usw.) und Webshops (xt:commerce, Magento, PrestaShop usw.) besteht die Möglichkeit, basierend auf den Standardpaketen für die Anforderungen der unterschiedlichen Einsatzgebiete und Online-Systeme adaptierte Web Hosts einzurichten.

Ziel ist es, den Betrieb dahingehend zu optimieren, dass sämtliche Werte der Webhosting-Infrastruktur problemlos angepasst werden können und der Betrieb jederzeit und performant gewährleistet werden kann. Ähnlich gestaltet sind die Mail-Hosting-Pakete. Durch die variable Gestaltung ist es möglich, z.B. Baufirmen extrem große Mail-Konten anzubieten, die bei üblichen Providern nicht möglich sind.

Aufgrund der oben genannten Vorteile setzen immer mehr Agenturen und Partner von perfect:net auf die individuell gestalteten Webhosting-Pakete. Für diese Partner soll nun eine Administrationsmöglichkeit, ein Partner-Portal, geschaffen werden, um einige wichtige Funktionen abzudecken.

Wegen der steigenden Anzahl an Kunden und dem damit einhergehenden hohen Betreuungsaufwand ist es sehr schwierig, die Anfragen der Vertriebspartner zeitnah zu beantworten. Dem Partner ist es im Moment nicht möglich, einfache Standardaufgaben ohne Kooperation mit perfect:net zu erledigen. Bei obligaten Anforderungen wie Produktbestellungen, Anlegen von neuen Kunden, Zusenden von Zugangsdaten, Änderungen von Stammdaten, Provisionsberechnungen usw. muss in jedem Fall eine Urgenz von perfect:net folgen, was in Summe zu erheblichem zeitlichen und administrativen Aufwand führt.

Mit dem Partner-Portal soll demnach eine Schnittstelle geschaffen werden, die die einfache Administration, Neubestellung und Verwaltung der Produkte und Kunden sowie eine Provisionsauflistung des jeweiligen Partners ermöglicht.



3. Abstract

The company perfect:net offers individual web hosting solutions in the Austrian and German market. In addition to the obligatory CMS solutions (Joomla, Typo3 and WordPress) and well established web shops (xt:commerce, Magento, PrestaShop), it's possible to set up a customized web infrastructure for highly diverse fields of application.

Due to stated reasons, more and more agencies and partners rely on the individually designed web hosting packages from perfect:net. The objective is to optimize business processes and provide a simple, flexible and reliable utility to maintain all individual solutions at any time and at a high level; an administrative partner portal.

Considering the increasing number of customers and the associated amount of maintenance required, it is very difficult to answer all the inquiries in a timely manner. At the moment, it is not possible for the partner to do even simple tasks without perfect:net staff. Tasks like product-/user management, changes to master data or calculation of commission needs to be done by perfect:net manually, which takes a considerable amount of time and administration. The partner portal is therefore intended to create a global interface that allows easy administration, ordering, administration of products and customers as well as a commission listing of the respective partner.



4. Danksagung

Zunächst möchten wir uns an dieser Stelle bei all denjenigen bedanken, die uns während der Anfertigung dieser Diplomarbeit unterstützt und motiviert haben. Ganz besonders gilt dieser Dank Herrn Prof. MMag. Bernd Hochwarter und Prof. Dipl.-Ing. Dr. Thomas Quaritsch, BSc, die unsere Arbeit und somit auch uns betreut haben. Dank Eurer fachlichen Expertise konntet Ihr uns immer wieder bei unserer Recherche und unseren Fragen unterstützen. Vielen Dank für eure Zeit und Mühen, die Ihr in unsere Arbeit investiert habt.

Daneben gilt unser Dank Max Mustermann, der zahlreiche Stunden lang Korrektur gelesen hat. Er wies auf Schwächen hin und konnte als Fachmann immer wieder zeigen, wo noch Nachbesserungsbedarf bestand.

Des Weiteren gilt unserer Dank Dieter Biernat, MSc, unserem Auftraggeber, ohne den das ganze Projekt nicht möglich gewesen wäre. Nicht zuletzt gebührt unseren Familien großer Dank, ohne deren Verständnis und Unterstützung dieses ganze Unternehmen schon im Vorhinein niemals zustande gekommen wäre. Herzlichen Dank für jedwede Unterstützung und den großen Beistand.



5. Kooperationsvereinbarung mit dem Auftraggeber

Zwischen
perfect:net
vertreten durch Dieter Biernat, MSc
und
PartnerDashboard
bestehend aus Gernot Krems, Manuel Sander, Johannes Zach

PRÄAMBEL

Das Projektteam und die Projektpartnerin/der Projektpartner beabsichtigen gemäß der Verordnung über die abschließenden Prüfungen in den berufsbildenden mittleren und höheren Schulen, BGB. II Nr. 70/2000 vom 24.2.2000, die Planung und Durchführung eines Diplomprojekts, welches die Erstellung eines Programmes zur Kunden- und Produktverwaltung als Ziel hat.

Durch die Zusammenarbeit soll insbesondere den Mitgliedern des Projektteams die Möglichkeit eingeräumt werden, im Rahmen ihrer schulischen Ausbildung bei der Erstellung der Diplomarbeit an die Verhältnisse im technischen Berufsleben herangeführt zu werden, um dabei die in der Schule erworbenen theoretischen Kenntnisse und Fähigkeiten in der Praxis anzuwenden bzw. zu erweitern. Hingewiesen wird in diesem Zusammenhang auf den unentgeltlichen Charakter dieser Vereinbarung.

§ 1 **Gegenstand**

Gegenstand ist die Erstellung von Arbeitsergebnissen zum Thema der Diplomarbeit. Dieses Thema ist der Projektbeschreibung und dem Pflichtenheft zu entnehmen das der Kooperationsvereinbarung beiliegt. Die Projektpartnerin/Der Projektpartner wird jedoch darauf hingewiesen, dass es sich um ein Projekt im Zusammenhang mit der schulischen Ausbildung handelt und daher jede Haftung des Projektteams, insbesondere in Hinsicht auf die Unentgeltlichkeit des Vertrages, ausgeschlossen ist. Nutzungs- und Verwertungsrechte von im Rahmen dieser Vereinbarung erstellten Arbeitsergebnissen stehen der Projektpartnerin/dem Projektpartner sowie dem Projektteam gemeinsam zu.



§ 2

Laufzeit

Die vorliegende Kooperation tritt am 01.09.2016 in Kraft und wird bis zum Ende der Reife- und Diplomprüfung der HTLuVA Pinkafeld abgeschlossen.

§3

Rechte und Pflichten des Projektteams

Die Mitglieder des Projektteams haben das Recht, die Räumlichkeiten der Projektpartnerin/des Projektpartners samt Infrastruktur und EDV-Infrastruktur im für die Projektabwicklung erforderlichen Ausmaß nach vorheriger schriftlicher Genehmigung durch die Projektpartnerin/den Projektpartner mitzubenutzen. Das Projektteam verpflichtet sich, die im Gegenstand genannten Arbeiten sorgfältig und unter möglichster Schonung der Interessen der Projektpartnerin/des Projektpartners durchzuführen. Das Projektteam unterliegt der Betriebsordnung der Projektpartnerin/des Projektpartners. Das Projektteam verpflichtet sich zur Geheimhaltung aller ihm zur Kenntnis gelangenden Geschäfts- und Betriebsgeheimnisse.

§ 4

Rechte und Pflichten der Projektpartnerin/des Projektpartners

Die Projektpartnerin/Der Projektpartner verpflichtet sich, dem Projektteam beratend zur Verfügung zu stehen und alles zu unterlassen, was der Vollendung des Projekts entgegensteht. Die Projektpartnerin/Der Projektpartner verpflichtet sich, dem Projektteam folgende Hilfsmittel zur Verfügung zu stellen: LAMP-Server, TLS-Zertifikat für Livesystem.

Sollte das Projektteam im Rahmen dieser Kooperationsvereinbarung eine Erfindung machen, die nach dem Gebrauchsmustergesetz bzw. dem Patentgesetz (PatG) schützbar ist, gilt diese Erfindung als Diensterfindung im Sinne des PatG und der §§ 6-19 PatG (in der geltenden Fassung) entsprechend. Das Projektteam verpflichtet sich, die Projektpartnerin/den Projektpartner von einer im Rahmen der Kooperationsvereinbarung gemachten Erfindung unverzüglich in Kenntnis zu setzen. Die Projektpartnerin/Der Projektpartner hat daraufhin das Recht, binnen vier Wochen ab dieser Bekanntgabe zu erklären, dass sie/er das Patentrecht für sich beansprucht. In diesem Fall steht dem Projektteam eine entsprechende Vergütung nach den einschlägigen Bestimmungen des PatG (in der geltenden Fassung) zu. Sollte das Projektteam im Rahmen dieser Kooperationsvereinbarung ein Werk schaffen, dem Schutz im Sinne des Urheberrechtsgesetzes zukommt, verpflichtet es sich, die Projektpartnerin/ den Projektpartner davon unverzüglich zu informieren. Die Projektpartnerin/Der Projektpartner hat daraufhin die Möglichkeit, binnen vier Wochen ab dieser Bekanntgabe mit dem Projektteam einen Werknutzungsvertrag abzuschließen.



§ 5

Einsicht und Präsentation

Da die Tätigkeit des Projektteams auch Inhalt bzw. Grundlage der an der HTLuVA Pinkafeld zu erstellenden Diplomarbeit ist, berechtigt die Projektpartnerin/der Projektpartner die zuständigen Organe des Bundes zur Einsicht und Kontrolle, um die in der Verordnung über die abschließenden Prüfungen an den berufsbildenden mittleren und höheren Schulen genannten Aufgaben zu erfüllen. Das Projektteam ist auch berechtigt, Ergebnisse der Diplomarbeit bei der mündlichen Reife- und Diplomprüfung zu präsentieren. Die zuständigen Organe des Bundes sind ihrerseits wiederum gegenüber jedermann zur Hütung der (?) Geschäfts- und Betriebsgeheimnisse der Projektpartnerin/des Projektpartners verpflichtet.

Graz, am 14.09.2016

Dieter Biernat

Projektpartner
(Dieter Biernat)

Pinkafeld, am 17.09.2016

Klaus Gernot Manuel Johannes Zach

Projektteam
(Gernot Krems, Manuel Sander, Johannes Zach)



6. Einleitung

Bei der Firma perfect:net werden zurzeit alle Prozesse im Unternehmen direkt über Herrn Biernat abgewickelt. Es gibt keine Software, die es den Vertriebspartnern unseres Auftraggebers ermöglicht, die Produkte (Webhostings und Domains), die sie den Kunden verkaufen, selbst einzurichten. Es werden die Daten und das Produkt der Neukunden aufgenommen und diese Informationen dann telefonisch oder per E-Mail an den Geschäftsführer übermittelt. Somit muss Herr Biernat jeden Kunden in seinem System manuell anlegen, die Produkte dem Kunden zuweisen und diese danach einrichten. Dass dieser Vorgang in Verbindung mit dem stetigen Wachstum der Firma zu Problemen führt, ist unausweichlich. Zum einen können Fehler beim Einrichten/Anlegen des Produktes/der Kunden passieren, zum anderen ist der Wartungs- und Betreuungsaufwand für eine Person kaum mehr zu stemmen.

Das Ziel unserer Diplomarbeit ist es, eine Vielzahl der Vorgänge im Unternehmen zu automatisieren. Den Partnern von Herrn Biernat soll es über eine Webseite möglich gemacht werden, Kunden anzulegen, Kunden zu verwalten, Produkte zu verkaufen und diese auch einzurichten. Dieser Vorgang der Produktbestellung soll jedoch nicht komplett automatisiert werden. Der Geschäftsführer möchte die Möglichkeit haben, vor der endgültigen Bestellung des Produktes die eingegebenen Daten noch prüfen und gegebenenfalls anpassen zu können. Dieser Vorgang ist aus Sicht der Automatisierung zwar nicht optimal, jedoch möchte Herr Biernat vor allem in der Anlaufphase die eingegebenen Daten kontrollieren. Diese Funktion wird aber nach einer gewissen Eingewöhnungsphase deaktiviert werden, sodass alle Vorgänge dann automatisch abgearbeitet werden.



PartnerDashboard

Grundlagen und Methoden





7. Grundlagen und Methoden

Es gibt die unterschiedlichsten Herangehensweisen an ein IT-Projekt. Die am häufigsten verwendeten sind jedoch das klassische Wasserfall-Modell und das Scrum-Modell. Beide Vorgehensmodelle haben klar ihre Vorzüge, wobei man nicht pauschal sagen kann, welches Modell das Beste ist, da es größtenteils auf das zu realisierende Projekt ankommt. Das Wasserfall-Modell ist im Gegensatz zum Scrum-Modell äußerst starr, weist jedoch eine hohe Planungssicherheit auf. Hierbei werden Projekte in einzelne Phasen (Konzeption, Design, technische Umsetzung, Inbetriebnahme, Support) aufgeteilt. Sobald eine Phase abgeschlossen ist, können darin getroffene Entscheidungen nicht mehr rückgängig gemacht werden. Jede abgeschlossene Phase dient als Fundament für die jeweils nächste. Das Wasserfall-Modell wird vor allem bei Projekten mit konstanten Anforderungen angewendet. Das Scrum-Modell hingegen setzt auf hohe Flexibilität. Dabei fällt jedoch sofort auf, dass wesentlich weniger Struktur als beim Wasserfall-Modell vorhanden ist. Hierbei wird keine langfristige Planung erstellt, sondern Sprints, die in der Regel einer Dauer von einer bis vier Wochen entsprechen. Dabei werden die einzelnen Themen pro Sprint bearbeitet, getestet und abgeschlossen.

7.1. Vorgehensmodell

Jedes Vorgehensmodell hat seine Vor- und Nachteile, weswegen wir uns für eine Mischung aus Wasserfall-Modell und Scrum-Modell entschieden haben.

Die Grundstruktur wurde über das Wasserfall-Modell abgewickelt, um sicherstellen zu können, dass das Datenbankmodell und eine solide Grundstruktur mit den wesentlichen Elementen wie Data Access Objects, Value Objects, Factories sowie Proxies zur Verfügung stehen und die Module in weiterer Folge separiert und einzeln in Sprints abgearbeitet werden können. Das Projekt war aufgeteilt in sieben Sprints zu je vier Wochen (ist das für alle klar, wie das gemeint ist?). In den ersten drei Sprints widmeten wir uns ausschließlich der Grundstruktur, um diese gut strukturiert/erweiterbar aufzubauen, was uns auch gelang.



7.2. Technisches Vorgehen

Zu Beginn des Projektes wurden im ersten Sprint grundlegende Entscheidungen getroffen, welche Entwurfsmuster, Standards und Tools für die Entwicklung herangezogen werden sollten. Da eine Webapplikation erstellt und eine klare Trennung der Komponenten angestrebt wurde, kam das Model-View-Presenter (MVP) Prinzip zum Einsatz. Neben der Konzeption des Datenbankmodells und der Erstellung von Mockups wurde zeitgleich die Entwicklungsumgebung (LAMP) aufgebaut.

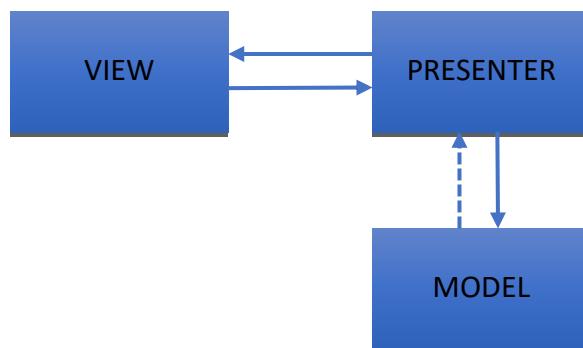


Abbildung 1 - MVP

In den folgenden drei Sprints galt es, die Vorbereitungen für das Model zu treffen und die Grundstruktur aufzubauen. Als Programmiersprache wurde PHP 5.4+ verwendet, ein MySQL Server stellte die Datenbank zur Verfügung. Unter Verwendung der auf PHP-Fig.org gelisteten PHP Standards Recommendations (PSRs) wurden Strukturen, Namespaces und Interfaces definiert sowie weitere Coding Standards eingesetzt.

Anschließend lag der Fokus in den verbleibenden Sprints auf dem Design der einzelnen Module der Webapplikation sowie deren Datenanbindung. Die Kommunikation zwischen Applikation und Server fand mithilfe des AJAX Konzeptes statt.



PartnerDashboard

Projektantrag





8. Projektantrag

Der Projektantrag ist- bei Bewilligung- der Startschuss eines Projektes. Bevor der Projektantrag zustande kommt, müssen die Ziele, der Ablauf, das Ergebnis und die Aufgabenverteilung geklärt sein. In den meisten Fällen wird er durch eine Kooperationsvereinbarung abgeschlossen.

8.1. Allgemeine Firmenbeschreibung



Abbildung 2 - Logo perfect:net

Unique Selling Proposition perfect:net

Das Unternehmen perfect:net bietet auf dem österreichischen und deutschen Markt individuell angepasstes Webhosting an. Neben den obligaten Webhosting-Paketen für Standard-CMS (Joomla, Typo3, WordPress usw.) und Webshops (xt:commerce, Magento, PrestaShop usw.) besteht die Möglichkeit, basierend auf den Standardpaketen für die Anforderungen der unterschiedlichen Einsatzgebiete und Online-Systeme adaptierte Web Hosts einzurichten. Ziel ist es, den Betrieb dahingehend zu optimieren, dass sämtliche Werte der Webhosting- Infrastruktur problemlos angepasst werden können und der Betrieb jederzeit und performant gewährleistet werden kann. Ähnlich gestaltet sind die Mail- Hosting- Pakete. Durch die variable Gestaltung ist es möglich, z.B. Baufirmen extrem große Mail-Konten anzubieten, die bei üblichen Providern nicht möglich sind. Aufgrund der oben genannten Vorteile setzen immer mehr Agenturen und Partner von perfect:net auf die individuell gestalteten Webhosting-Pakete. Für diese Partner soll nun eine Administrationsmöglichkeit, ein Partner-Portal, geschaffen werden, welches vorerst die im Folgenden genannten Funktionen abdecken soll.



8.2. Problemstellung

Aufgrund der steigenden Anzahl an Kunden und dem damit einhergehenden erhöhten Betreuungsaufwand ist es sehr schwierig, die Anfragen der Partner zeitnah zu beantworten. Dem Partner ist es im Moment nicht möglich, einfache Standardaufgaben ohne Kooperation mit perfect:net zu erledigen. Bei obligaten Anforderungen wie Produktbestellungen, Anlegen von neuen Kunden, Zusenden von Zugangsdaten, Änderungen von Stammdaten, Provisionsberechnung usw. muss in jedem Fall eine Urgenz von perfect:net folgen, was in Summe zu erheblichem zeitlichen und administrativen Aufwand führt.

8.3. Zielsetzung

Um möglichst vielen der oben beschriebenen Anforderungen ohne Urgenz von perfect:net-Mitarbeitern gerecht werden zu können, soll ein Werkzeug in Form eines Partner-Portals geplant und umgesetzt werden, welches dem Partner umfangreiche Möglichkeiten bietet.

Nach serverseitiger Authentifizierung stehen folgende Funktionen zur Verfügung:

- **Informativ**
 - Anzeige der bis dato vom Partner angelegten Kunden
 - Automatischer Provisionsbericht zu einem vorgegebenen Zeitintervall (z.B.: Quartal) an eine vordefinierte Partner Email Adresse
 - FAQ (interaktives Manual (Screenshots mit kundenspezifischen Daten), Standard Text)
 - Kontaktformular
- **Administrativ**
 - Anlegen von neuen Kunden
 - Bestellung von Produkten für neue Kunden
 - Bestellung von Produkten für Bestandskunden
- **Technisch**
 - Für sämtliche Aktionen gibt es ein globales Logging, das sämtliche Einträge in einer eigenen Tabelle speichert
 - Hohe Sicherheit wird durch Two-Factor Authentication via Google Authenticator gewährleistet
 - Responsive Design/Adaptierung für Tablets und Smartphones
 - Role-based Access



- **Verrechnung**

- Übersicht der Verrechnungszeiträume für die Produkte der Kunden
- Provisionsauswertungen für Kunden
- Provisionsauswertungen für alle Produkte
- Provisionsauswertungen für einen bestimmten Zeitraum

Mit dem Partner-Portal soll demnach eine Schnittstelle geschaffen werden, die die einfache Administration, Neubestellung und Verwaltung der Produkte und Kunden sowie eine Provisionsauflistung des jeweiligen Partners ermöglicht.

8.4. Ablauf

Login im Partner-Bereich

- Partner loggt sich auf der Webseite mit seinem Account ein (Login, Passwort)
- Die Login-Daten werden serverseitig übergeben und mit der Datenbank verglichen
- Danach erfolgt die 2. Phase der Anmeldung via Google Authenticator

Kunden-Übersicht

- In der Kundenübersicht kann der Partner mittels Web-Formular neue Kunden anlegen
- Eine Übersicht bestehender Kunden wird angezeigt
 - Anzeige der Stammdaten
 - Statusanzeige über den Bestellvorgang
 - Über ein Bestellsystem kann für den jeweiligen Kunden ein neues Produkt bestellt werden.

Produkt-Übersicht

- Über ein Drop-Down-Feld, das automatisch generiert wird, nachdem zuvor sämtliche verfügbaren Produkte aus der Datenbank ausgelesen wurden, kann ein neues, zum Kunden gehöriges Produkt mittels Formular bestellt werden.
- Eine Übersicht bereits für den Kunden bestellter Produkte wird angezeigt:
 - ID des Produktes
 - Name des Produktes
 - Bestelldatum des Produktes
 - Statusanzeige über den Bestellvorgang



8.5. Ergebnis

Durch die Möglichkeit, mit Hilfe des Partner-Portals allfällige Anfragen des Partners effizient, automatisch und ohne Urgenz seitens perfect:net abwickeln zu können, kann eine rasche und zeitlich unabhängige Abhandlung etwaiger Anforderungen erfolgen. Neben der zeitlichen Ersparnis wird dem Partner damit ein Werkzeug geboten, das einen Überblick über bestehende Kunden, deren Produkte und über die Provisionsabrechnung bietet. Zudem wird die Neubestellung, Änderung und Administration von Produkten und Kunden effizient und unkompliziert ermöglicht.



8.6. Aufgabenstellungen der einzelnen Teammitglieder

Manuel Sander (Projektleiter)

- Rechnungsmodul
- Kundenverwaltung
- Backend
- Installationsmodul



Abbildung 3 - Manuel Sander

Krems Gernot

- Login
- Two-Factor Authentication
- Kundenverwaltung Frontend
- Technische Anbindung Backend



Abbildung 4 - Gernot Krems

Johannes Zach

- Externe Schnittstellen
- Access Management
- Order Management
- Online Dokumentation (FAQ)



Abbildung 5 - Johannes Zach



PartnerDashboard

Projektvorstudie





9. Projektvorstudie

In der Vorstudie wird die Ausgangslage analysiert, um einen möglichst umfangreichen Überblick zu bekommen. Danach wird der SOLL-Zustand definiert. Essentiell für die Projektplanung mittels Scrum ist das Einholen der Userstories. Die Vorstudie muss zusätzlich Variantenentscheide, die funktionalen Anforderungen, und zu guter Letzt die zeitlichen Anforderungen enthalten.

9.1. Ausgangslage

Aufgrund der steigenden Anzahl an Kunden und einhergehendem erhöhtem Betreuungsaufwand, ist es sehr schwierig, die Anfragen der Partner zeitnah zu beantworten. Dem Partner ist es im Moment nicht möglich, einfache Standardaufgaben ohne Kooperation mit perfect:net zu erledigen. Bei obligaten Anforderungen wie Produktbestellungen, Anlegen von neuen Kunden, Zusenden von Zugangsdaten, Änderungen von Stammdaten, Provisionsberechnung usw. muss in jedem Fall eine Urgenz von perfect:net folgen, was summa summarum zu erheblichem zeitlichen und administrativen Aufwand führt.

9.1.1. IST-Prozesse

Jede Anfrage wird direkt beim Auftraggeber (Telefon, Email, SMS) gemeldet. Dies hat zur Folge, dass sämtliche Prozesse von einer einzigen Person abhängig sind.

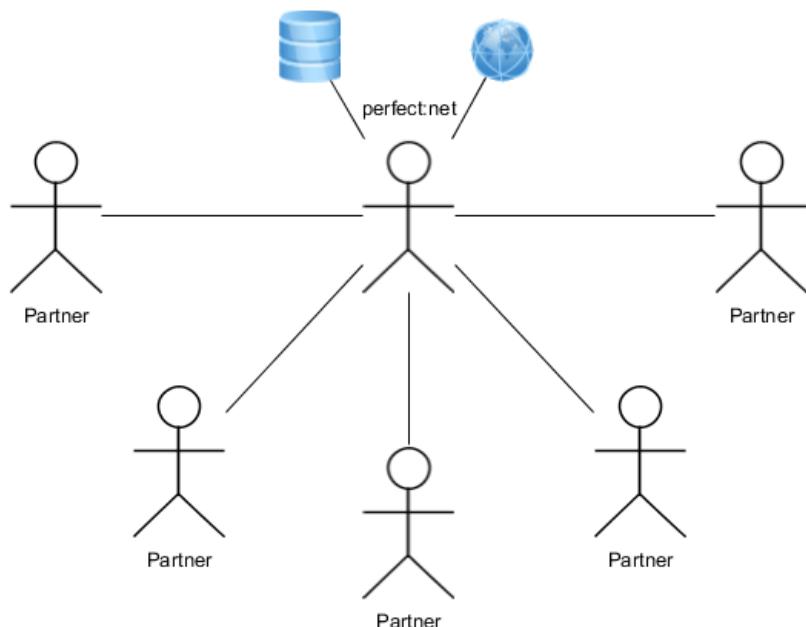


Abbildung 6 - IST-Prozess



9.1.1.1. Bestellprozess

Herr Biernat wird telefonisch oder per Email von den Partnern kontaktiert, welches Produkt er für einen seiner Kunden benötigt. Daraufhin wird dem Partner das Angebot übermittelt, und von ihm letztendlich an den Kunden weitergegeben.

9.1.1.2. Einrichtungsprozess

Die Daten (Username, Passwort, Domain) des Kunden werden manuell in die Datenbank eingetragen und auf dem Webserver provisioniert (Web Host). Zuerst muss hierbei ein Userhandle angefertigt, danach der DNS Eintrag angelegt, und zum Schluss die Domainregistriert werden. Zusätzlich muss alles auch im Backoffice eingetragen werden.

9.1.1.3. Verrechnung

Der Verrechnungszeitraum wird je nach Produkt definiert. Im Backoffice sieht man die zu verrechnenden Produkte, und zwei Mal im Monat ist ein Stichtag, an dem die Verrechnung durchgeführt wird.

9.1.2. Anwendungsfälle - Use Case

Wenn ein Kunde ein neues Produkt benötigt, wendet er sich an einen der Partner von perfect:net. Der Partner nimmt die Bestellung auf und leitet diese an Herrn Biernat weiter. Alle weiteren Schritte, wie die Einrichtung und die Verrechnung, werden direkt von Herrn Biernat erledigt und sind somit alleinig von ihm abhängig.

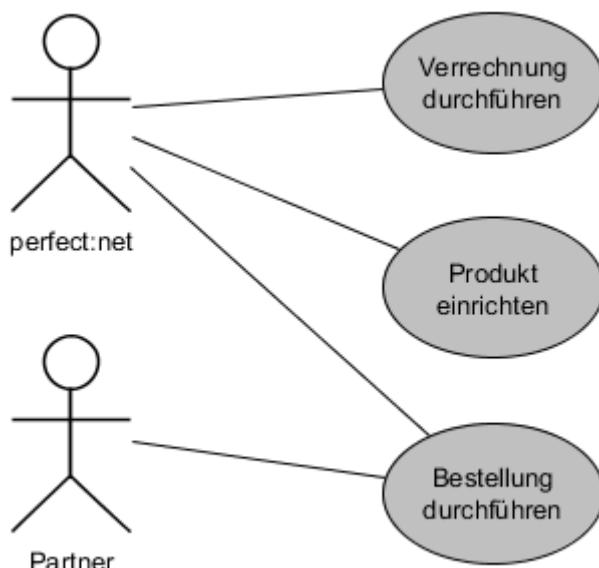


Abbildung 7 - IST-Produktbestellung



9.1.2.1. Use Case - Bestellprozess

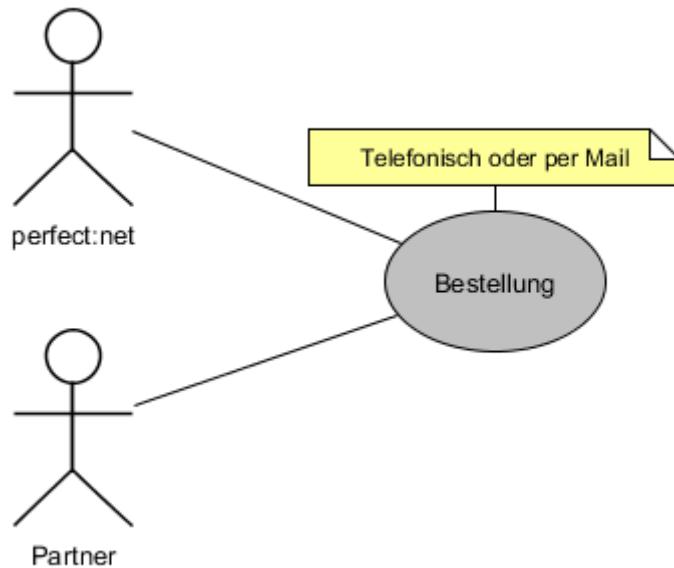


Abbildung 8 - IST-Bestellprozess

9.1.2.2. Use Case – Einrichtungsprozess

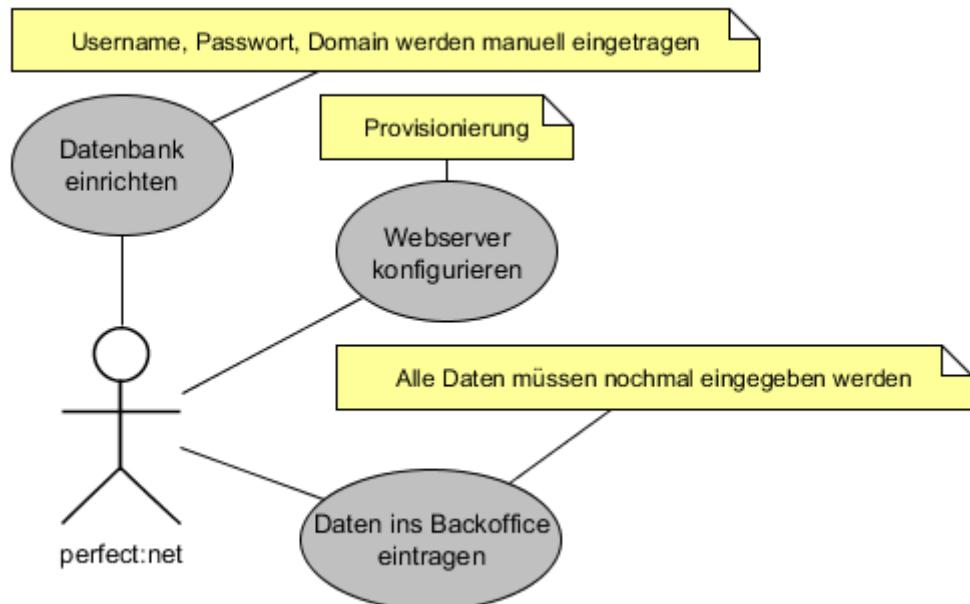


Abbildung 9 - IST-Einrichtungsprozess



9.1.2.3. Use Case – Verrechnung

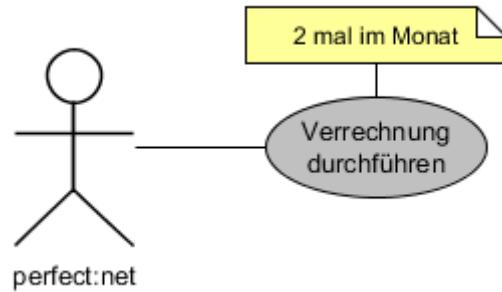


Abbildung 10 - IST-Verrechnungsprozess

9.1.3. Vorabsprachen mit dem Anwender/Auftraggeber

Der Partner hat keine Möglichkeit, seine Kunden und deren Produkte einzusehen. Der komplette Rechnungsversand muss manuell erledigt werden. Das Einpflegen der Daten geschieht noch nicht automatisiert. Alle Daten müssen zuerst in die Datenbank und danach im Backoffice eingetragen werden.

9.2. SWOT Analyse

9.2.1. Stärken (Strengths)

- Qualität des Produktes
- Motivation des Teams
- Klare Aufgabenverteilung
- Nicht auf finanzielle Mittel angewiesen
- Wartungsfreundliches Produkt
- Problemlösungskompetenz
- Zuverlässigkeit
- Kurze Entscheidungswege
- Flexibilität

9.2.2. Schwächen (Weaknesses)

- Managementerfahrung
- Mitarbeitererfahrung
- Service nach Beendigung
- Zeitdruck



9.3. SOLL Zustand

9.3.1. Ziele

- Das Sicherheitsniveau der Partnerauthentifizierung soll durch den zusätzlichen Einsatz von SMS-Passcodes oder dem Google Authenticator erhöht werden, um den Zugriff auf Kunden- und Verrechnungsinformationen durch Dritte zu verhindern.
- Die Zahl der Anfragen und der dadurch entstehende Mehraufwand bei Systemausfällen soll durch eine schnellere Verteilung von Informationen über wichtige Ereignisse gesenkt werden. Eine Anbindung an ein SMS-Gateway bietet die Möglichkeit, Partner schnell und unkompliziert über den aktuellen Status unterrichten zu können.
- Der Wartungsaufwand bestehender Daten soll durch eine Anbindung an das vorhandene IT-System minimiert werden. Durch einen zyklischen Abgleich werden Informationen automatisch aktualisiert und Dateninkonsistenzen vermieden.
- Der Konfigurationsaufwand für Endkundensysteme soll durch die Bereitstellung einer zentralen Wissensdatenbank (FAQ) auf ein Minimum reduziert werden. Partner sollen das Know-how für eine qualifizierte Auskunft über gängige Applikationen/Systeme beziehen können.
- Die sich wiederholende manuelle Erstellung von Provisionsberichten und deren Versand soll durch einen Automatismus ersetzt werden, um den Auftraggeber zu entlasten.
- Eine Bündelung der Kommunikationskanäle soll mit einer zentralen Plattform und einheitlichen Formularen erreicht werden. Anfragen sollen gesammelt dargestellt, priorisiert und protokolliert werden können.



9.4. SOLL - Prozesse

9.4.1. Bestellprozess

Der Partner steigt in das PartnerDashboard ein und legt dort den neuen Kunden und sein Produkt an. Falls der Kunde schon vorhanden ist, muss diesem nur das neue Produkt hinzugefügt werden. Danach ergeht ein Mail an Herrn Biernat mit allen nötigen Daten, und dieser gibt die Bestellung nur noch frei. Bei Sonderpreisen kontaktiert Herr Biernat den Partner direkt um die Details zu klären.

9.4.2. Einrichtungsprozess

Nach der Bestellung durch den Partner bekommt Herr Biernat ein automatisch generiertes Mail mit allen relevanten Daten. Nach einer Kontrolle der Daten gibt er die Bestellung frei und das Produkt wird eingerichtet. Sonderformen müssen wie gehabt manuell eingegeben werden.

9.4.3. Verrechnung

Pro Monat gibt es zwei Verrechnungstage. An diesen Tagen wird ein Prozess angestoßen, der alle zahlungspflichtigen Kunden inkl. Produkt aus der Datenbank ausliest und ein Mail generiert.

9.4.4. Domainbestellung

Automatische Bestellung und Einrichtung der Domain über Hetzner.de mittels Mail-Robot.



9.5. SOLL - Anwendungsfälle - Use Cases

Der Kunde wendet sich weiterhin an den Partner. Dieser meldet sich dann beim PartnerDashboard an und gibt dort die Kundendaten und das Produkt in ein Formular ein, das dann an Herrn Biernat gesendet wird.

9.5.1. Use Case – Bestellprozess

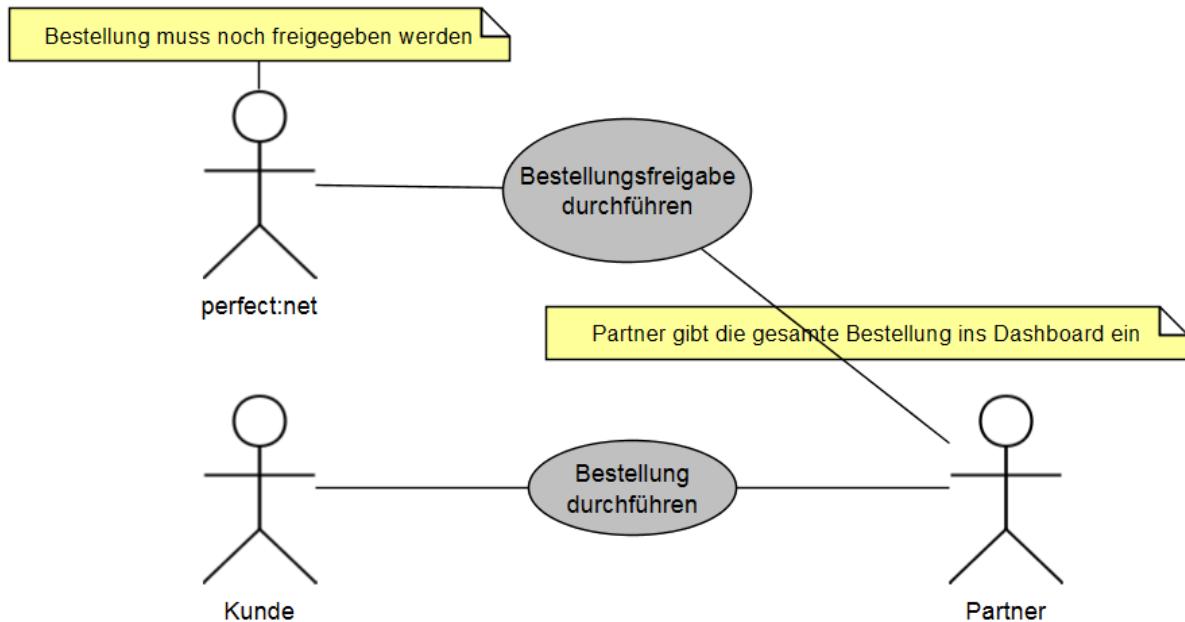


Abbildung 11 - SOLL-Bestellprozess

9.5.2. Use Case – Einrichtungsprozess

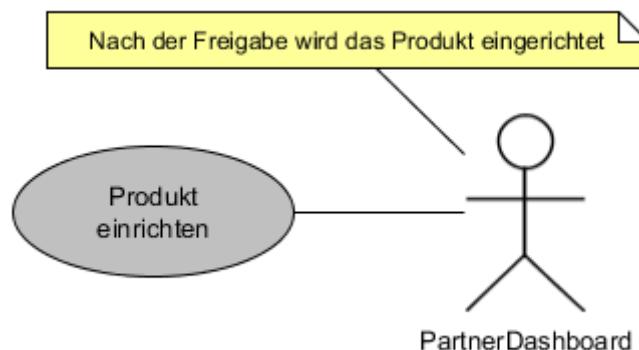


Abbildung 12 - SOLL-Einrichtungsprozess



9.5.3. Use Case – Verrechnung

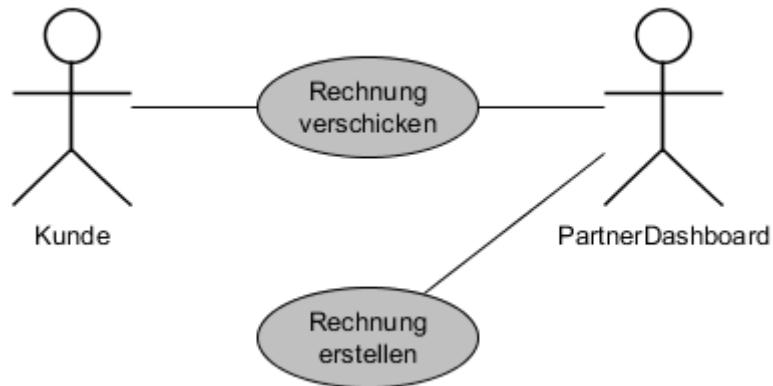


Abbildung 13 - SOLL-Verrechnungsprozess

9.5.4. Use Case – Domainbestellung



Abbildung 14 - SOLL-Domainbestellung



9.6. Userstories

Als	... will ich	damit
Auftraggeber	meinen Partnern schnelle Informationen über Systemausfälle bzw. Wartungsarbeiten liefern können,	präventive Maßnahmen durchgeführt und etwaige Kundenanfragen schneller erledigt werden können.
Auftraggeber	eine Gesamtübersicht der Partner und deren Kunden,	Zahlungen, Provisionen sowie das Einkommen einfach einzusehen sind.
Auftraggeber	eine Übersicht über Anmeldungen und diverse Änderungen im Dashboard (» Log-Funktion),	eine eventuelle Fehleranalyse vereinfacht werden kann.
Auftraggeber	ein hohes Sicherheitsniveau bei der Benutzerauthentifizierung,	der Missbrauch durch Dritte vermieden wird.
Auftraggeber/ Partner	dass das Dashboard auf verschiedenen Medien (PC, Smartphone, Tablet, ...) erreichbar ist,	die Informationen schnell und ständig verfügbar sind.
Auftraggeber/ Partner	eine zentrale Kommunikationsschnittstelle,	der Verlauf der Kommunikation protokolliert wird und Änderungsvorschläge gesammelt werden können.
Auftraggeber/ Partner	die Möglichkeit zur selbständigen Neuanlage und Stammdatenpflege der Kundendaten,	ein Mehraufwand bzw. mögliche Fehlerquellen während der Informationsweitergabe vermieden werden können.
Partner	Benachrichtigungen über Ausfälle und Wartungsarbeiten,	präventive Maßnahmen durchgeführt und etwaige Kundenanfragen schneller erledigt werden können.
Partner	einen zentralen Überblick über meinen Kundenstamm,	allgemeine Kundeninformationen, offene bzw. bereits geleistete Zahlungen sowie eine Produktübersicht einsehbar sind.
Partner	eine Bestellmöglichkeit direkt über das Dashboard,	der Bestellprozess schneller abgewickelt werden kann.
Partner	eine Wissensdatenbank über häufig auftretende Fragen und Probleme (FAQ),	Endkunden bei der Konfiguration mit Rat und Tat zur Seite gestanden werden kann.
Partner	eine Übersicht über alle Produkte und etwaige Aktionen,	eine kompetenter Beratung des Endkunden möglich ist.
Partner	die Möglichkeit für eine aktuelle Provisionsübersicht und einen regelmäßigen Provisionsbericht erhalten,	selbständige Kontrollen durchgeführt werden können und eine Einkommensübersicht vorhanden ist.

Tabelle 1 – Userstories



9.7. Variantenentscheide

Durch Variantenentscheide lassen sich zwei oder mehrere Technologien, Programme usw. durch vorher festgelegte Kriterien und eine Gewichtung miteinander vergleichen, um die – für das Projekt – bestmögliche Entscheidung treffen zu können. Nach dem Festlegen der Gewichtung werden die Produkte anhand des Schulnotensystems verglichen. Als letzten Schritt wird die Gewichtung mit der Note multipliziert sowie die Ergebnisse der einzelnen Kriterien zusammengerechnet.

9.7.1. Auswahlentscheidung der Programmiersprache

Für das vorliegende Projekt bestand die Möglichkeit, es mit den Programmiersprachen JAVA (Variante 1) und PHP (Variante 2) zu realisieren. Als Bewertungskriterien wurden der Installationsaufwand mit einer Wertigkeit von 10 Punkten, die Programmierbarkeit mit einer Wertigkeit von 80 Punkten, das vorhandene Knowhow mit einer Wertigkeit von 70 Punkten, die Linux Kompatibilität mit einer Wertigkeit von 90 Punkten und die Anbindung an ein bestehendes System mit einer Wertigkeit von 100 Punkten herangezogen. Mithilfe einer Nutzwertanalyse wurden die Varianten nach dem Schulnotensystem bewertet.

Kriterien	Gewichtung	Variante 1 - JAVA		Variante 2 - PHP	
		Bewertung	Ergebnis	Bewertung	Ergebnis
Installationsaufwand	10	2	20	3	30
Programmierbarkeit	80	2	160	2	160
Vorhandenes Knowhow	70	3	210	3	210
Linux Kompatibilität	90	1	90	1	90
Anbindung an bestehendes System	100	4	400	1	100
Gesamtbewertung	350	2,4	880	2	590

Tabelle 2 - Variantenentscheid-Programmiersprache

Ergebnis:

PHP



9.7.2. Auswahlentscheidung der Authentifizierungsmethoden

Für das vorliegende Projekt bestand die Möglichkeit es mittels der Authentifizierungsmethoden Google Authenticator (Variante 1) und einer SMS-Authentication (Variante 2) zu realisieren. Als Bewertungskriterien wurden die Wartbarkeit mit einer Wertigkeit von 50 Punkten, die Kosten mit einer Wertigkeit von 100 Punkten, die Benutzerfreundlichkeit mit einer Wertigkeit von 70 Punkten, die Umsetzbarkeit mit einer Wertigkeit von 60 Punkten, die Verfügbarkeit mit einer Wertigkeit von 100 Punkten, die Plattformunabhängigkeit mit einer Wertigkeit von 70 Punkten und ob die Methode zeitgemäß ist mit einer Wertigkeit von 50 Punkten herangezogen. Mithilfe einer Nutzwertanalyse wurden die Varianten nach dem Schulnotensystem bewertet.

		Variante 1 – Google Authenticator		Variante 2 – SMS Authentication	
Kriterien	Gewichtung	Bewertung	Ergebnis	Bewertung	Ergebnis
Wartbarkeit	50	3	150	2	100
Kosten	100	1	100	5	500
Benutzerfreundlichkeit	70	2	140	1	70
Umsetzbarkeit	60	3	180	3	180
Verfügbarkeit	100	1	100	2	200
Plattformunabhängigkeit	70	2	140	1	70
Zeitgemäß	50	1	50	2	100
Gesamtbewertung	500	1,86	860	2,29	1220

Tabelle 3 - Variantenentscheid-Authentication

Ergebnis:

Google Authenticator



9.8. Requirements on a higher level

9.8.1. Functional Requirements

- Bereitstellung der Hardwareanforderungen
- Log
- Planen und Erstellen der Datenbank
- Planung der Anbindung an die bestehende Datenbank
- Erstellen des Webportals
- Erstellen der Auswertungen
- Gewährleistung der sicherheitsrelevanten Aspekte
- Erstellen der FAQs
- Kommunikationswege definieren
- Provisionsberechnung

9.8.2. Non Functional Requirements

- Die Bedienung des Webportals soll den gängigen Standards entsprechen
- Die Sicherheit der erfassten Daten soll gewährleistet sein; dies betrifft den Schutz vor Datenverlust sowie den Zugriff durch Dritte
- Die Performance der Core-Systeme darf nicht beeinträchtigt werden

9.9. Zeitliche Anforderungen

Das Projekt endet im April 2017.



9.10. Ablaufdiagramm

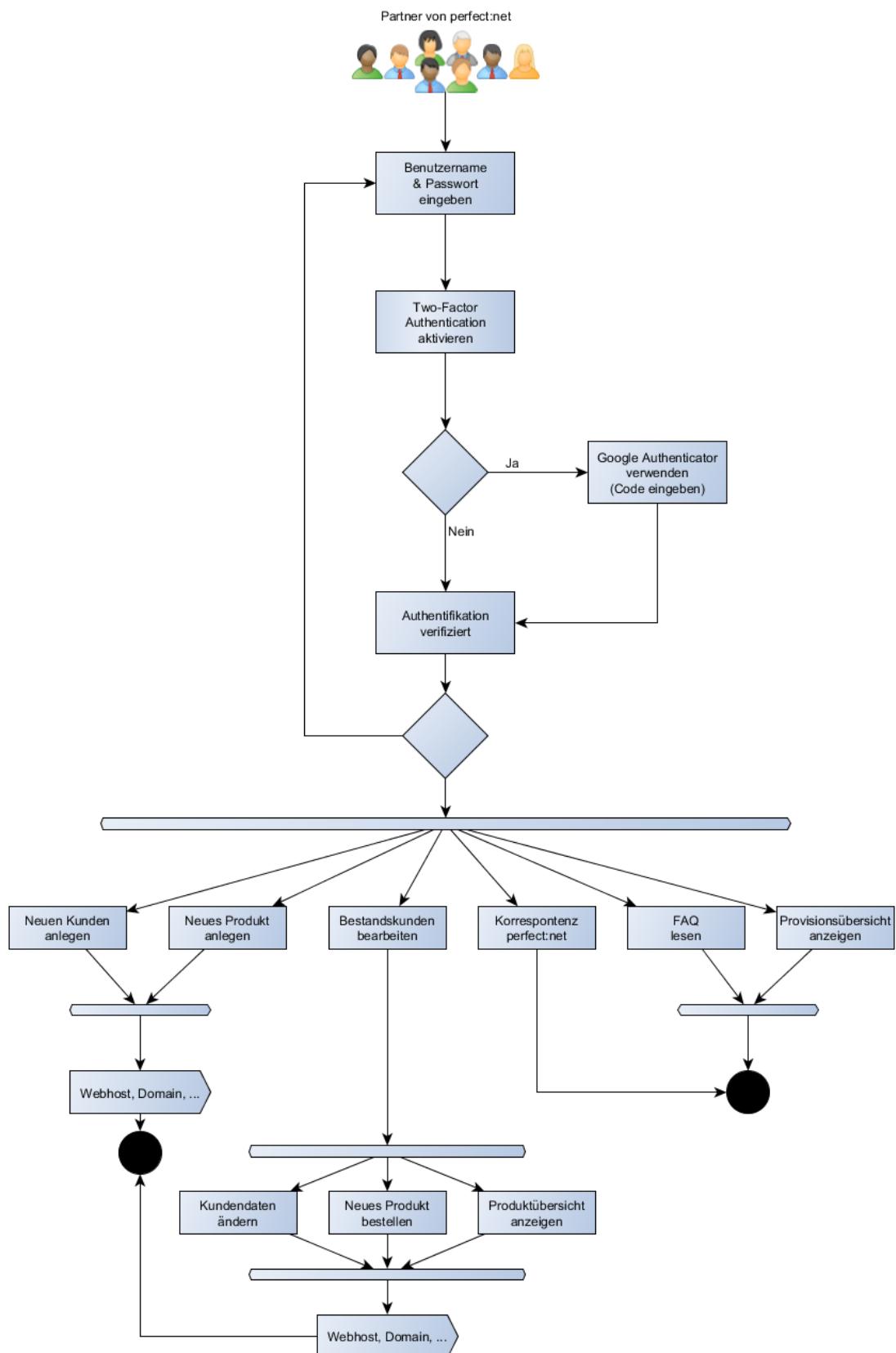


Abbildung 15 - Ablaufdiagramm



PartnerDashboard

Mockups





10. Mockups

Die Mockups dienen – ähnlich wie auch Wireframes – der visuellen und funktionellen Planung von Projekten jeglicher Art. Durch sie wird eine Idee greifbar gemacht. Es ist ein essentieller Schritt in der Planung eines jeden Projekts, da man dem Kunden schon den ersten visuellen Input zu der künftigen Anwendung geben kann. Mockups müssen nicht zwingend 1:1 umgesetzt werden, allerdings müssen Abweichungen mit dem Auftraggeber abgesprochen werden.

10.1. Login

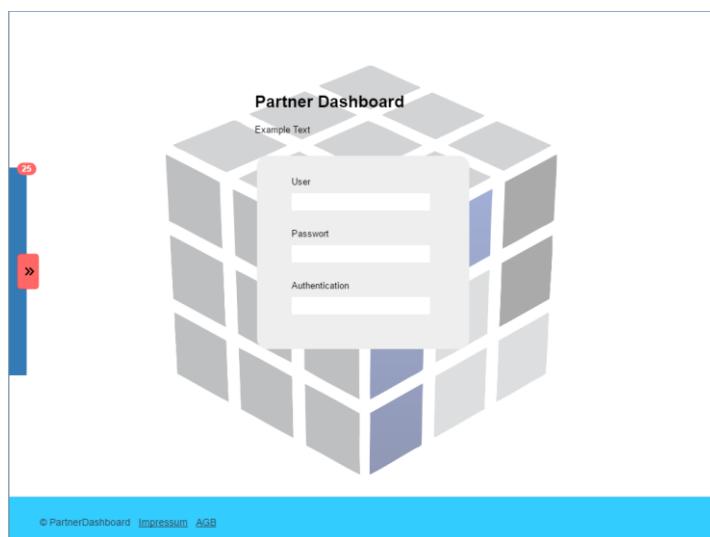


Abbildung 16 - Mockups-Login

Diese Abbildung zeigt den Anmeldeschirm des PartnerDashboards. Eingegeben werden der Username, das Passwort und gegebenenfalls der Code des Google Authenticators.



10.2. Kundenübersicht, Kunden anlegen

Kundenübersicht News Rechnungsübersicht

FAQ • Logout

Hallo Partner

Neuer Kunde // gegebenfalls mit Button löschen

John Doe +
Jane Doe +
Joey Doe +
Jean Doe +

Produkt 1
Produkt 2
Produkt 3

Statistik
Kunden: 95
Produkte: 170
Provision: 370€

+ Previous Next →

© PartnerDashboard Impressum AGB

Abbildung 17 - Mockups-Kundenübersicht

Kundenübersicht News Rechnungsübersicht

FAQ • Logout

Neuer Kunde

Anrede: Herr
Vorname: Test box
Nachname: Test box
Adresse: Test box
PLZ: Test box
Ort: Test box
Land: Test box

Geburtsdatum: Test box
Telefon: Test box
Mobile: Test box
Fax: Test box
VAT: Test box

Zurück Erstellen

© PartnerDashboard Impressum AGB

Abbildung 18 - Mockups-Kunden anlegen

Kundenübersicht: Die Übersicht über alle aktiven Kunden und die dazugehörigen Produkte. Von dieser Ansicht kann zu „Kunden anlegen“, „Kunden bearbeiten“, „Kundeninformationen“ und „Produkt zu Kunden hinzufügen“ navigiert werden.

Neuer Kunde: Neue Kunden anlegen.

10.3. Kundeninformationen, Produktdetails

Kundenübersicht News Rechnungsübersicht

FAQ • Logout

Kundeninformationen

Herr John Doe
Am Sportplatz 18
5474 Oberhausen
Germany
D-12 1985
0161/54532
0699/98238745
john.doe@gmail.at
0161/87237793
VAT: 123456789

Produktübersicht

Produkt 1 Details
Produkt 2 Details

© PartnerDashboard Impressum AGB

Abbildung 19 - Mockups-Kundeninformationen

Kundenübersicht News Rechnungsübersicht

FAQ • Logout

Produktdetails

John Doe

:startUp
300 MB Speicherplatz: 300MB usw.
€4.17 pro Monat

:startUp
600 MB Speicherplatz: 600MB usw.
€4.17 pro Monat

© PartnerDashboard Impressum AGB

Abbildung 20 - Mockups-Produktdetails

Kundeninformationen: Alle Informationen zum ausgewählten Kunden inklusive Produktübersicht.

Produktdetails: Alle Einstellungen zum jeweiligen Produkt.



10.4. Provisionen, Produkt hinzufügen

Provisionen

Diverse Auswertungen

Tabelle Monatsweise

Entwicklung Partner & Produkte

Monat	Partner	Produkte
1	-10	-10
2	-5	-5
3	10	10
4	20	20
5	30	30
6	40	40
7	50	50
8	60	60
9	70	70

Abbildung 21 - Mockups-Provisionen

Produkt hinzufügen		FAQ ▾	Loggen ↗
Professional 4 Proz.			
Professional 4 Proz.			
Light Version	Personal Project	Developer Bundle	Premium Package
FREE	\$99 pro	\$350 pro	\$999 pro
<ul style="list-style-type: none"> <input checked="" type="checkbox"/> 2 Jahre access to all storage locations <input checked="" type="checkbox"/> Unlimited storage <input checked="" type="checkbox"/> Unlimited download quota <input checked="" type="checkbox"/> Cash on Delivery <input checked="" type="checkbox"/> All time updates <input checked="" type="checkbox"/> Unlimited access to all files <input checked="" type="checkbox"/> Allowed to re-use per sale 	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> 2 Years access to all storage locations <input checked="" type="checkbox"/> Unlimited storage <input checked="" type="checkbox"/> Unrestricted download quota <input checked="" type="checkbox"/> Cash on Delivery <input checked="" type="checkbox"/> All time updates <input checked="" type="checkbox"/> Unrestricted access to all files <input checked="" type="checkbox"/> Allowed to re-use per sale 	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> 2 Years access to all storage locations <input checked="" type="checkbox"/> Unlimited storage <input checked="" type="checkbox"/> Unrestricted download quota <input checked="" type="checkbox"/> Cash on Delivery <input checked="" type="checkbox"/> All time updates <input checked="" type="checkbox"/> Unrestricted access to all files <input checked="" type="checkbox"/> Allowed to re-use per sale 	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> 2 Years access to all storage locations <input checked="" type="checkbox"/> Unlimited storage <input checked="" type="checkbox"/> Unrestricted download quota <input checked="" type="checkbox"/> Cash on Delivery <input checked="" type="checkbox"/> All time updates <input checked="" type="checkbox"/> Unrestricted access to all files <input checked="" type="checkbox"/> Allowed to re-use per sale
Download Now!	Purchase as digital	Purchase as digital	Purchase as digital
Or login with	We accept all major credit cards	We accept all major credit cards	We accept all major credit cards

Abbildung 22 - Mockups-Produkt hinzufügen

Provisionen: Übersicht der Provisionen des Partners.

Produkt hinzufügen: Produkte für einen Kunden bestellen.

10.5. FAQ, News

[Kundenübersicht](#) [News](#) [Rechnungsübersicht](#) [FAQ ▾](#) [Logout](#)

Abbildung 23 - Mockups-FAQ



Kundenübersicht News Rechnungsübersicht

FAQ ▾ Logout []

News

30.08.2016 Wartungsarbeiten
Am Sonntag sind Wartungsarbeiten von 20:00 bis 21:00 geplant
News ...
Description.
News ...
Description.

© PartnerDashboard [Impressum](#) [AGB](#)

Abbildung 24 - Mockups-News

FAQ (Frequently Asked Questions): Übersicht von allen häufig auftretenden Fragen. Diese können vom Administrator geändert und erweitert werden.

News: Hier kann Herr Biernat alle für die Partner relevanten Nachrichten schalten. Diese Funktion ist überaus interessant für Wartungsarbeiten, besondere Angebote, oder Systemausfälle.



10.6. Rechnungsübersicht, Benutzerverwaltung

Rechnungsnummer	Kundennummer	Kunde	Datum	Betrag
RE123456	0001	Victor Fleming	21.07.2016	30,99€
RE234567	0002	Hans Wurst	22.07.2016	24,99€
RE345678	0003	Franz Ferdinand	23.07.2016	19,99€

PDF's generieren

Abbildung 25 - Mockups-Rechnungsübersicht

Anrede	Geburtsdatum
Herr	01.01.1980
Vorname	06641234567
Nachname	06641234567
Nachnamenreiter	
Adresse	joh.doe@guard.com
PLZ	Fax
Ort	06641234567
Land	VAT/PIN
Interessat	Telefon
Ostwestfalen	Beratergruppe
	Admin

Abbildung 26 - Mockups-Benutzerverwaltung

Rechnungsübersicht: Hier werden für den angemeldeten Partner alle Rechnungen der Kunden angezeigt. Außerdem wird dargestellt, ob der Kunde seine Rechnungen schon bezahlt hat. Zusätzlich kann der Partner die Rechnungen als PDF downloaden, um diese gegebenenfalls nochmals an den Kunden versenden zu können.

Benutzerverwaltung: Die Benutzerverwaltung zeigt den Administrativen Bereich des PartnerDashboards. Hier können Benutzer (Partner) angelegt sowie Berechtigungen/Berechtigungsgruppen erstellt bzw. zugeteilt werden.



PartnerDashboard

Projektplanung





11. Projektplanung

Die Projektplanung ist eine der wichtigsten Aufgaben beim Projektmanagement. Hier entsteht der Meilensteinplan in dem fixe Meilensteine in dem Projekt definiert werden. Meilensteine und Sprints werden in einem Gantt-Diagramm dargestellt, um das Projekt visualisieren zu können. Weiters gehört zur Projektplanung auch die Risikoanalyse, die Stakeholder Analyse, und zum Abschluss ein Resümee über das gesamte Projekt.

11.1. Meilensteinplan

Meilensteine definieren gewisse Punkte in einem Projekt, an denen Aktivitäten abgeschlossen werden sollen. Mit dem Meilensteinplan soll sichergestellt werden, dass die im Projektplan festgelegten Termin- und Qualitätsanforderungen eingehalten werden.

Nr.	Meilensteinbeschreibung	SOLL-Datum	IST-Datum	Abweichung
1	Projektstart	29.07.2016	29.07.2016	0
2	Anforderungsanalyse abgeschlossen	26.08.2016	25.08.2016	-1
3	Implementierung abgeschlossen	24.03.2017	25.03.2017	1
4	Systemtest abgeschlossen	03.04.2017	03.04.2017	0
5	Projektende	07.04.2017	07.04.2017	0
6	Übergabe der Druckdatei an die Buchbinderei	01.05.2017	28.04.2017	-3
7	Übergabe der Diplomarbeit an den Betreuungslehrer: • Diplomarbeit gebunden in dreifacher Ausführung • DVD: Source Code, Dokumente	05.05.2016	05.05.2017	0

Tabelle 4 - Meilensteinplan



11.2. Gantt-Diagramm

Das Gantt-Diagramm dient der grafischen Darstellung eines Projektes mit den einzelnen Projektphasen (Meilensteine, Sprints). Es ist ein enormer Vorteil für die Übersichtlichkeit eines Projektes und veranschaulicht, was wann zu passieren hat.



Abbildung 27 - Gantt-Diagramm

11.3. Risikoanalyse

„Projektrisiken sind Ereignisse, deren Eintreten negative Auswirkungen auf den Projekterfolg hat. Jedes Risiko ist gekennzeichnet durch die Wahrscheinlichkeit des Eintritts sowie das erwartete Ausmaß des Schadens. Projektorientiertes Risikomanagement identifiziert, klassifiziert und bewertet mögliche Projektrisiken mit dem Ziel, geeignete Maßnahmen zur Bewältigung der Risiken zu entwickeln und durchzuführen.“

Quelle: Systemplanung und Projektentwicklung (MANZ, 2013) (19.04.2017)



11.4. Stakeholder Analyse

Stakeholder sind Personen, die direkt oder indirekt Einfluss auf den Ablauf eines Projektes haben. Durch die Stakeholder Analyse soll genau erklärt werden, wer auf welche Weise an dem Projekt beteiligt ist.

Stakeholder	Bedeutung	Entfernung	Erwartungen an das Team	Erwartungen vom Team an den Stakeholder	Konflikte	Potentiale	Strategie
Auftraggeber	1	1	planmäßige Abwicklung des Projekts, erfolgreicher Projektabschluss, Statusberichte	detaillierte Angaben, Verfügbarkeit, Bereitschaft zur Unterstützung	Nichteinhaltung der Termine, unzureichende Umsetzung der Angaben, minderwertige Informationen	Bereitstellung der Testumgebung, Vermittlung von technischem Knowhow	Regelmäßige Statusupdates, Treffen nach Bedarf
Partner	1	2	zuverlässige Umsetzung des Projekts, Umsetzung der Anforderungen	planmäßige Abwicklung des Projekts, erfolgreicher Projektabschluss	zu hohe Erwartungshaltung, kein Feedback	neue Ideen, Verbesserung der Usability	Befragungen
Projektteam	1	1			Streit, Kommunikationsprobleme, private Probleme, Überlastung		regelmäßige Meetings, Team Building Maßnahmen, konkrete Arbeitsverteilung,
Projektbetreuer	1	1		Informationen und Umsetzung von Vorschlägen	Meinungsverschiedenheiten, schlechte Kommunikation	beratende Funktion	laufende Informationen und Abstimmungen
Techniker	3	4	leicht wartbares System, Dokumentation	fachliche Unterstützung, Ideenlieferant	zu hohe Erwartungshaltung, kein Feedback	neue Ideen, Verbesserung der Usability	Befragungen

Tabelle 5 - Stakeholder Analyse



11.5. Sprint-Backlog

In der nachfolgenden Grafik finden sich alle Userstories und Tasks des Projektes.

All Issues Switch filter ▾	
Order by Created ▾ ▾	
<input checked="" type="checkbox"/> PAR-35	Rechnungsmodul
<input checked="" type="checkbox"/> PAR-34	Benutzer-/Berechtigungsmanagement
<input checked="" type="checkbox"/> PAR-33	Frontend Basis
<input checked="" type="checkbox"/> PAR-32	Tests
<input checked="" type="checkbox"/> PAR-31	AJAX dispatcher
<input checked="" type="checkbox"/> PAR-30	Proxys
<input checked="" type="checkbox"/> PAR-29	Interfaces
<input checked="" type="checkbox"/> PAR-28	DAOs
<input checked="" type="checkbox"/> PAR-27	Factorys
<input checked="" type="checkbox"/> PAR-26	Value Objects
<input checked="" type="checkbox"/> PAR-25	Grundstruktur
<input checked="" type="checkbox"/> PAR-21	Value Objects
<input checked="" type="checkbox"/> PAR-20	Factorys erstellen
<input checked="" type="checkbox"/> PAR-19	SQL Installationsdatei
<input checked="" type="checkbox"/> PAR-18	Login
<input checked="" type="checkbox"/> PAR-17	DAO's UMBENENNEN
<input checked="" type="checkbox"/> PAR-16	Value Objects
<input checked="" type="checkbox"/> PAR-15	Datenbank fertigstellen
<input checked="" type="checkbox"/> PAR-14	Mockups fertigstellen
<input checked="" type="checkbox"/> PAR-13	Provisionsauswertung
<input checked="" type="checkbox"/> PAR-12	Übersicht über Aktionen und Produkte
<input checked="" type="checkbox"/> PAR-11	FAQ's
<input checked="" type="checkbox"/> PAR-10	Bestellungen abwickeln
<input checked="" type="checkbox"/> PAR-9	Übersicht über Kunden
<input checked="" type="checkbox"/> PAR-8	Benachrichtigungen
<input checked="" type="checkbox"/> PAR-7	Kunden anlegen und verwalten
<input checked="" type="checkbox"/> PAR-6	Kommunikationsschnittstelle
<input checked="" type="checkbox"/> PAR-5	Responsive Design
<input checked="" type="checkbox"/> PAR-4	Two-Factor Authentication
<input checked="" type="checkbox"/> PAR-3	Log-Funktion
<input checked="" type="checkbox"/> PAR-2	Übersicht über Partner und deren Kunden
<input checked="" type="checkbox"/> PAR-1	Informationen schnell weitergeben

Abbildung 28 - Sprint-Backlog



11.6. Projekt-Retrospektive

Wir haben bei dem PartnerDashboard einen enormen Zeitaufwand in die Grundstruktur investiert und dadurch wurden die Sprintziele oftmals zu hoch angesetzt. Im Laufe des Projekts wurde unsere Planung jedoch immer präziser, was größtenteils auf die abgeschlossene Grundstruktur zurückzuführen war, da wir dadurch die einzelnen Module besser abgrenzen konnten. Mit dem Beginn der Arbeit an der Website war es hingegen wieder schwieriger, klare Trennlinien zu finden. Wenn wir in einem Sprint nicht alle dazugehörigen Ziele abschließen konnten, wurden diese spätestens im darauffolgenden Sprint erfolgreich abgeschlossen, und letztendlich wurden dann auch alle Ziele erreicht.

Jedes Teammitglied war erpicht darauf, die einzelnen Meilensteine fristgerecht abzuschließen, was uns auch bei fast allen gelang.

Ein essentieller Punkt jedes Projekts ist das Vorgehensmodell der Projektplanung. Hierbei haben wir uns für eine Mischung aus dem Wasserfall-Modell und dem Scrum-Modell entschieden, was sich für unser Projekt als perfekte Mischung herausstellte. Wir waren uns sehr schnell einig, dass wir die Grundstruktur über das Wasserfall-Modell abwickeln wollten, da dies das Fundament des PartnerDashboards ist. Für die einzelnen Module (Bestellmodul, Rechnungsmodul, Kundenmodul usw.) hingegen bot sich das Scrum-Modell an, da man hierbei die Problemstellungen gut separieren und einzeln abarbeiten konnte.

Das Team legte von Anfang an sehr viel Wert auf die Kommunikation. Wir hatten wöchentlich ein Team-Meeting über Skype, um den Fortschritt und die weiteren Schritte zu besprechen. Das gelang uns bis zu den letzten beiden Sprints, denn ab diesem Zeitpunkt kam uns allen die Schule und der Beruf etwas in die Quere, was eine Abhaltung von regelmäßigen Meetings unmöglich machte.

Zusammenfassend ist zu sagen, dass jeder von uns viele Erfahrungen im Bereich der IT-Projekte sammeln konnte. Auch wurden wir damit konfrontiert, dass man nicht alle Eventualitäten bei der Projektplanung berücksichtigen kann. Im Endeffekt ist genau das der Mehrwert solcher Projekte, da man mit Problemen konfrontiert wird, die man vorher niemals vermutet hätte.



PartnerDashboard

Verwendete Tools





12. Verwendete Tools

Um eine moderne Software-Anwendung erfolgreich und komfortabel im Team entwickeln zu können, benötigt man einiges an Zusatzsoftware. Die nächsten Punkte zeigen die Tools, die wir verwendet haben, aufgeteilt in die einzelnen Projektbereiche.

12.1. Projektplanung

Für eine erfolgreiche Projektplanung benötigt man eine zuverlässige Software, in der zum einen die Sprints abgebildet, zum anderen aber auch alle Userstories und Tasks eingetragen werden können. Für diesen Teil haben wir Jira verwendet, da man damit hervorragend alle Aspekte des Scrum-Modells abbilden und verwalten kann. Für den Projektplan wurde Gantter for Google Drive verwendet. Für das Zeichnen der Mockups, des ER-Diagrammes und der Use-Case Diagramme haben wir yED und Pencil eingesetzt. Weiters haben wir für kleinere Tasks, Termine und Erinnerungen das Tool Wunderlist verwendet. Für die gemeinsame Ablage der Dokumente haben wir uns für Google Drive entschieden.

12.1.1.Jira



Abbildung 29 - Logo-Jira

Jira wurde von Atlassian entwickelt und ist für das Projektmanagement eine überaus angesehene Software, die unter anderem von SAP, IBM, Microsoft und BMW genutzt wird. Was dieses Produkt besonders attraktiv macht, ist der Umstand, dass kostenlose Lizenzen für Einzelpersonen, gemeinnützige Einrichtungen und Open Source Projekte angeboten werden.



12.1.2. Gantter for Google Drive



Abbildung 30 - Logo-Gantter

Gantter for Google Drive ist eine kostenlose Alternative zu Tools wie MS-Project. Der Funktionsumfang reicht natürlich bei weitem nicht an eine professionelle Software heran, jedoch sind die Grundfunktionen abgedeckt, und das Arbeiten im Team wird durch Google Drive gewährleistet.

12.1.3. yED



Abbildung 31 - Logo-yED

Mit yED kann man hervorragend Diagramme erstellen und bearbeiten. Es werden die unterschiedlichsten Diagrammarten sowie Use-Case-Diagramme, Ablaufdiagramme, ER-Diagramme, Mind-Maps usw. unterstützt. Mit yED bekommt man kostenlos eine großartige Software.

12.1.4. Pencil



Abbildung 32 - Logo-Pencil

Auf die Open-Source Software Pencil sind wir gestoßen, als es an das Zeichnen der Mockups ging. Mit Pencil lassen sich komplexe und professionelle Mockups/Wireframes realisieren. Es können Frameworks wie Bootstrap und diverse Collections zusätzlich zu den Standardkomponenten eingebunden werden, was natürlich die Mockups authentischer wirken lässt.



12.1.5. Wunderlist



Abbildung 33 - Logo-Wunderlist

Wunderlist ist in der Basisversion kostenlos nutzbar. Man kann ToDo-Listen für ein Team freigeben und Aufgaben mit Fertigstellungsdatum zuweisen. Außerdem können Kommentare, Dateien und Subtasks zu den einzelnen Tasks hinzugefügt werden.

12.1.6. Google Drive



Abbildung 34 - Logo-Google Drive

Bei dem Filehosting-Dienst Google Drive haben wir alle Dokumente, Vorlagen, Stundennachweise usw. abgelegt, um diese – durch freigegebene Ordner – gemeinsam verwalten zu können.



12.2. Projektdokumentation

Für die Dokumentation der Diplomarbeit haben wir das Office Paket von Microsoft verwendet. Schriftstücke wie Projektantrag, Vorstudie, Sprintberichte und letztendlich die Diplomarbeit wurden mit Microsoft Word erstellt. Für den Stundennachweis wurde Microsoft Excel und für die Präsentation der Diplomarbeit Microsoft PowerPoint verwendet.

12.2.1. Microsoft Word



Abbildung 35 - Logo-Microsoft Word

Großartiges Textverarbeitungsprogramm der Firma Microsoft. Eignet sich auch für das Erstellen von umfangreichen Dokumenten. Word ist ohne Zweifel die Standardsoftware für jegliche Art von Textverarbeitung.

12.2.2. Microsoft Excel



Abbildung 36 - Logo-Microsoft Excel

Excel ist das am weitesten verbreitete Tabellenkalkulationsprogramm. Es können sowohl komplexe Berechnungen als auch diverse Statistiken und Auswertungen damit realisiert werden. Dieses Programm ist ebenfalls Bestandteil der Office-Suite.

12.2.3. Microsoft PowerPoint



Abbildung 37 - Logo-Microsoft PowerPoint

Ein weiteres Produkt aus der Office-Suite, das für das Erstellen und Vorführen von Präsentationen konzipiert wurde.



12.3. Entwicklungsumgebung

Das PartnerDashboard ist eine Web-Anwendung. Zu den verwendeten Sprachen zählen PHP, HTML, CSS, JS für die Entwicklung sowie SQL für die Datenbank. Da es dafür nur bedingt Entwicklungsumgebungen gibt, haben wir uns für den Atom Texteditor entschieden. Für die Verwaltung der Datenbank haben wir MySQL Workbench verwendet. Das Testen der Applikation erfolgte mit diversen Browsern. Als grundlegendes Web-Service wurde Apache eingesetzt.

12.3.1. Atom



Abbildung 38 - Logo-Atom

Diesen freien Texteditor kann man nach seinen eigenen Vorstellungen optimal anpassen. Er wird beworben mit „A hackable text editor for the 21st Century“, und das ist er auch. Es gibt unterschiedlichste Themes und Packages zur Erweiterung. Die Versionierung funktioniert problemlos, und er hat auch Syntax-Highlighting für sehr viele Sprachen.

12.3.2. Apache



Abbildung 39 - Logo-Apache

Apache ist ein leistungsstarker Open-Source Webserver, der für mehrere Plattformen zur Verfügung steht. Aufgrund seines modularen Aufbaus wurde Apache zu einem der wichtigsten HTTP-Server in der heutigen Zeit.



12.3.3. MySQL Workbench



Abbildung 40 - Logo-MySQL Workbench

Mit der MySQL Workbench kann man die MySQL Datenbank auf eine sehr benutzerfreundliche Art verwalten. Es können ER-Modelle erstellt und direkt in die Datenbank übertragen werden. Daten können über die grafische Oberfläche eingefügt, geändert oder gelöscht werden. Grundsätzlich können alle Einstellungen der Datenbank über die grafische Oberfläche bearbeitet werden.

12.3.4. Browser



Abbildung 41 - Logo-Mozilla Firefox



Abbildung 42 - Logo-Google Chrome



Abbildung 43 - Logo-Microsoft Edge



Abbildung 44 - Logo-Microsoft Internet Explorer

Die Evaluierung der Applikation erfolgte mit Google Chrome, Mozilla Firefox, Microsoft Edge und Microsoft Internet Explorer.



12.4. Versionierungstool

Damit das Arbeiten in einem Team erfolgreich verläuft, bedarf es der Versionierung. Wir haben uns entschieden, dafür Bitbucket zu verwenden. Für den sauberen Abgleich der Daten haben wir Source Tree eingesetzt.

12.4.1. Bitbucket



Abbildung 45 - Logo-Bitbucket

Bitbucket ist ein weiteres Produkt der Firma Atlassian, das wir verwenden. Es ist ein webbasierter Filehosting-Dienst für die Entwicklung von Software. Die Versionierungssysteme Git und Mercurial werden unterstützt.

12.4.2. Source Tree



Abbildung 46 - Logo-Source Tree

Source Tree wurde ebenfalls von Atlassian entwickelt. Es ist ein Git-Client der sowohl für Windows als auch Mac gratis zur Verfügung gestellt wird. Dort geht es zum einen um die Visualisierung der Entwicklung und zum anderen um die Verwaltung der Repositories.



12.5. Zusätzliche Tools

Um die Kommunikation im Team aufrecht zu erhalten, haben wir größtenteils Skype und WhatsApp verwendet. Skype war enorm wichtig für die Team-Meetings, um dort mittels Screen-Sharing Entscheidungen über Designs und andere Dinge schnell abwickeln zu können. Über WhatsApp wurden die Termine für die Meetings ausgemacht oder kleinere Tasks und Informationen im Team verteilt.

12.5.1. Skype



Abbildung 47 - Logo-Skype

Skype, 2003 gegründet und nach mehrmaligem Besitzerwechsel 2011 von Microsoft gekauft, ist ein kostenloser Instant-Messaging-Dienst. Die Funktionen reichen von Screen-Sharing über Dateiübertragung zu einfacher Telefonie und Videotelefonie.

12.5.2. WhatsApp



Abbildung 48 - Logo-WhatsApp

WhatsApp ist ebenfalls ein Instant-Messaging-Dienst, der seit dem Jahr 2014 zu Facebook gehört.



PartnerDashboard

Technische Realisierung





13. Technische Realisierung

Die Systemarchitektur und die Softwarearchitektur sind der Grundstock der technischen Realisierung, da hier geklärt wird, welche Hard- und Softwareanforderungen an das System gestellt werden. Weiters wird hier das Klassendiagramm und das Datenbankdesign aufgezeigt. Abschließend werden die einzelnen Module des Projekts erklärt.

13.1. Systemarchitektur

In der ersten Phase der Entwicklung setzen wir auf einen Raspberry Pi 2 Model B mit einem LAMP (Linux, Apache, MySQL und PHP) Server. Nachdem die erste Phase abgeschlossen war, wechselten wir auf die Hardware unseres Auftraggebers. perfect:net hostet mehrere dedicated Server bei der Firma Hetzner Online GmbH.

13.1.1. Entwicklungsumgebung

Raspberry Pi 2 B

- ARM Cortex-A7 Quad 900MHz
- 1024 MB RAM
- 10/100 LAN



Abbildung 49 - Raspberry Pi 2B



Abbildung 50 - LAMP



13.1.2. Livesystem

Dell PowerEdge R430 Server

- 2 x Intel E5 Multi Core
- 64 GB RAM
- RAID 5 mit Dell 600 GB 10K SAS 3.5
- 2x Gigabit LAN



Abbildung 51 - Dell Server

Als Betriebssystem kommt ein Linux Debian 6.0 zum Einsatz. Darauf läuft ein LAMP Server mit Apache 2.2.16 -1~dotdeb.0, MySQL 5.1.73 und PHP 5.4.31.



13.1.3. Absicherung mittels Transport Layer Security (TLS)

Um die Daten unseres Auftraggebers nicht in Klartext über das Internet zu transportieren, entschlossen wir uns für Hypertext Transfer Protocol Secure (HTTPS). Darüber hinaus war es sehr wichtig, die Datenintegrität zu gewährleisten. HTTPS steht für die Verwendung von HTTP mittels TLS. TLS ist besser bekannt unter seiner Vorgängerbezeichnung Secure Sockets Layer (SSL). Unser Auftraggeber perfect:net stellte uns ein entsprechendes Zertifikat zur Verfügung.

TLS besteht grundsätzlich aus zwei Ebenen, dem TLS Record Protocol und dem TLS Handshake Protocol. Mit dem TLS Handshake Protokoll tauschen Server und Client die Verschlüsselungsstandards und in weiterer Folge die kryptografischen Schlüssel aus, bevor die Datenübertragung beginnt.

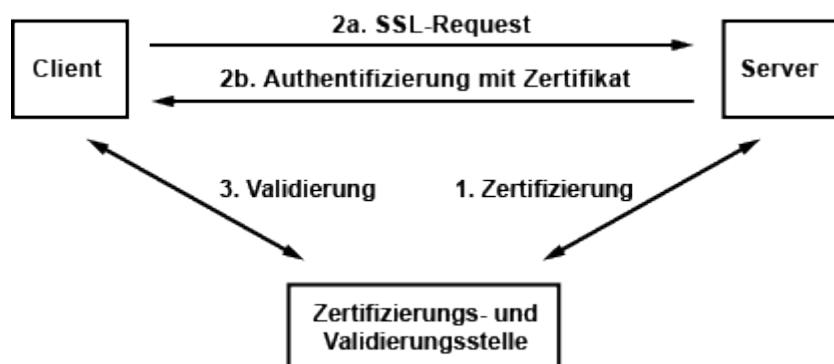


Abbildung 52 - TLS



13.2. Softwarearchitektur

Für Webapplikationen ist es üblich, nach dem Model-View-Presenter (MVP) Prinzip zu arbeiten. Davon ausgehend wurden die Applikation sowie die Entwicklungsschritte in drei Schichten gegliedert:

- User Interface (View)
- Business Logic (Presenter)
- PHP Core & Database (Model)

Um ein leistungsstarkes sowie einheitliches User Interface erstellen zu können, wurde auf das freie Framework Bootstrap (Version 3.3.7) und die freie JavaScript-Bibliothek jQuery Core (Version 3.1.1) gesetzt. Die im Bootstrap enthaltenen Gestaltungsvorlagen für Typografie, Formulare sowie weitere UI-Elemente vereinfachten das Erstellen des Designs. Die anfallenden DOM-Manipulationen, die die Grundlage eines dynamischen Inhalts bilden, konnten mit jQuery abgedeckt werden. Durch die Verwendung dieser beiden Komponenten wird zudem die Browser-Kompatibilität erhöht und die Einsatzmöglichkeit der Applikation aufgrund eines responsiven Designs erweitert.

Die Kommunikation zwischen dem User Interface und der dahinterliegenden Business Logic findet größtenteils unter Verwendung der AJAX Technologie statt. Daten werden in Form von JSON-Paketen übermittelt, und mithilfe einer dedizierten Dispatcher-Klasse werden asynchrone Anfragen entgegengenommen. Innerhalb der Business Logic sind zudem jene Methoden implementiert, die die Anbindung zu externen Services steuern bzw. bei weiteren Applikationsinstanzen unterschiedlich sein könnten. Die Verwendung eines neuen SMS-Gateways oder ein Wechsel zu einem neuen Webhosting Dienstleister kann einfach und modular durchgeführt werden, ohne die Grundstruktur zu beeinflussen.

Für die Basis der gesamten Applikation wurde ein pures PHP Konstrukt entwickelt. Unter Einhaltung der PHP Standards Recommendations (PSRs) wurden sämtliche Klassen in Namespaces gegliedert. Die für den Datenzugriff notwendigen ValueObjects und erweiternde Proxyklassen nutzen gemeinsame, strukturbildende Interfaces sowie Basisklassen, um Redundanzen zu vermeiden.



Durch den Einsatz von Factory- und Data Access Object Patterns wurde ein großes Maß an Kontinuität sowie die Möglichkeit, weitere Speichersysteme anzubinden, geschaffen.

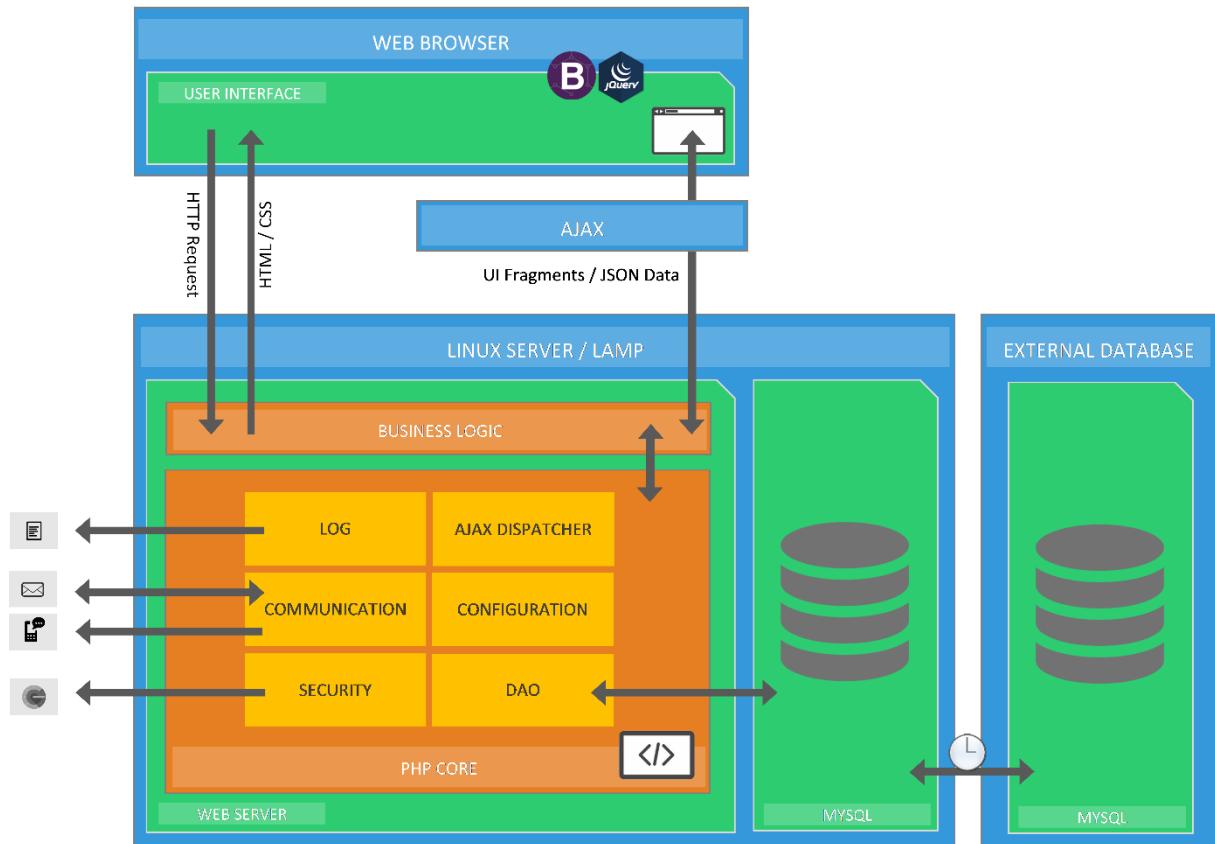


Abbildung 53 - Softwarearchitektur



13.3. Klassendiagramm

13.3.1. Namespace „Data“

13.3.1.1. Value Object

General

 Contact
  __construct()
  getId()
  setId(value)
  getFirstname()
  setFirstname(value)
  getLastname()
  setLastname(value)
  getTitleId()
  setTitleId(value)
  getTelephoneNumber()
  setTelephoneNumber(value)
  getMobile()
  setMobile(value)
  getFax()
  setFax(value)
  getMail()
  setMail(value)
  getDateOfBirth()
  setDateOfBirth(value)
  getVAT()
  setVAT(value)
  getCreateDate()
  setDateCreate(value)
  getDefaultAddressId()
  setDefaultAddressId(value)
  getContactId()
  setContactId(value)
  getContactTypeld()
  setContactTypeld(value)
  getNotes()
  setNotes(value)
  getActive()
  setActive(value)
  getCustomerNumber()
  setCustomerNumber(value)
  toArray()

Abbildung 54 - VO-Contact

 Invoice
  __construct()
  getId()
  setId(value)
  getContractId()
  setContractId(value)
  getInvoiceStatId()
  setInvoiceStatId(value)
  getInvoiceNumber()
  setInvoiceNumber(value)
  getInvoiceDate()
  setInvoiceDate(value)
  getInvoiceTotal()
  setInvoiceTotal(value)
  getPayment()
  setPayment(value)
  getCreateDate()
  setDateCreate(value)
  getDueDate()
  setDueDate(value)
  getDateOfPayment()
  setDateOfPayment(value)
  getInvoiceDeliveryTypId()
  setInvoiceDeliveryTypId(value)
  getDunningLevelId()
  setDunningLevelId(value)
  getPaymentMethodId()
  setPaymentMethodId(value)
  toArray()

Abbildung 55 - VO-Invoice

 Contract
  __construct()
  getId()
  setId(value)
  getCustomerId()
  setCustomerId(value)
  getPartnerId()
  setPartnerId(value)
  getVat()
  setVat(value)
  getStartDate()
  setStartDate(value)
  getEndDate()
  setEndDate(value)
  getContractStateId()
  setContractStateId(value)
  getPaymentMethodId()
  setPaymentMethodId(value)
  getOrderId()
  setOrderId(value)
  toArray()

Abbildung 56 - VO-Contract



C Account	C Address	C BulletinBoard
<code>m g __construct()</code>	<code>m g __construct()</code>	<code>m g __construct()</code>
<code>m g getId()</code>	<code>m g getId()</code>	<code>m g getId()</code>
<code>m g setId(value)</code>	<code>m g setId(value)</code>	<code>m g setId(value)</code>
<code>m g getContactId()</code>	<code>m g getContactId()</code>	<code>m g getTitle()</code>
<code>m g setContactId(value)</code>	<code>m g setContactId(value)</code>	<code>m g setTitle(value)</code>
<code>m g getUsername()</code>	<code>m g getAddress()</code>	<code>m g getMessage()</code>
<code>m g setUsername(value)</code>	<code>m g setAddress(value)</code>	<code>m g setMessage(value)</code>
<code>m g getPassword()</code>	<code>m g getPostalCode()</code>	<code>m g getStartDate()</code>
<code>m g setPassword(value)</code>	<code>m g setPostalCode(value)</code>	<code>m g setStartDate(value)</code>
<code>m g getEnableTwoFactorAuth()</code>	<code>m g getCity()</code>	<code>m g getEndDate()</code>
<code>m g setEnableTwoFactorAuth(value)</code>	<code>m g setCity(value)</code>	<code>m g setEndDate(value)</code>
<code>m g getSecurityIdentifier()</code>	<code>m g getCountryId()</code>	<code>m g toArray()</code>
<code>m g setSecurityIdentifier(value)</code>	<code>m g setCountryId(value)</code>	
<code>m g getTwoFactorSecretKey()</code>	<code>m g toArray()</code>	
<code>m g setTwoFactorSecretKey(value)</code>		
<code>m g toArray()</code>		

Abbildung 57 - VO-Account

Abbildung 58 - VO-Address

Abbildung 59 - VO-BulletinBoard

C Country
<code>m g __construct()</code>
<code>m g getId()</code>
<code>m g setId(value)</code>
<code>m g getName()</code>
<code>m g setName(value)</code>
<code>m g getAlpha2Code()</code>
<code>m g setAlpha2Code(value)</code>
<code>m g toArray()</code>

Abbildung 60 - VO-Country

C Currency
<code>m g __construct()</code>
<code>m g getId()</code>
<code>m g setId(value)</code>
<code>m g getName()</code>
<code>m g setName(value)</code>
<code>m g getCode()</code>
<code>m g setCode(value)</code>
<code>m g toArray()</code>

Abbildung 61 - VO-Currency

C Settings
<code>m g __construct()</code>
<code>m g getId()</code>
<code>m g setId(value)</code>
<code>m g getDefaultCurrencyId()</code>
<code>m g setDefaultCurrencyId(value)</code>
<code>m g getDefaultInvoiceDeliveryTypeid()</code>
<code>m g setDefaultInvoiceDeliveryTypeid</code>
<code>m g toArray()</code>

Abbildung 62 - VO-Settings

C RequestType
<code>m g __construct()</code>
<code>m g getId()</code>
<code>m g setId(value)</code>
<code>m g getName()</code>
<code>m g setName(value)</code>
<code>m g toArray()</code>

Abbildung 63 - VO-RequestType

C Faq
<code>m g __construct()</code>
<code>m g getId()</code>
<code>m g setId(value)</code>
<code>m g getCategoryid()</code>
<code>m g setCategoryId(value)</code>
<code>m g getQuestion()</code>
<code>m g setQuestion(value)</code>
<code>m g getAnswer()</code>
<code>m g setAnswer(value)</code>
<code>m g toArray()</code>

Abbildung 64 - VO-Faq

C TaskType
<code>m g __construct()</code>
<code>m g getId()</code>
<code>m g setId(value)</code>
<code>m g getName()</code>
<code>m g setName(value)</code>
<code>m g toArray()</code>

Abbildung 65 - VO-TaskType



C Log
<code>m __construct()</code>
<code>m getId()</code>
<code>m setId(value)</code>
<code>m getCreateDate()</code>
<code>m setDateCreate(value)</code>
<code>m getLogCategoryId()</code>
<code>m setLogCategoryId(value)</code>
<code>m getLogTypeid()</code>
<code>m setLogTypeid(value)</code>
<code>m getLogSeverityId()</code>
<code>m setLogSeverityId(value)</code>
<code>m getAffectedAccountId()</code>
<code>m setAffectedAccountId(value)</code>
<code>m getLogMessage()</code>
<code>m setLogMessage(value)</code>
<code>m toArray()</code>

Abbildung 66 - VO-Log

C Product
<code>m __construct()</code>
<code>m getId()</code>
<code>m setId(value)</code>
<code>m getName()</code>
<code>m setName(value)</code>
<code>m getDescription()</code>
<code>m setDescription(value)</code>
<code>m getProductTypeId()</code>
<code>m setProductTypeId(value)</code>
<code>m getCreateDate()</code>
<code>m setDateCreate(value)</code>
<code>m getEndDate()</code>
<code>m setDateEnd(value)</code>
<code>m getPrice()</code>
<code>m setPrice(value)</code>
<code>m toArray()</code>

Abbildung 67 - VO-Product

C Request
<code>m __construct()</code>
<code>m getId()</code>
<code>m setId(value)</code>
<code>m getRequestTypeId()</code>
<code>m setRequestTypeId(value)</code>
<code>m getAccountId()</code>
<code>m setAccountId(value)</code>
<code>m getDescription()</code>
<code>m setDescription(value)</code>
<code>m getExecutionStateId()</code>
<code>m setExecutionStateId(value)</code>
<code>m getCreateDate()</code>
<code>m setDateCreate(value)</code>
<code>m getEndDate()</code>
<code>m setDateEnd(value)</code>
<code>m toArray()</code>

Abbildung 68 - VO-Request

C TaskInquiry
<code>m __construct()</code>
<code>m getId()</code>
<code>m setId(value)</code>
<code>m getTaskId()</code>
<code>m setTaskId(value)</code>
<code>m getKey()</code>
<code>m setKey(value)</code>
<code>m getValue()</code>
<code>m setValue(value)</code>
<code>m toArray()</code>

Abbildung 69 - VO-TaskInquiry

C TaskParameter
<code>m __construct()</code>
<code>m getId()</code>
<code>m setId(value)</code>
<code>m getTaskId()</code>
<code>m setTaskId(value)</code>
<code>m getKey()</code>
<code>m setKey(value)</code>
<code>m getValue()</code>
<code>m setValue(value)</code>
<code>m toArray()</code>

Abbildung 70 - VO-TaskParameter

C Permission
<code>m __construct()</code>
<code>m getId()</code>
<code>m setId(value)</code>
<code>m getName()</code>
<code>m setName(value)</code>
<code>m getCategoryid()</code>
<code>m setCategoryId(value)</code>
<code>m toArray()</code>

Abbildung 71 - VO-Permission

C Order
<code>m __construct()</code>
<code>m getId()</code>
<code>m setId(value)</code>
<code>m getContactId()</code>
<code>m setContactId(value)</code>
<code>m getCreateDate()</code>
<code>m setDateCreate(value)</code>
<code>m getEndDate()</code>
<code>m setDateEnd(value)</code>
<code>m getOrderStateId()</code>
<code>m setOrderStateId(value)</code>
<code>m toArray()</code>

Abbildung 72 - VO-Order

C Task
<code>m __construct()</code>
<code>m getId()</code>
<code>m setId(value)</code>
<code>m getRequestId()</code>
<code>m setRequestId(value)</code>
<code>m getTaskTypeId()</code>
<code>m setTaskTypeId(value)</code>
<code>m getExecutionDate()</code>
<code>m setDateExecution(value)</code>
<code>m getExecutionStateId()</code>
<code>m setExecutionStateId(value)</code>
<code>m getParentTaskId()</code>
<code>m setParentTaskId(value)</code>
<code>m toArray()</code>

Abbildung 73 - VO-Task



Junction

C ProductDetail
m __construct()
m getId()
m setId(value)
m getProductDetailId()
m setProductDetailId(value)
m getProductName()
m setProductName(value)
m getDetailValue()
m setDetailValue(value)
m toArray()

Abbildung 74 - VO-ProductDetail

C GroupMember
m __construct()
m getGroupId()
m setGroupId(value)
m getAccountId()
m setAccountId(value)

Abbildung 75 - VO-GroupMember

C OrderedProduct
m __construct()
m getOrderid()
m setOrderId(value)
m getProductId()
m setProductId(value)

Abbildung 76 - VO-OrderedProduct

C BulletinBoardDelivery
m __construct()
m getBulletinBoardId()
m setBulletinBoardId(value)
m getDeliveryTypeId()
m setDeliveryTypeId(value)

Abbildung 77 - VO-BulletinBoardDelivery

C GroupPermission
m __construct()
m getGroupId()
m setGroupId(value)
m getAuthorizationRoleId()
m setAuthorizationRoleId(value)

Abbildung 78 - VO-GroupPermission

Metadata

C Metadata
m __construct()
m getId()
m setId(value)
m getName()
m setName(value)
m toArray()

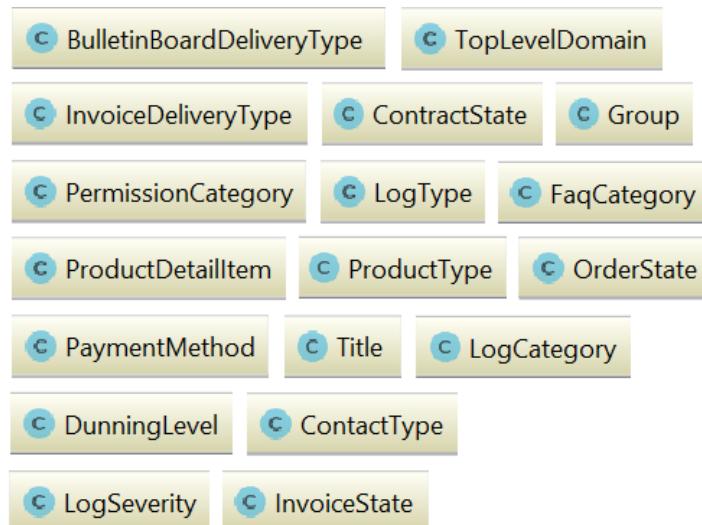


Abbildung 79 - VO-Metadata

13.4. Datenbankdesign

Aufgrund der Übersichtlichkeit wurde auf das ER-Modell verzichtet und direkt das Relationen-Modell eingefügt.

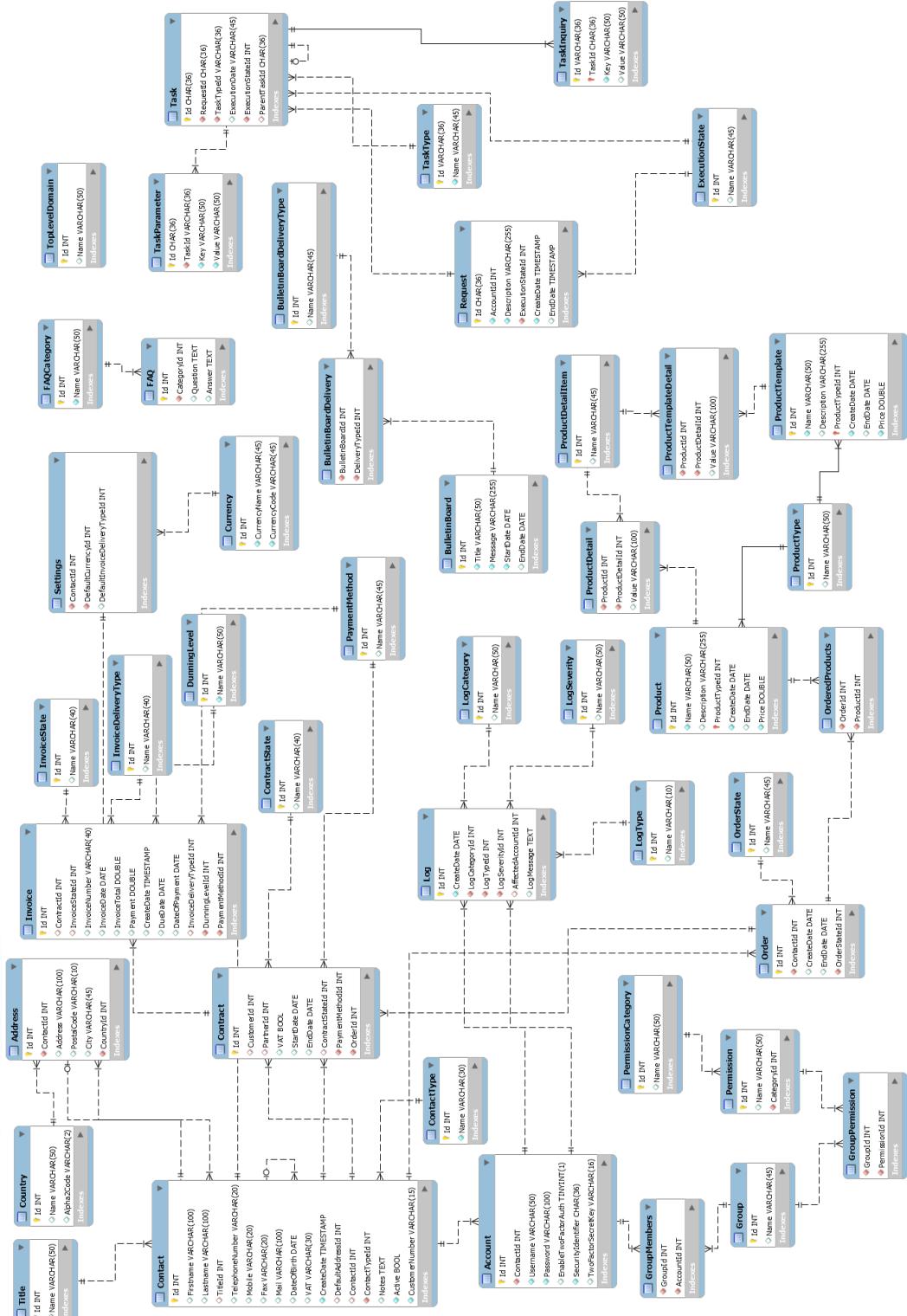


Abbildung 81 - Datenbankdesign



13.5. Beschreibung der Funktionalität

Die Funktionalität des PartnerDashboards umfasst fünf Kernmodule. Der erste und essentiellste Punkt ist die Authentifizierung, danach kommt das Kundenmodul, in dem Kunden angelegt und verwaltet werden können. Die folgenden Module sind das Bestellmodul und das Rechnungsmodul. Hier werden Bestellungen abgewickelt, die daraus resultierenden Rechnungen generiert und verprovisioniert. Abschließend kommt noch das Administrationsmodul, das es unserem Auftraggeber ermöglicht, diverse Einstellungen im Programm vorzunehmen.

13.5.1. Authentifizierung

Ein großes Augenmerk muss auf die Sicherheit gelegt werden, um einerseits die sensiblen Kundendaten schützen zu können, andererseits aber auch die firmeneigenen Daten optimal gesichert zu haben. Genau durch diesen Aspekt kamen wir zu dem Entschluss, eine Two-Factor Authentication für das Login im PartnerDashboard einzusetzen. Nach umfangreicher Recherche waren wir uns einig, dass wir den Google Authenticator verwenden wollen.

Wenn man auf die Website des PartnerDashboards gelangt, muss man sich mit seinem Benutzernamen und einem Passwort authentifizieren, das mit dem Standard Algorithmus von PHP (bcrypt) verschlüsselt ist. Sobald die eingegebenen Daten verifiziert sind, erscheint ein Dialogfenster, in dem man den von Google generierten Code eingibt. Dieser Code ist genau 30 Sekunden lang gültig, danach wird ein neuer generiert.

13.5.2. Kundenmodul

Eine der Kernfunktionen des PartnerDashboards ist die Kundenverwaltung. Hier hat der Partner einen Überblick über alle seine Kunden und die dazugehörigen Produkte. Sobald ein Partner einen neuen Kunden anlegt, wird dieser automatisch dem Partner zugeordnet. Der Partner hat in diesem Modul verschiedenste Möglichkeiten. Es können die Daten der Kunden geändert werden, Kundenadressen bearbeitet und neue hinzugefügt werden. Hier ist noch zu erwähnen, dass es beliebig viele Adressen pro Kunden geben kann, jedoch nur eine Standardadresse, die vom Programm kontrolliert wird und nicht falsch eingegeben werden kann.



13.5.3. Bestellmodul

Im Bestellmodul können zu einem ausgewählten Kunden Produkte hinzugefügt werden. Im ersten Schritt wird ein Kunde ausgewählt, danach wird dem Kunden ein vorgefertigtes Produkt zugewiesen. Nach dem Zuweisen wird das Produkt konfiguriert und im letzten Schritt kostenpflichtig bestellt. Um eine Domain zu bestellen, muss zuerst der im PartnerDashboard integrierte Verfügbarkeitscheck gemacht werden. Dabei wird geprüft, ob die gewünschte Domain noch frei ist. Dieser Schritt muss auch beim Erwerb eines Web Hosting Paketes gemacht werden. Bei den Paketen müssen nicht alle fünf Domains sofort zugewiesen werden, dies kann nach und nach passieren. Alle anderen Konfigurationen des Pakets müssen jedoch bei der Bestellung abgeschlossen sein. Alle bestellten Produkte werden zuerst in einen Warenkorb gespeichert und können von dort in eine tatsächliche Bestellung übergehen. Der Warenkorb wird, sobald er erstellt wird, in der Datenbank gespeichert, um Bestellungen nicht sofort abschließen zu müssen. Er bleibt so lange gespeichert, bis entweder die Bestellung abgeschlossen oder der Warenkorb wieder geleert wird.

13.5.4. Rechnungsmodul

Das Rechnungsmodul ist ein komplett automatisierter Vorgang, der einmal pro Monat die Rechnungen erstellt und an die jeweiligen Kunden sendet. Weiters wird hier die Provisionsabrechnung für die Partner erstellt. Für die Provisionsabrechnung werden alle bezahlten Rechnungen von den Kunden des Partners herangezogen, und für die verschiedenen Pakete unterschiedliche prozentuale Anteile an die Partner ausgeschüttet.

13.5.5. Administrationsmodul

Für unseren Auftraggeber war es wichtig, dass er sowohl Partner als auch Kunden über das Programm verwalten kann. Hierfür wurde eine Benutzerverwaltung implementiert, in der Benutzer angelegt und administriert werden können. Zusätzlich wurde im Administrationsmodul der Zugriff über ein rollenbasiertes Konzept erstellt. Hier wird es dem Administrator ermöglicht, verschiedenste Gruppen zu erstellen und ihnen Berechtigungen und Mitglieder hinzuzufügen oder zu entziehen. Somit wird genauestens geregelt, welche Gruppe welche Module sehen und welche Aktionen ausführen darf.



PartnerDashboard

Qualitätsmanagement





14. Qualitätsmanagement

Beginnend in der Planungsphase wurden Qualitätsrichtlinien und -ziele definiert, die es während der Entwicklung bis hin zur Fertigstellung zu erreichen galt. Um die zur Verfügung stehende Zeit optimal zu nutzen, wurden große Teile der Qualitätskontrolle automatisiert und Oberflächentests am Ende der Sprints manuell durchgeführt. Erkannte Fehler wurden aufgezeigt und protokolliert.

14.1. Programmierrichtlinien

Zusätzlich zur Überprüfung der Ergebnisse ist es unerlässlich, dass bereits in der Entwicklungsphase klare Qualitätsnormen vereinbart werden. In unserem Fall haben die Recherchen ergeben, dass die PHP Standards Recommendations (PSRs; php-fig.org) als ideales Kompendium dienen.

14.1.1. Eingesetzte PSRs

- PSR-1 – Basic Coding Standard
- PSR-2 – Coding Style Guide
- PSR-3 – Logger Interface
- PSR-4 – Autoloading Standard

Parallel zur formalen Gestaltung des Quellcodes, um die Lesbarkeit zu erhöhen, orientierten wir uns an erprobten Entwurfsmustern/Design Patterns, um die Testbarkeit sowie Wartbarkeit zu steigern.



14.1.2. Verwendete Design Patterns

- Value Object Pattern
- Data Access Object Pattern
- Factory Pattern
- Singleton Pattern
- Proxy Pattern

Darüber hinaus, wurde nach dem Model-View-Presenter (MVP) Prinzip gearbeitet. Die strikte Trennung der einzelnen Komponenten verbessert die Testbarkeit zusätzlich.

14.2. Testdurchführung

14.2.1. Komponententest

Der Komponententest soll sicherstellen, dass einzelne Elemente der Applikation ihre funktionellen Anforderungen erfüllen. Der Schwerpunkt liegt auf den einzelnen Klassen, den daraus resultierenden Objektinstanzen sowie deren Relation zueinander. Diese Form der Tests wird automatisiert abgebildet.

14.2.2. Integrationstest

Komplette Abläufe über mehrere Komponenten werden abgebildet und kontrolliert. Die Testfälle beschreiben dokumentierte UseCases sowie entdeckte Fehler während der Entwicklungsphase. Das Testsystem simuliert hierbei die einzelnen Schritte, sodass kein User Interface benötigt wird.

14.2.3. Systemtest

Nach erfolgreichem Komponenten- und Integrationstest werden UseCases manuell durchlaufen und die Ergebnisse protokolliert. Aufgrund des Zeitaufwandes war diese Form des Testens erst nach den Entwicklungsarbeiten angedacht.



14.3. Teststrategie

Aus Qualitätssicht kann ein Sprint stets als PDCA-Zyklus gesehen werden, da wiederholt die vier Phasen durchlaufen werden. Eventuelle Probleme werden in den nächsten Sprint übernommen.

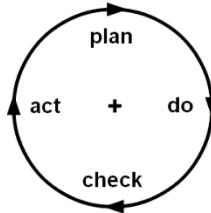


Abbildung 81 - Qualitätsmanagement-PDCA-Zyklus

Während der Entwicklungsphase kann für laufende Tests die zur Verfügung stehende Datenbank verwendet werden. Abschließende Systemtests werden erst als vollwertig betrachtet, wenn diese mit einer Datenbank im Ursprungszustand durchgeführt wurden.

14.4. Komponententests

QUALITYASSURANCE » VALUEOBJECT: ACCOUNT	
QA_newAccount_getId	'getId()' must deliver null-Value after initialization
QA_newAccount_getContactId	'getContactId()' must deliver null-Value after initialization
QA_newAccount_getUsername	'getUsername()' must deliver null-Value after initialization
QA_newAccount_getPassword	'getPassword()' must deliver null-Value after initialization
QA_newAccount_getEnableTwoFactorAuth	'getEnableTwoFactorAuth()' must deliver 'bool(false)' after initialization
QA_newAccount_setId	'setId(10)'; must deliver 'int(10)' with getter
QA_newAccount_setContactId	'setContactId(20)'; must deliver 'int(20)' with getter
QA_newAccount_setUsername	'setUsername("testuser")'; must deliver 'testuser' with getter
QA_newAccount_setPassword	'setPassword("testpass")'; must deliver 'testpass' with getter
QA_newAccount_setEnableTwoFactorAuth	'setEnableTwoFactorAuth(true)'; must deliver 'bool(true)' with getter

Abbildung 82 - Qualitätsmanagement-VO-Account

QUALITYASSURANCE » VALUEOBJECT: ADDRESS	
QA_newAddress_getId	'getId()' must deliver null-Value after initialization
QA_newAddress_getContactId	'getId()' must deliver null-Value after initialization
QA_newAddress_getAddress	'getId()' must deliver null-Value after initialization
QA_newAddress_getPostalCode	'getId()' must deliver null-Value after initialization
QA_newAddress_getCity	'getId()' must deliver null-Value after initialization
QA_newAddress_getCountryId	'getId()' must deliver null-Value after initialization
QA_newAddress_setId	'setId(10)'; must deliver 'int(10)' with getter
QA_newAddress_setContactId	'setContactId(11)'; must deliver 'int(11)' with getter
QA_newAddress_setAddress	'setAddress("Testaddress")'; must deliver 'Testaddress' with getter
QA_newAddress_setPostalCode	'setPostalCode(8200)'; must deliver 'int(8200)' with getter
QA_newAddress_setCity	'setCity("Gleisdorf")'; must deliver 'Gleisdorf' with getter
QA_newAddress_setCountryId	'setCountryId(12)'; must deliver 'int(12)' with getter

Abbildung 83 - Qualitätsmanagement-VO-Address



QUALITYASSURANCE » VALUEOBJECT: BULLETINBOARD	
QA_newBulletinBoard_getId	'getId()' must deliver null-Value after initialization
QA_newBulletinBoard_getTitle	'getTitle()' must deliver null-Value after initialization
QA_newBulletinBoard_getMessage	'getMessage()' must deliver null-Value after initialization
QA_newBulletinBoard_getStartDate	'getStartDate()' must deliver null-Value after initialization
QA_newBulletinBoard_getEndDate	'getEndDate()' must deliver null-Value after initialization
QA_newBulletinBoard_setId	'setId(5)'; must deliver 'int(5)' with getter
QA_newBulletinBoard_setTitle	'setTitle("TestTitle")'; must deliver 'TestTitle' with getter
QA_newBulletinBoard_setMessage	'setMessage("TestMessage")'; must deliver 'TestMessage' with getter
QA_newBulletinBoard_setStartDate	'setStartDate('2016-12-10')'; must deliver '2016-12-10' with getter
QA_newBulletinBoard_setEndDate	'setEndDate('2016-12-10')'; must deliver '2016-12-10' with getter

Abbildung 84 - Qualitätsmanagement-VO-BulletinBoard

QUALITYASSURANCE » VALUEOBJECT: CONTACT	
QA_newContact_getId	'getId()' must deliver null-Value after initialization
QA_newContact_getFirstname	'getFirstname()' must deliver null-Value after initialization
QA_newContact_getLastname	'getLastname()' must deliver null-Value after initialization
QA_newContact_getTitleId	'getTitleId()' must deliver null-Value after initialization
QA_newContact_getTelephoneNumber	'getTelephoneNumber()' must deliver null-Value after initialization
QA_newContact_getMobile	'getMobile()' must deliver null-Value after initialization
QA_newContact_getFax	'getFax()' must deliver null-Value after initialization
QA_newContact_getMail	'getMail()' must deliver null-Value after initialization
QA_newContact_getDateOfBirth	'getDateOfBirth()' must deliver null-Value after initialization
QA_newContact_getVAT	'getVAT()'; must deliver bool 'false' after initialization
QA_newContact_getCreateDate	'getCreateDate()' must deliver null-Value after initialization
QA_newContact_getDefaultAddressId	'getDefaultAddressId()' must deliver null-Value after initialization
QA_newContact_setId	'setId(10)'; must deliver 'int(10)' with getter
QA_newContact_setFirstname	'setFirstname("TestVorname")'; must deliver 'TestVorname' with getter
QA_newContact_setLastname	'setLastname("TestNachname")'; must deliver 'TestNachname' with getter
QA_newContact_setTitleId	'setTitleId(10)'; must deliver 'int(10)' with getter
QA_newContact_setTelephoneNumber	'setTelephoneNumber("+4331122646")'; must deliver '+4331122646' with getter
QA_newContact_setMobile	'setMobile("+436767515432")'; must deliver '+436767515432' with getter
QA_newContact_setFax	'setFax("+43311226461")'; must deliver '+43311226461' with getter
QA_newContact_setMail	'setMail("test.mail@inode.at")'; must deliver 'test.mail@inode.at' with getter
QA_newContact_setDateOfBirth	' setDateOfBirth('2016-12-10')'; must deliver '2016-12-10' with getter
QA_newContact_setVAT	'setVAT(true)'; must deliver bool 'false' with getter
QA_newContact_setCreateDate	'setCreateDate('2016-12-10')'; must deliver '2016-12-10' with getter
QA_newContact_setDefaultAddressId	'setDefaultAddressId(11)'; must deliver 'int11' with getter
QA_newContact_setContactId	'setContactId(12)'; must deliver 'int(10)' with getter

Abbildung 85 - Qualitätsmanagement-VO-Contact



QUALITYASSURANCE » VALUEOBJECT: CONTRACT	
QA_newContract_getId	'getId()' must deliver null-Value after initialization
QA_newContract_getCustomerId	'getCustomerId()' must deliver null-Value after initialization
QA_newContract_getPartnerId	'getPartnerId()' must deliver null-Value after initialization
QA_newContract_getVat	'getVat()' must deliver null-Value after initialization
QA_newContract_getStartDate	'getStartDate()' must deliver null-Value after initialization
QA_newContract_getEndDate	'getEndDate()' must deliver null-Value after initialization
QA_newContract_getContractStateId	'getContractStateId()' must deliver null-Value after initialization
QA_newContract_getPaymentMethodId	'getPaymentMethodId()' must deliver null-Value after initialization
QA_newContract_getOrderId	'getOrderId()' must deliver null-Value after initialization
QA_newContract_setId	'setId(5)'; must deliver 'int(5)' with getter
QA_newContract_setCustomerId	'setCustomerId(10)'; must deliver 'int(10)' with getter
QA_newContract_setPartnerId	'setPartnerId(3)'; must deliver 'int(3)' with getter
QA_newContract_setVat	'setVat(1)'; must deliver 'int(1)' with getter
QA_newContract_setStartDate	'setCreateDate('2017-01-09')'; must deliver '2017-01-09' with getter
QA_newContract_setEndDate	'setCreateDate('2017-01-09')'; must deliver '2017-01-09' with getter
QA_newContract_setContractStateId	'setContractStateId(2)'; must deliver 'int(2)' with getter
QA_newContract_setPaymentMethodId	'setPaymentMethodId(1)'; must deliver 'int(1)' with getter
QA_newContract_setOrderId	'setOrderId(100)'; must deliver 'int(100)' with getter

Abbildung 86 - Qualitätsmanagement-VO-Contract

QUALITYASSURANCE » VALUEOBJECT: CURRENCY	
QA_newCurrency_getId	'getId()' must deliver null-Value after initialization
QA_newCurrency_getName	'getName()' must deliver null-Value after initialization
QA_newCurrency_getCode	'getCode()' must deliver null-Value after initialization
QA_newContract_setId	'setId(5)'; must deliver 'int(5)' with getter
QA_newCurrency_setName	'setName("TestUser")'; must deliver 'TestUser' with getter
QA_newCurrency_setCode	'setCode(8200)'; must deliver 'int(8200)' with getter

Abbildung 87 - Qualitätsmanagement-VO-Currency

QUALITYASSURANCE » VALUEOBJECT: FAQ	
QA_newFaq_getId	'getId()' must deliver null-Value after initialization
QA_newFaq_getCategoryId	'getCategoryId()' must deliver null-Value after initialization
QA_newFaq_getQuestion	'getQuestion()' must deliver null-Value after initialization
QA_newFaq_getAnswer	'getAnswer()' must deliver null-Value after initialization
QA_newFaq_setId	'setId(100)'; must deliver 'int(100)' with getter
QA_newFaq_setCategoryId	'setCategoryId(99)'; must deliver 'int(99)' with getter
QA_newFaq_setQuestion	'setQuestion(3)'; must deliver 'int(3)' with getter
QA_newFaq_setAnswer	'setAnswer(4)'; must deliver 'int(4)' with getter

Abbildung 88 - Qualitätsmanagement-VO-FAQ



QUALITYASSURANCE » VALUEOBJECT: INVOICE	
QA_newInvoice_getId	'getId()' must deliver null-Value after initialization
QA_newInvoice_getContractId	'getContractId()' must deliver null-Value after initialization
QA_newInvoice_getInvoiceStateId	'getInvoiceStateId()' must deliver null-Value after initialization
QA_newInvoice_getInvoiceNumber	'getInvoiceNumber()' must deliver null-Value after initialization
QA_newInvoice_getInvoiceDate	'getInvoiceDate()' must deliver null-Value after initialization
QA_newInvoice_getInvoiceTotal	'getInvoiceTotal()' must deliver null-Value after initialization
QA_newInvoice_getPayment	'getPayment()' must deliver null-Value after initialization
QA_newInvoice_getCreateDate	'getCreateDate()' must deliver null-Value after initialization
QA_newInvoice_getDueDate	'getDueDate()' must deliver null-Value after initialization
QA_newInvoice_getDateOfPayment	'getDateOfPayment()' must deliver null-Value after initialization
QA_newInvoice_getInvoiceDeliveryTypeId	'getInvoiceDeliveryTypeId()' must deliver null-Value after initialization
QA_newInvoice_getDunningLevelId	'getDunningLevelId()' must deliver null-Value after initialization
QA_newInvoice_getPaymentMethodId	'getPaymentMethodId()' must deliver null-Value after initialization
QA_newInvoice_setId	'setId(25)'; must deliver 'int(25)' with getter
QA_newInvoice_setContractId	'setContractId(50)'; must deliver 'int(50)' with getter
QA_newInvoice_setInvoiceStateId	'setInvoiceStateId(3)'; must deliver 'int(3)' with getter
QA_newInvoice_setInvoiceNumber	'setInvoiceNumber(33)'; must deliver 'int(33)' with getter
QA_newInvoice_setInvoiceDate	'setInvoiceDate('2017-01-09')'; must deliver '2017-01-09' with getter
QA_newInvoice_setInvoiceTotal	'setInvoiceTotal(1532)'; must deliver 'int(1532)' with getter
QA_newInvoice_setPayment	'setPayment(1)'; must deliver 'int(1)' with getter
QA_newInvoice_setCreateDate	'setCreateDate('2017-01-01')'; must deliver '2017-01-01' with getter
QA_newInvoice_setDueDate	'setDueDate('2018-01-09')'; must deliver '2018-01-09' with getter
QA_newInvoice_setDateOfPayment	'setDueDate('2018-11-11')'; must deliver '2018-11-11' with getter
QA_newInvoice_setInvoiceDeliveryTypeId	'setInvoiceDeliveryTypeId(4)'; must deliver 'int(4)' with getter
QA_newInvoice_setDunningLevelId	'setDunningLevelId(2)'; must deliver 'int(2)' with getter
QA_newInvoice_setPaymentMethodId	'setPaymentMethodId(1)'; must deliver 'int(1)' with getter

Abbildung 89 - Qualitätsmanagement-VO-Invoice

QUALITYASSURANCE » VALUEOBJECT: LOG	
QA_newLog_getId	'getId()' must deliver null-Value after initialization
QA_newLog_getCreateDate	'getCreateDate()' must deliver null-Value after initialization
QA_newLog_getLogCategoryId	'getLogCategoryId()' must deliver null-Value after initialization
QA_newLog_getLogTypeId	'getLogTypeId()' must deliver null-Value after initialization
QA_newLog_getLogSeverityId	'getLogSeverityId()' must deliver null-Value after initialization
QA_newLog_getAffectedAccountId	'getAffectedAccountId()' must deliver null-Value after initialization
QA_newLog_getLogMessage	'getLogMessage()' must deliver null-Value after initialization
QA_newLog_setId	'setId(22)'; must deliver 'int(22)' with getter
QA_newLog_setCreateDate	'setCreateDate('2017-01-09')'; must deliver '2017-01-09' with getter
QA_newLog_setLogCategoryId	'setLogCategoryId(3)'; must deliver 'int(3)' with getter
QA_newLog_setLogTypeId	'setLogTypeId(5)'; must deliver 'int(5)' with getter
QA_newLog_setLogSeverityId	'setLogSeverityId(7)'; must deliver 'int(7)' with getter
QA_newLog_setAffectedAccountId	'setAffectedAccountId(11)'; must deliver 'int(11)' with getter
QA_newLog_setLogMessage	'setLogMessage("Daten konnten nicht gespeichert werden!")'; must deliver 'Daten konnten nicht gespeichert werden!' with getter

Abbildung 90 - Qualitätsmanagement-VO-Log



QUALITYASSURANCE » VALUEOBJECT: ORDER	
QA_newOrder_getId	'getId()' must deliver null-Value after initialization
QA_newOrder_getContactId	'getContactId()' must deliver null-Value after initialization
QA_newOrder_getCreateDate	'getCreateDate()' must deliver null-Value after initialization
QA_newOrder_getEndDate	'getEndDate()' must deliver null-Value after initialization
QA_newOrder_getOrderStateId	'getOrderStateId()' must deliver null-Value after initialization
QA_newOrder_setId	'setId(9)'; must deliver 'int(9)' with getter
QA_newOrder_setContactId	'setContactId(101)'; must deliver 'int(101)' with getter
QA_newOrder_setCreateDate	'setCreateDate(''2017-01-09'')'; must deliver '2017-01-09' with getter
QA_newOrder_setEndDate	'setEndDate(''2018-01-09'')'; must deliver '2018-01-09' with getter
QA_newOrder_setOrderStateId	'setOrderStateId(2)'; must deliver 'int(2)' with getter

Abbildung 91 - Qualitätsmanagement-VO-Order

QUALITYASSURANCE » VALUEOBJECT: PRODUCT	
QA_newProduct_getId	'getId()' must deliver null-Value after initialization
QA_newProduct_getName	'getName()' must deliver null-Value after initialization
QA_newProduct_getDescription	'getDescription()' must deliver null-Value after initialization
QA_newProduct_getCreateDate	'getCreateDate()' must deliver null-Value after initialization
QA_newProduct_getEndDate	'getEndDate()' must deliver null-Value after initialization
QA_newProduct_getPrice	'getPrice()' must deliver null-Value after initialization
QA_newProduct_setId	'setId(34)'; must deliver 'int(34)' with getter
QA_newProduct_setName	'setName("Webhost Large")'; must deliver 'Webhost Large' with getter
QA_newProduct_setDescription	'setDescription("Webspace 20GB + 20 Mailadressen")'; must deliver 'Webspace 20GB + 20 Mailadressen' with getter
QA_newProduct_setCreateDate	'setCreateDate(''2017-01-09'')'; must deliver '2017-01-09' with getter
QA_newProduct_setEndDate	'setEndDate(''2018-01-09'')'; must deliver '2018-01-09' with getter
QA_newProduct_setPrice	'setPrice(72)'; must deliver 'int(72)' with getter

Abbildung 92 - Qualitätsmanagement-VO-Product

QUALITYASSURANCE » VALUEOBJECT: PRODUCTDETAIL	
QA_newProductDetail_getId	'getId()' must deliver null-Value after initialization
QA_newProductDetail_getProductDetailId	'getProductDetailId()' must deliver null-Value after initialization
QA_newProductDetail_getProductDetailName	'getProductDetailName()' must deliver null-Value after initialization
QA_newProductDetail_getProductDetailValue	'getProductDetailValue()' must deliver null-Value after initialization
QA_newProductDetail_setId	'setId(34)'; must deliver 'int(34)' with getter
QA_newProductDetail_setProductDetailId	'setProductDetailId(34)'; must deliver 'int(34)' with getter
QA_newProductDetail_setProductDetailName	'setProductDetailName("TestDetail")'; must deliver 'TestDetail' with getter
QA_newProductDetail_setProductDetailValue	'setProductDetailValue(33)'; must deliver 'int(33)' with getter

Abbildung 93 - Qualitätsmanagement-VO-Productdetail



14.5. Integrationstests

QUALITYASSURANCE » INTEGRATION TEST 001: MANAGE CUSTOMER	
QA_01_CreateEmptyCustomerObject	Created object must contain NULL values only
QA_02_SetCustomerTestData	Set proper values for customer test object
QA_03_CreateCustomerInDatabase	Save customer data to database
QA_04_DisposeCustomerObject	Dispose customer object
QA_05_LoadCustomerFromDatabase	Load customer data from database
QA_06_ValidateCustomerData	Validate loaded data with expected result
QA_07_ModifyCustomerTestData	Modify customer object
QA_08_UpdateCustomerInDatabase	Update customer data in database
QA_09_DisposeCustomerObject	Dispose customer object
QA_10_LoadCustomerFromDatabase	Load customer data from database
QA_11_ValidateCustomerData	Validate loaded data with expected result
QA_12_DeleteCustomerFromDatabase	Delete temporary customer from database

Abbildung 94 - Qualitätsmanagement-Integrationstest-Manage Customer

QUALITYASSURANCE » INTEGRATION TEST 002: MANAGE ACCOUNT	
QA_01_CreateEmptyCustomerObject	Created object must contain NULL values only
QA_02_SetCustomerTestData	Set proper values for customer test object
QA_03_CreateCustomerInDatabase	Save customer data to database
QA_04_CreateEmptyAccountObject	Created object must contain NULL values only
QA_05_SetAccountTestData	Set proper values for account test object
QA_06_CreateAccountInDatabase	Save customer data to database
QA_07_DisposeAccountObject	Dispose account object
QA_08_LoadAccountFromDatabase	Load account data from database
QA_09_ValidateAccountData	Validate loaded data with expected result
QA_10_ModifyAccountTestData	Modify account data
QA_11_UpdateAccountInDatabase	Update account data in database
QA_12_DisposeAccountObject	Dispose account object
QA_13_LoadAccountFromDatabase	Load account data from database
QA_14_ValidateAccountData	Validate loaded data with expected result
QA_15_DeleteAccountFromDatabase	Delete temporary account from database
QA_16_DeleteCustomerFromDatabase	Delete temporary customer from database

Abbildung 95 - Qualitätsmanagement-Integrationstest-Manage Account

QUALITYASSURANCE » INTEGRATION TEST 003: MANAGE FAQ	
QA_01_CreateEmptyFaqObject	Created object must contain NULL values only
QA_02_SetFaqTestData	Set proper values for faq test object
QA_03_CreateFaqInDatabase	Save faq data to database
QA_04_DisposeFaqObject	Dispose faq object
QA_05_LoadFaqObjectFromDatabase	Load faq data from database
QA_06_ValidateFaqData	Validate loaded data with expected result
QA_07_ModifyFaqTestData	Modify faq data
QA_08_UpdateFaqInDatabase	Update faq data in database
QA_09_DisposeFaqObject	Dispose faq object
QA_10_LoadFaqFromDatabase	Load faq data from database
QA_11_ValidateFaqData	Validate loaded data with expected result
QA_12_DeleteFaqFromDatabase	Delete temporary faq from database

Abbildung 96 - Qualitätsmanagement-Integrationstest-Manage FAQ



QUALITYASSURANCE » INTEGRATION TEST 004: MANAGE BULLETINBOARD	
QA_01_CreateEmptyBulletinBoardObject	Created object must contain NULL values only
QA_02_SetBulletinBoardTestData	Set proper values for bulletinboard test object
QA_03_CreateBulletinBoardInDatabase	Save bulletinboard data to database
QA_04_DisposeBulletinBoardObject	Dispose bulletinboard object
QA_05_LoadBulletinBoardFromDatabase	Load bulletinboard data from database
QA_06_ValidateBulletinBoardData	Validate loaded data with expected result
QA_07_ModifyBulletinBoardTestData	Modify bulletinboard data
QA_08_UpdateBulletinBoardInDatabase	Update bulletinboard data in database
QA_09_DisposeBulletinBoardObject	Dispose bulletinboard object
QA_10_LoadBulletinBoardFromDatabase	Load bulletinboard data from database
QA_11_ValidateBulletinBoardData	Validate loaded data with expected result
QA_12_DeleteBulletinBoardFromDatabase	Delete temporary bulletinboard from database

Abbildung 97 - Qualitätsmanagement-Integrationstest-Manage BulletinBoard

QUALITYASSURANCE » INTEGRATION TEST 005: ORDER PRODUCT	
QA_01_CreateEmptyProductObject	Created object must contain NULL values only
QA_02_CreateEmptyOrderObject	Created object must contain NULL values only
QA_03_CreateDummyCustomerInDatabase	Insert temporary customer into database
QA_04_LoadDummyProductTemplate	Load dummy product template
QA_05_SetTemplateDataOnProductObject	Use loaded template as ordered product
QA_06_CreateProductInDatabase	Save product to database
QA_07_SetOrderTestData	Set test order data (OrderState::CART)
QA_08_CreateOrderInDatabase	Save order to database
QA_09_DisposeObjects	Dispose temporary objects
QA_10_LoadOrderFromDatabase	Load order data from database
QA_11_ModifyOrderTestData	Modify order data (OrderState::SUBMITTED)
QA_12_UpdateOrderInDatabase	Update order data in database
QA_13_DisposeObjects	Dispose temporary objects
QA_14_LoadOrderFromDatabase	Load order data from database
QA_15_ValidateOrderData	Validate loaded data with expected result
QA_16_DeleteOrderFromDatabase	Delete temporary order from database
QA_17_DeleteProductFromDatabase	Delete temporary product from database
QA_18_DeleteCustomerFromDatabase	Delete temporary customer from database

Abbildung 98 - Qualitätsmanagement-Integrationstest-Order Product

QUALITYASSURANCE » INTEGRATION TEST 006: ACCESS MANAGEMENT	
QA_01_CreateDummyAccountInDatabase	Insert temporary account into database
QA_02_LoadSecurityToken	Load security token
QA_03_ValidateSecurityToken	Validate loaded data with expected result (no access)
QA_04_DisposeObject	Dispose account object
QA_05_AddDummyAccountToAdministratorGroup	Add dummy account to administrators
QA_06_LoadSecurityToken	Load security token
QA_07_ValidateSecurityToken	Validate loaded data with expected result (full access)
QA_08_DeleteDummyAccountFromDatabase	Delete temporary account from database

Abbildung 99 - Qualitätsmanagement-Integrationstest-Access Management



14.6. Systemtests

Die hier abgebildeten Tests sind der letzte Stand des Testprozesses. Auf das Darstellen des Testverlaufs sowie der dazugehörigen Fehlerbehebung wurde bewusst verzichtet.

14.6.1. Userstory 1

Als Auftraggeber möchte ich meinen Partnern schnelle Informationen über Systemausfälle bzw. Wartungsarbeiten liefern können.

14.6.1.1. Logischer Testfall

Neue Nachricht im BulletinBoard (News) erstellen.

14.6.1.2. Konkreter Testfall

- Als Administrator anmelden
- Im Dashboard zur Administration wechseln
- BulletinBoard (News) öffnen
- Button "Eintrag erstellen" klicken
- Neuen Eintrag erstellen
- Button "Speichern" klicken
- Überprüfen, ob die neue Nachricht verfügbar ist.

14.6.1.3. Testergebnis

✓ / OK

14.6.2. Userstory 2

Als Auftraggeber möchte ich eine Gesamtübersicht der Partner und deren Kunden.

14.6.2.1. Logischer Testfall

Partnerübersicht und verbundene Kunden öffnen.

14.6.2.2. Konkreter Testfall

- Als Administrator anmelden
- Im Dashboard zur Administration wechseln
- Benutzerverwaltung öffnen



- Type auf "Partner" ändern, um ausschließlich Partner anzuzeigen
- Auf Steuerungssymbol klicken und "Kundenübersicht" wählen

14.6.2.3. Testergebnis

✓ / OK

14.6.3. Userstory 3

Als Auftraggeber möchte ich Einsicht über Anmeldungen und diverse Änderungen im Dashboard (» LOG-Funktion).

14.6.3.1. Logischer Testfall

Anmeldung und Benutzeränderung durchführen.

14.6.3.2. Konkreter Testfall

- Log-Datei öffnen
- Prüfen, ob Einträge für Anmeldungen korrekt vorhanden sind
- Prüfen, ob Benutzeränderung protokolliert ist

14.6.3.3. Testergebnis

✓ / OK

14.6.4. Userstory 4

Als Auftraggeber möchte ich ein hohes Sicherheitsniveau bei der Benutzerauthentifizierung.

14.6.4.1. Logischer Testfall

Anmeldung mit Two-Factor Authentication aktivieren und durchführen.

14.6.4.2. Konkreter Testfall

- Als Administrator anmelden
- Im Dashboard zur Administration wechseln
- Benutzerverwaltung öffnen
- Neuen Benutzer erstellen
- Mit neuem Benutzerkonto anmelden



- Im Benutzerprofil Two-Factor Authentication aktivieren
- Erneute Anmeldung mit Benutzername, Passwort und Security Code.

14.6.4.3. Testergebnis

✓ / OK

14.6.5. Userstory 5

Als Auftraggeber/Partner möchte ich, dass das Dashboard auf verschiedenen Geräten wie PC und Tablet erreichbar ist.

14.6.5.1. Logischer Testfall

Öffnen des PartnerDashboards auf verschiedenen Geräten.

14.6.5.2. Konkreter Testfall

- Verwenden eines Windows Computers (22“ / 13“)
- Verwenden von Tablets (Microsoft Surface, Apple iPad 4)

14.6.5.3. Testergebnis

✓ / OK

14.6.6. Userstory 6

Als Auftraggeber/Partner möchte ich die Möglichkeit zur selbstständigen Neuanlage und Stammdatenpflege der Kundendaten.

14.6.6.1. Logischer Testfall

Neuen Benutzer anlegen und anschließende Änderung durchführen.

14.6.6.2. Konkreter Testfall

- Als Benutzer anmelden
- Im Dashboard im Bereich „Allgemein“ den Punkt Kundenübersicht wählen
- Datensatz des gewünschten Kunden » Klick auf "Information-Button"
- Änderungsbutton auf Kundeninformationsseite klicken
- Informationen im „Kunden bearbeiten“-Dialog anpassen und speichern



- Überprüfen, ob Daten korrekt gespeichert sind

14.6.6.3. Testergebnis

✓ / OK

14.6.7. Userstory 7

Als Partner möchte ich eine Bestellmöglichkeit direkt über das Dashboard.

14.6.7.1. Logischer Testfall

Neue Produktbestellung für Kunden durchführen.

14.6.7.2. Konkreter Testfall

- Als Benutzer anmelden
- Im Dashboard in „Allgemein“ den Punkt „Produkte“ wählen
- Betroffenen Kunden auswählen
- Gewünschte Produkte in den Warenkorb legen
- Auf Warenkorb-Symbol in Menüleiste klicken
- Bestellung abschließen
- Überprüfen, ob dem Kunden die korrekten Produkte zugewiesen sind

14.6.7.3. Testergebnis

✓ / OK

14.6.8. Userstory 8

Als Partner möchte ich eine Wissensdatenbank über häufig auftretende Fragen und Probleme (FAQ).

14.6.8.1. Logischer Testfall

Eintrag der Wissensdatenbank öffnen.

14.6.8.2. Konkreter Testfall

- Als Benutzer anmelden
- Im Dashboard im Bereich „Allgemein“ den Punkt "FAQ" wählen



- Überprüfen, ob FAQs verfügbar sind

14.6.8.3. Testergebnis

✓ / OK

14.6.9. Userstory 9

Als Partner möchte ich eine Übersicht aller Produkte und etwaiger Aktionen.

14.6.9.1. Logischer Testfall

Produktübersicht öffnen.

14.6.9.2. Konkreter Testfall

- Als Benutzer anmelden
- Im Dashboard im Bereich „Allgemein“ den Punkt "Produkte" wählen
- Überprüfen, ob Produkte verfügbar sind

14.6.9.3. Testergebnis

✓ / OK

14.6.10. Userstory 10

Als Partner möchte ich die Möglichkeit für eine aktuelle Provisionsübersicht und einen regelmäßigen Provisionsbericht erhalten.

14.6.10.1. Logischer Testfall

Rechnungsübersicht öffnen und Provisionsbericht herunterladen.

14.6.10.2. Konkreter Testfall

- Als Partner anmelden
- Im Dashboard im Bereich „Allgemein“ den Punkt Provisionsübersicht wählen
- Überprüfen, ob korrekte Einträge verfügbar sind
- Provisionsbericht herunterladen und auf Korrektheit prüfen.

14.6.10.3. Testergebnis

✓ / OK



14.6.11. Testfall: Berechtigungsverwaltung / Partner

14.6.11.1. Logischer Testfall

- Partner darf nur auf seine Kunden Zugriff haben
- Partner darf keine Administrationsmenus sehen können
- Partner muss Zugriff auf die ihm zugewiesenen Menüpunkte haben

14.6.11.2. Konkreter Testfall

- Als Administrator anmelden
- Im Dashboard zur Administration wechseln
- Benutzerverwaltung öffnen
- Neuen Benutzer erstellen
- Im Dashboard zur Administration wechseln
- Menüpunkt "Berechtigungen" öffnen
- Benutzergruppe "Partner" erstellen
- Der Benutzergruppe die notwendigen Berechtigungen zuteilen
- Benutzer zur Gruppe "Partner" hinzufügen
- Mit Benutzerkonto anmelden
- Kontrolle: Zugriff nur auf eigene Kunden möglich
- Kontrolle: Administrationsmenu darf nicht sichtbar sein
- Kontrolle: Zugriff auf zugewiesene Menüpunkte möglich

14.6.11.3. Testergebnis

✓ / OK

14.6.12. Testfall: Berechtigungsverwaltung - Administrator

14.6.12.1. Logischer Testfall

- Administrator muss auf alle Kunden Zugriff haben
- Administrator muss Administrationsmenus sehen können
- Administrator muss Zugriff auf sämtliche Menüpunkte haben



14.6.12.2. Konkreter Testfall

- Als Administrator anmelden
- Im Dashboard im Bereich Administration den Punkt Benutzerverwaltung öffnen
- Neuen Benutzer erstellen
- Im Dashboard zur Administration wechseln
- Menüpunkt "Berechtigungen" öffnen
- Benutzer zur Gruppe "Administratoren" hinzufügen
- Mit neuem Administratorkonto anmelden
- Kontrolle: Zugriff auf sämtliche Kunden
- Kontrolle: Administrationsmenu muss sichtbar sein
- Kontrolle: Zugriff auf sämtliche Menüpunkte möglich

14.6.12.3. Testergebnis

 / OK

14.6.13. Testfall: Verfügbarkeitscheck

14.6.13.1. Logischer Testfall

- Prüfen, ob eine Domain noch verfügbar ist

14.6.13.2. Konkreter Testfall

- Als Benutzer anmelden
- Im Dashboard den Menüpunkt "Verfügbarkeitscheck" wählen
- Verfügbaren Domainnamen eingeben
- Kontrolle: Domain muss als verfügbar angezeigt werden
- Vergebenen Domainnamen eingeben
- Kontrolle: Domain muss als „vergeben“ angezeigt werden.

14.6.13.3. Testergebnis

 / OK



14.6.14. Testfall: Rechnungsübersicht

14.6.14.1. Logischer Testfall

- Rechnungen sind verfügbar
- Rechnungen können als PDF heruntergeladen werden

14.6.14.2. Konkreter Testfall

- Als Benutzer anmelden
- Im Dashboard den Menüpunkt "Rechnungsübersicht" wählen
- Rechnung öffnen und auf Korrektheit prüfen
- Rechnung per "Download"-Symbol herunterladen und auf Korrektheit prüfen

14.6.14.3. Testergebnis

✓ / OK

14.6.15. Testfall: Produkt kündigen

14.6.15.1. Logischer Testfall

- Kündigen eines Produkts

14.6.15.2. Konkreter Testfall

- Als Benutzer anmelden
- Im Dashboard im Bereich „Allgemein“ den Menüpunkt Kundenübersicht wählen
- Kunden auswählen und Detailinformationen anzeigen
- Unter Produktinformation bestehendes Produkt kündigen
- Überprüfen, ob Produktkündigung erfolgreich durchlaufen wird

14.6.15.3. Testergebnis

✓ / OK



PartnerDashboard

Literatur- und Quellenhinweise





15. Literatur- und Quellenhinweise

Entwicklungsumgebung (13.1.1.)

Raspberry Pi 2 B (www.cdn-reichelt.de) - 18.04.2017

LAMP (www.unixmen.com) – 18.04.2017

Livesystem (13.1.2.)

Dell Server (www.optodata.com) – 18.04.2017

Absicherung mittels Transport Layer Security (13.1.3.)

TLS (www.elektronik-kompendium.de) – 18.04.2017

Verwendete Tools (14.)

Alle unter Punkt 14 (Verwendete Tools) verwendeten Logos stammen von den offiziellen Herstellerseiten.

Teststrategie (14.3.)

Qualitätsmanagement-PDCA-Zyklus (<https://commons.wikimedia.org/wiki/File:PDCA-Cycle-Kaizen.png>) – 24.04.2017



PartnerDashboard

Abbildungsverzeichnis





16. Abbildungsverzeichnis

Abbildung 1 - MVP.....	22
Abbildung 2 - Logo perfect:net.....	24
Abbildung 3 - Manuel Sander.....	28
Abbildung 4 - Gernot Krems.....	28
Abbildung 5 - Johannes Zach.....	28
Abbildung 6 - IST-Prozess	30
Abbildung 7 - IST-Produktbestellung.....	31
Abbildung 8 - IST-Bestellprozess	32
Abbildung 9 - IST-Einrichtungsprozess	32
Abbildung 10 - IST-Verrechnungsprozess	33
Abbildung 11 - SOLL-Bestellprozess	36
Abbildung 12 - SOLL-Einrichtungsprozess	36
Abbildung 13 - SOLL-Verrechnungsprozess	37
Abbildung 14 - SOLL-Domainbestellung.....	37
Abbildung 15 - Ablaufdiagramm	42
Abbildung 17 - Mockups-Login.....	44
Abbildung 18 - Mockups-Kundenübersicht.....	45
Abbildung 19 - Mockups-Kunden anlegen	45
Abbildung 20 - Mockups-Kundeninformationen	45
Abbildung 21 - Mockups-Produktdetails.....	45
Abbildung 22 - Mockups-Provisionen	46
Abbildung 23 - Mockups-Produkt hinzufügen	46
Abbildung 24 - Mockups-FAQ.....	46
Abbildung 25 - Mockups-News	46
Abbildung 26 - Mockups-Rechnungsübersicht	47
Abbildung 27 - Mockups-Benutzerverwaltung	47
Abbildung 28 - Gantt Diagramm	50
Abbildung 29 - Sprint-Backlog	52
Abbildung 30 - Logo-Jira.....	55
Abbildung 31 - Logo-Gantter	56



Abbildung 32 - Logo-yED	56
Abbildung 33 - Logo-Pencil.....	56
Abbildung 34 - Logo-Wunderlist	57
Abbildung 35 - Logo-Google Drive	57
Abbildung 36 - Logo-Microsoft Word.....	58
Abbildung 37 - Logo-Microsoft Excel	58
Abbildung 38 - Logo-Microsoft PowerPoint.....	58
Abbildung 39 - Logo-Atom	59
Abbildung 40 - Logo-Apache	59
Abbildung 41 - Logo-MySQL Workbench	60
Abbildung 42 - Logo-Mozilla Firefox.....	60
Abbildung 43 - Logo-Google Chrome	60
Abbildung 44 - Logo-Microsoft Edge.....	60
Abbildung 45 - Logo-Microsoft Internet Explorer	60
Abbildung 46 - Logo-Bitbucket.....	61
Abbildung 47 - Logo-Source Tree	61
Abbildung 48 - Logo-Skype.....	62
Abbildung 49 - Logo-WhatsApp	62
Abbildung 50 - Raspberry Pi 2B	64
Abbildung 51 - LAMP	64
Abbildung 52 - Dell Server.....	65
Abbildung 53 - TLS.....	66
Abbildung 54 - Softwarearchitektur.....	68
Abbildung 55 - VO-Contact.....	69
Abbildung 56 - VO-Invoice.....	69
Abbildung 57 - VO-Contract	69
Abbildung 58 - VO-Account.....	70
Abbildung 59 - VO-Address	70
Abbildung 60 - VO-BulletinBoard	70
Abbildung 61 - VO-Country	70
Abbildung 62 - VO-Currency.....	70
Abbildung 63 - VO-Settings	70



Abbildung 64 - VO-RequestType	70
Abbildung 65 - VO-Faq	70
Abbildung 66 - VO-TaskType	70
Abbildung 67 - VO-Log.....	71
Abbildung 68 - VO-Product	71
Abbildung 69 - VO-Request.....	71
Abbildung 70 - VO-TaskInquiry.....	71
Abbildung 71 - VO-TaskParameter.....	71
Abbildung 72 - VO-Permission	71
Abbildung 73 - VO-Order.....	71
Abbildung 74 - VO-Task	71
Abbildung 75 - VO-ProductDetail.....	72
Abbildung 76 - VO-GroupMember.....	72
Abbildung 77 - VO-OrderedProduct.....	72
Abbildung 78 - VO-BulletinBoardDelivery.....	72
Abbildung 79 - VO-GroupPermission	72
Abbildung 80 - VO-Metadata	72
Abbildung 81 - Datenbankdesign	73
Abbildung 82 - Qualitätsmanagement-PDCA-Zyklus.....	79
Abbildung 83 - Qualitätsmanagement-VO-Account	79
Abbildung 84 - Qualitätsmanagement-VO-Address.....	79
Abbildung 85 - Qualitätsmanagement-VO-BulletinBoard.....	80
Abbildung 86 - Qualitätsmanagement-VO-Contact	80
Abbildung 87 - Qualitätsmanagement-VO-Contract.....	81
Abbildung 88 - Qualitätsmanagement-VO-Currency	81
Abbildung 89 - Qualitätsmanagement-VO-FAQ.....	81
Abbildung 90 - Qualitätsmanagement-VO-Invoice	82
Abbildung 91 - Qualitätsmanagement-VO-Log	82
Abbildung 92 - Qualitätsmanagement-VO-Order	83
Abbildung 93 - Qualitätsmanagement-VO-Product.....	83
Abbildung 94 - Qualitätsmanagement-VO-Productdetail.....	83
Abbildung 95 - Qualitätsmanagement-Integrationstest-Manage Customer.....	84



Abbildung 96 - Qualitätsmanagement-Integrationstest-Manage Account	84
Abbildung 97 - Qualitätsmanagement-Integrationstest-Manage FAQ.....	84
Abbildung 98 - Qualitätsmanagement-Integrationstest-Manage BulletinBoard	85
Abbildung 99 - Qualitätsmanagement-Integrationstest-Order Product	85
Abbildung 100 - Qualitätsmanagement-Integrationstest-Access Management.....	85
Abbildung 101 - Projektantrag	116
Abbildung 102 - Benutzerhandbuch-Login.....	118
Abbildung 103 - Benutzerhandbuch-Two-Factor Authentication.....	118
Abbildung 104 - Benutzerhandbuch-Startseite.....	119
Abbildung 105 - Benutzerhandbuch-Navigation.....	119
Abbildung 106 - Benutzerhandbuch-Dashboard.....	120
Abbildung 107 - Benutzerhandbuch-Kundenübersicht-Logo.....	121
Abbildung 108 - Benutzerhandbuch-Kundenübersicht.....	121
Abbildung 109 - Benutzerhandbuch-Kunden anlegen	122
Abbildung 110 - Benutzerhandbuch-Produktinformationen	122
Abbildung 111 - Benutzerhandbuch-Kundeninformationen	123
Abbildung 112- Benutzerhandbuch-Kunden bearbeiten	123
Abbildung 113- Benutzerhandbuch-Adresse hinzufügen	123
Abbildung 114- Benutzerhandbuch-Adresse ändern.....	123
Abbildung 115 - Benutzerhandbuch-Kunden auswählen-Logo	124
Abbildung 116 - Benutzerhandbuch-Kunden auswählen	124
Abbildung 117 - Benutzerhandbuch-Rechnungsübersicht	124
Abbildung 118 - Benutzerhandbuch-Musterrechnung	125
Abbildung 119 - Benutzerhandbuch-Produktübersicht-Logo	126
Abbildung 120 - Benutzerhandbuch-Produktübersicht	126
Abbildung 121 - Benutzerhandbuch-News-Logo	127
Abbildung 122 - Benutzerhandbuch-News	127
Abbildung 123 - Benutzerhandbuch-FAQ-Logo	127
Abbildung 124 - Benutzerhandbuch-FAQ	127
Abbildung 125 - Administrationshandbuch-Basiskonfiguration	129
Abbildung 126 - Administrationshandbuch-Dashboard	129
Abbildung 127 - Administrationshandbuch-Benutzerverwaltung-Logo	130



Abbildung 128 - Administrationshandbuch-Benutzerverwaltung	130
Abbildung 129 - Administrationshandbuch-Benutzeranlage.....	130
Abbildung 130 - Administrationshandbuch-Benutzerverwaltung 2	131
Abbildung 131 - Administrationshandbuch-TFA	131
Abbildung 132 - Administrationshandbuch-QR-Code.....	131
Abbildung 133 - Administrationshandbuch-Berechtigungen-Logo	132
Abbildung 134 - Administrationshandbuch-Berechtigungen	132
Abbildung 135 - Administrationshandbuch-Berechtigungsgruppenmitglieder.....	132
Abbildung 136 - Administrationshandbuch-Requests-Logo	133
Abbildung 137 - Administrationshandbuch-Requests	133
Abbildung 138 - Administrationshandbuch-Requestanfragen-Logo	134
Abbildung 139 - Administrationshandbuch-Request Inquiry.....	134
Abbildung 140 - Administrationshandbuch-News Verwaltung-Logo	135
Abbildung 141 - Administrationshandbuch-News	135
Abbildung 142 - Administrationshandbuch-Newseinträge bearbeiten.....	135
Abbildung 143 - Administrationshandbuch-FAQ Verwaltung-Logo.....	136
Abbildung 144 - Administrationshandbuch-FAQ	136
Abbildung 145 - Administrationshandbuch-FAQ Kategorien.....	136
Abbildung 146 - Administrationshandbuch-FAQ Kategorie Auswahl	137
Abbildung 147 - Administrationshandbuch-FAQ Filter	137
Abbildung 148 - Sprintstatusbericht-Ergebnis	147
Abbildung 149 - Sprintstatusbericht-Ergebnis	149
Abbildung 150 - Sprintstatusbericht-Übersicht	151
Abbildung 151 - Sprintstatusbericht-Übersicht	153
Abbildung 152 - Sprintstatusbericht-Übersicht	155
Abbildung 153 - Sprintstatusbericht-Übersicht	157
Abbildung 154 - Sprintstatusbericht-Übersicht	159
Abbildung 155 - Ordnerstruktur 1-100	167
Abbildung 156 - Ordnerstruktur 101-200	167
Abbildung 157 - Ordnerstruktur 201-300	168
Abbildung 158 - Ordnerstruktur 301-400	168
Abbildung 159 - Ordnerstruktur 401-500	169



Abbildung 160 - Ordnerstruktur 501-600	169
Abbildung 161 - Ordnerstruktur 601-700	170
Abbildung 162 - Ordnerstruktur 701-751	170



PartnerDashboard

Tabellenverzeichnis





17. Tabellenverzeichnis

Tabelle 1 – Userstories	38
Tabelle 2 - Variantenentscheid-Programmiersprache	39
Tabelle 3 - Variantenentscheid-Authentication	40
Tabelle 4 - Meilensteinplan	49
Tabelle 5 - Stakeholder Analyse	51
Tabelle 6 - Begleitprotokoll	114
Tabelle 7 - Stundennachweis-Gernot Krems-Übersicht.....	139
Tabelle 8 - Stundennachweis-Gernot Krems-PTD	139
Tabelle 9 - Stundennachweis-Gernot Krems-Stunden pro Monat.....	140
Tabelle 10 - Stundennachweis-Manuel Sander-Übersicht.....	141
Tabelle 11 - Stundennachweis-Manuel Sander-PTD.....	141
Tabelle 12 - Stundennachweis-Manuel Sander-Stunden pro Monat	142
Tabelle 13 - Stundennachweis-Johannes Zach-Übersicht.....	143
Tabelle 14 - Stundennachweis-Johannes Zach-PTD	143
Tabelle 15 - Stundennachweis-Johannes Zach-Stunden pro Monat	144
Tabelle 16 - Sprintstatusbericht-Sprint 1.1	146
Tabelle 17 - Sprintstatusbericht-Sprint 1.2	146
Tabelle 18 - Sprintstatusbericht-Sprint 1.3	146
Tabelle 19 - Sprintstatusbericht-Sprint 1.4	146
Tabelle 20 - Sprintstatusbericht-Sprint 1.5	147
Tabelle 21 - Sprintstatusbericht-Sprint 1.6	147
Tabelle 22 - Sprintstatusbericht-Sprint 2.1	148
Tabelle 23 - Sprintstatusbericht-Sprint 2.2	148
Tabelle 24 - Sprintstatusbericht-Sprint 2.3	148
Tabelle 25 - Sprintstatusbericht-Sprint 2.4	148
Tabelle 26 - Sprintstatusbericht-Sprint 2.5	149
Tabelle 27 - Sprintstatusbericht-Sprint 2.6	149
Tabelle 28 - Sprintstatusbericht-Sprint 3.1	150
Tabelle 29 - Sprintstatusbericht-Sprint 3.2	150
Tabelle 30 - Sprintstatusbericht-Sprint 3.3	150



Tabelle 31 - Sprintstatusbericht-Sprint 3.4	150
Tabelle 32 - Sprintstatusbericht-Sprint 3.5	151
Tabelle 33 - Sprintstatusbericht-Sprint 3.6	151
Tabelle 34 - Sprintstatusbericht-Sprint 4.1	152
Tabelle 35 - Sprintstatusbericht-Sprint 4.2	152
Tabelle 36 - Sprintstatusbericht-Sprint 4.3	152
Tabelle 37 - Sprintstatusbericht-Sprint 4.4	152
Tabelle 38 - Sprintstatusbericht-Sprint 4.5	153
Tabelle 39 - Sprintstatusbericht-Sprint 4.6	153
Tabelle 40 - Sprintstatusbericht-Sprint 5.1	154
Tabelle 41 - Sprintstatusbericht-Sprint 5.2	154
Tabelle 42 - Sprintstatusbericht-Sprint 5.3	154
Tabelle 43 - Sprintstatusbericht-Sprint 5.4	154
Tabelle 44 - Sprintstatusbericht-Sprint 5.5	155
Tabelle 45 - Sprintstatusbericht-Sprint 5.6	155
Tabelle 46 - Sprintstatusbericht-Sprint 6.1	156
Tabelle 47 - Sprintstatusbericht-Sprint 6.2	156
Tabelle 48 - Sprintstatusbericht-Sprint 6.3	156
Tabelle 49 - Sprintstatusbericht-Sprint 6.4	156
Tabelle 50 - Sprintstatusbericht-Sprint 6.5	157
Tabelle 51 - Sprintstatusbericht-Sprint 6.6	157
Tabelle 52 - Sprintstatusbericht-Sprint 7.1	158
Tabelle 53 - Sprintstatusbericht-Sprint 7.2	158
Tabelle 54 - Sprintstatusbericht-Sprint 7.3	158
Tabelle 55 - Sprintstatusbericht-Sprint 7.4	158
Tabelle 56 - Sprintstatusbericht-Sprint 7.5	159
Tabelle 57 - Sprintstatusbericht-Sprint 7.6	159
Tabelle 58 - Meeting Protokolle-Kickoff Meeting-1.....	161
Tabelle 59 - Meeting Protokolle-Kickoff Meeting-2.....	161
Tabelle 60 - Meeting Protokolle-Kickoff Meeting-3.....	162
Tabelle 61 - Meeting Protokolle-Kickoff Meeting-4.....	162
Tabelle 62 - Meeting Protokolle-Kickoff Meeting-5.....	162



Tabelle 63 - Meeting Protokolle-Kickoff Meeting-6.....	163
Tabelle 64 - Meeting Protokolle-Meeting Schnittstelle-1.....	164
Tabelle 65 - Meeting Protokolle-Meeting Schnittstelle-2.....	164
Tabelle 66 - Meeting Protokolle-Meeting Schnittstelle-3.....	165



PartnerDashboard

Zusammenfassung





18. Zusammenfassung

Herr Biernat vertreibt neben seiner Tätigkeit als Fotograf und Webentwickler als Geschäftsführer seines Einzelunternehmens perfect:net mit Hilfe von Partnern Webhosting-Pakete und Domains. Im derzeitigen System gehen alle Anfragen und Produktbestellungen von Partnern direkt an den ihn, und er muss die Produkt- und Kundendaten manuell in sein System eingeben und die Bestellung durchführen. Dieser Zustand ist für einen Einmannbetrieb nur bis zu einem gewissen Punkt bewältigbar. Dieser ist mittlerweile für Herr Biernat erreicht, was sich dadurch äußert, dass regelmäßig Aufträge aufgrund von Zeitmangel nicht angenommen werden können.

Durch intensives Brainstorming über ein System, das ihm einen Großteil der Standardaufgaben abnehmen könnte, entstand die erste Idee des PartnerDashboards. In dem Zusammenhang wurde auch der Funktionsumfang des Programmes geklärt. Herr Biernat war es wichtig, dass es sich um eine Website handelt, damit sie von Partnern von überall erreichbar ist, und dass die von den Partnern eingetragenen Daten in sein derzeitiges Backend automatisch übertragen werden. Ein weiteres Anliegen des Auftraggebers war auch, dass das PartnerDashboard modular aufgebaut ist. Damit soll gewährleistet werden, dass im ersten Schritt zwar alle Vorgänge automatisch ablaufen, er jedoch alle übertragenen Daten nochmals kontrollieren und gegebenenfalls ändern kann, bevor die Bestellung dann manuell durch einen „Abschließen“ Button tatsächlich abgesendet wird. Diese Funktion wird nach einer gewissen Übergangsphase deaktiviert werden.

Realisiert wurde das PartnerDashboard als zentrales Werkzeug das als Bindeglied zwischen perfect:net, Partnern und deren Kunden fungiert. Vorgefertigte Web Hosting Pakete können auf Knopfdruck bestellt und einem erfassten Kunden zugewiesen werden. Die Bestellprozesse wurden optimiert und die dazugehörige Rechnungslegung erfolgt automatisiert in einheitlicher Form.

Das PartnerDashboard stellte uns als Team vor eine sehr große Herausforderung, da wir bis zu diesem Zeitpunkt sehr wenig Erfahrung mit der Entwicklung einer Website und den damit verbundenen Technologien hatten.



PartnerDashboard

Ausblick





19. Ausblick

Der erste Schritt ist die Installation der Datenbank mit dem Installationsmodul. Daraufhin wird die Anwendung auf einem Server des Auftraggebers installiert. Der nächste Schritt ist die Initialbefüllung der Datenbank des PartnerDashboards mit dem aktuellen Datenstand der Daten im Backend des Auftraggebers.

Darauf folgt eine zweiwöchige Testphase, in der mit dem Auftraggeber und einem ausgewählten Partner alle Funktionen und Einstellungsmöglichkeiten durchgearbeitet werden, um gegebenenfalls noch kleine Anpassungen vornehmen zu können.

Nach dieser Phase wird das PartnerDashboard für alle Partner erreichbar sowie nutzbar sein. Hierfür sind ebenfalls zwei Wochen geplant, um allen Partnern eine Eingewöhnungsphase mit Support bieten zu können.

Wenn das System allen Anforderungen des Auftraggebers genügt und die Partner sich eingearbeitet haben, werden diverse Kontrollmechanismen ausgeschaltet, sodass alle Vorgänge automatisch ablaufen.

Durch den modularen Aufbau des PartnerDashboards kann es problemlos erweitert und gegebenenfalls auch für andere Unternehmen angepasst werden.



PartnerDashboard

Persönliches Fazit





20. Persönliches Fazit

Im Juni 2016 stand fest, dass wir – Manuel Sander, Gernot Krems und Johannes Zach – ein Team bilden würden. Der Zusammenschluss ist nicht nur aufgrund unserer guten Beziehung zueinander entstanden, sondern resultierte auch daraus, dass jedes Teammitglied von uns in einem anderen Bereich schon wertvolle Erfahrungen gesammelt hat. Nach kurzer Suche nach einem geeigneten Projektpartner für unsere Diplomarbeit konnten wir unseren jetzigen Auftraggeber perfect:net gewinnen.

Da unser Team entschieden hat, unser Projekt in PHP und den damit verbundenen Technologien zu programmieren, stellte uns dies zu Beginn des Projektes vor eine neue Herausforderung, die wir zu unserer größten Zufriedenheit gemeistert haben. Das aufgebaute Wissen können wir außerdem sehr gut für zukünftige Projekte verwenden.

Rückblickend ist zu sagen, dass durch diese Projektarbeit jeder von uns einen guten Einblick in die Softwareentwicklung in einem Team erhalten hat. Auch zwischenmenschliche Konflikte sowie zeitliche Engpässe konnten uns nicht vom Kurs abbringen, sondern schweißten uns nur enger zusammen.

Wir werden die vergangene Zeit sicher noch lange in positiver Erinnerung behalten, freuen uns aber schon auf neue Ziele.



PartnerDashboard

Anhang





21. Anhang

21.1. Begleitprotokoll

Thema des übergeordneten komplexen Aufgabenbereichs oder Projekts

PartnerDashboard

Individuelle Themenstellung

Programmierung eines Kundenverwaltungstools

Betreuer

Prof. Dipl.-Ing. Dr. Thomas Quaritsch, BSc

E-Mail-Adresse: thomas.quaritsch@htlpinkafeld.at

Telefonnummer: +43 3357 / 42491

Name des Diplomanden

Gernot Krems

E-Mail-Adresse: gernot.krems@htlpinkafeld.at

Telefonnummer: +43 676 / 75 15 432

Name des Kooperationspartners

Dieter Biernat

E-Mail-Adresse: dieter.biernat@perfectnet.at

Telefonnummer: +43 650 / 33 66 605

Teammitglieder: Manuel Sander, Gernot Krems, Johannes Zach

Datum der Besprechung	Teilnehmer der Besprechung	Vereinbarung	Termin zur Erledigung
23.09.2016	Thomas Quaritsch, Manuel Sander, Gernot Krems, Johannes Zach	ER Modellierung	21.10.2016
21.10.2016	Thomas Quaritsch, Manuel Sander, Gernot Krems, Johannes Zach	Grundstruktur Backend	09.12.2016
09.12.2016	Thomas Quaritsch, Manuel Sander, Gernot Krems, Johannes Zach	Grundstruktur Frontend	27.01.2017
27.01.2017	Thomas Quaritsch, Manuel Sander, Gernot Krems, Johannes Zach	Finalisierung Frontend, Backend	24.03.2017
24.03.2017	Thomas Quaritsch, Manuel Sander, Gernot Krems, Johannes Zach	Unity Test Durchsprache	07.04.2017

Tabelle 6 - Begleitprotokoll

Unterschrift des Lehrers: Dr. Thomas Quaritsch

Datum: 14.04.2017

Unterschrift des Schülers: Manuel Sander



21.2. Abnahme Design

PartnerDashboard

Abnahme Design

Auftraggeber: Dieter Biernat, MSc

Auftragnehmer: Manuel Sander, Gernot Krems, Johannes Zach

Auftrag: PartnerDashboard

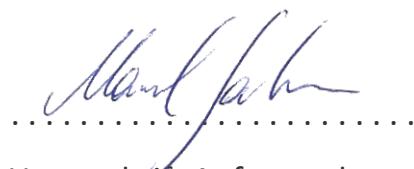
Nach der Durchsicht/Durchsprache der Mockups am 05.10.2016, wurde das Design vom Auftraggeber abgenommen. Der Auftragnehmer behält sich das Recht vor, gegebenenfalls kleine Veränderungen am Design durchzuführen, um die technischen Anforderungen des Projekts realisieren zu können.

Graz, 27.10.2016

Ort, Datum

.....


Unterschrift Auftraggeber

.....


Unterschrift Auftragnehmer



21.3. Projektantrag



Erklärung

Die unterfertigten Kandidaten/Kandidatinnen haben gemäß § 34 Abs. 3 Z 1 und § 37 Abs. 2 Z 2 des Schulunterrichtsgesetzes in Verbindung mit den Bestimmungen der „Prüfungsordnung BMHS, Bildungsanstalten“, BGBl. II Nr. 177/2012 i.d.g.F. die Ausarbeitung einer Diplomarbeit/Abschlussarbeit mit folgender Aufgabenstellung gewählt:

PartnerDashboard (Gesamtprojekt)

Individuelle Aufgabenstellungen im Rahmen des Gesamtprojektes:

- Gernot Krems (6ABIF): **Login – Two Factor Authentication, Kundenverwaltung Frontend, Technische Anbindung Backend**
- Johannes Zach (6ABIF): **Externe Schnittstellen, Access Management, Order Management, Online Dokumentation (FAQ)**
- Manuel Sander (6ABIF): **Rechnungsmodul, Kundenverwaltung Backend, Installations- und Updatemodul**

Die Kandidaten/Kandidatinnen nehmen zur Kenntnis, dass die Diplomarbeit/Abschlussarbeit in eigenständiger Weise und außerhalb des Unterrichtes zu bearbeiten und anzufertigen ist, wobei Ergebnisse des Unterrichtes mit einbezogen werden können, die jedenfalls als solche entsprechend kenntlich zu machen sind.

Die Abgabe der vollständigen Diplomarbeit/Abschlussarbeit hat in digitaler und in zweifach ausgedruckter Form bis spätestens **05.04.2017** beim zuständigen Betreuer/der zuständigen Betreuerin zu erfolgen.

Die Kandidaten/Kandidatinnen nehmen weiters zur Kenntnis, dass ein Abbruch der Diplomarbeit/Abschlussarbeit nicht möglich ist.

Kandidaten/Kandidatinnen:

Gernot Krems (6ABIF)

Johannes Zach (6ABIF)

Manuel Sander (6ABIF)

Datum und Unterschrift bzw. Handysignatur:

24.09.2016, Gernot Krems

24.09.2016, Johannes Zach

24.09.2016, Manuel Sander

Abbildung 100 - Projektantrag



PartnerDashboard

Benutzerhandbuch





21.4. Benutzerhandbuch

21.4.1. Login

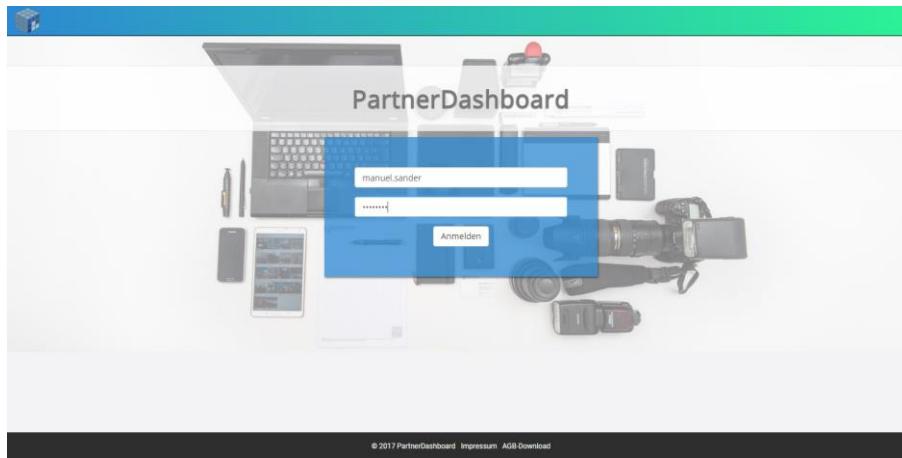


Abbildung 101 - Benutzerhandbuch-Login

Der Partner meldet sich mit Benutzername und Passwort an. Falls die optionale Two-Factor Authentifizierung aktiv ist, erhält der Benutzer zusätzlich eine Eingabemaske für einen Security-Code, der über den Google Authenticator Service verifiziert wird.

21.4.2. Two-Factor Authentication

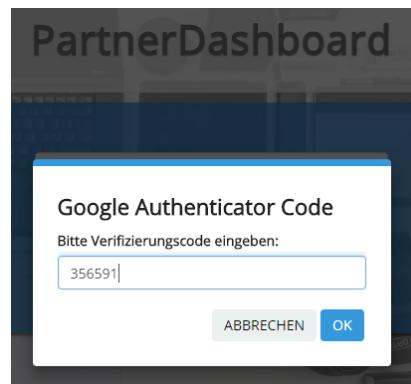


Abbildung 102 - Benutzerhandbuch-Two-Factor Authentication



21.4.3. Startseite

The screenshot shows the PartnerDashboard start page. At the top, there is a navigation bar with links for Dashboard, Kundenübersicht, Rechnungsübersicht, and News. On the right side of the navigation bar, there are links for Kunde, Warenkorb (0), FAQ, and Sander, Manuel. Below the navigation bar, the main content area is titled "Dashboard" and "PartnerDashboard". It contains two main sections: "Allgemein" and "Utilities". The "Allgemein" section includes links for Kundenübersicht, Rechnungsübersicht, Produkte, Statistik, News, and FAQ. The "Utilities" section includes a link for Verfügbarkeitscheck. At the bottom of the page, there is a footer with copyright information: © 2017 partner.dashboard Impressum AGB-Download.

Abbildung 103 - Benutzerhandbuch-Startseite

Die Startseite nach der Anmeldung ist das Dashboard, in dem die verfügbaren Module gelistet sind.

21.4.4. Navigation

The screenshot shows the PartnerDashboard navigation bar. It includes links for Dashboard, Kundenübersicht, Rechnungsübersicht, and News. On the right side, there are links for Kunde, Warenkorb (0), FAQ, and Benutzerverwaltung. The "Benutzerverwaltung" link is highlighted in blue, indicating it is the current active page.

Abbildung 104 - Benutzerhandbuch-Navigation

In der Navigation findet man zuerst das Dashboard. Dies ist die Seite, die direkt nach dem Anmelden erscheint. „Kundenübersicht“, „Rechnungsübersicht“, „News“ und „FAQ“ sind die für den Partner wichtigsten Funktionen des PartnerDashboards. Bei „Kunde“ ist der derzeit ausgewählte Kunde angeführt und bei „Warenkorb“ sieht man die Produkte, welche man für den ausgewählten Kunden gespeichert hat. Der letzte Punkt der Navigation betrifft die „Benutzerverwaltung“ selbst. Hier können Benutzereinstellungen vorgenommen werden, und der Partner kann sich vom System abmelden.



21.4.5. Dashboard

The screenshot shows the PartnerDashboard interface. At the top, there is a navigation bar with links for 'Dashboard', 'Kundenübersicht', 'Rechnungsübersicht', 'News', 'Kunde: Sander, Manuel', 'Warenkorb: 0', 'FAQ', and a dropdown menu for 'Sander, Manuel'. Below the navigation bar, the main area is titled 'Dashboard' and 'PartnerDashboard'. It features two main sections: 'Allgemein' and 'Utilities'. The 'Allgemein' section contains links for 'Kundenübersicht', 'Rechnungsübersicht', 'Produkte', 'Statistik', 'News', and 'FAQ'. The 'Utilities' section contains a single link for 'Verfügbarkeitscheck'. At the bottom of the page, there is a footer with the text '© 2017 partner.dashboard Impressum AGB-Download'.

Abbildung 105 - Benutzerhandbuch-Dashboard

Das Dashboard enthält alle Funktionen, die dem Partner zu Verfügung stehen. Im allgemeinen Bereich stehen die Funktionen „Kundenübersicht“, „Rechnungsübersicht“, „Produkte“, „Statistik“, „News“ und „FAQ“ zur Auswahl.

Im Bereich Tools ist der „Verfügbarkeitscheck“ auswählbar.



21.4.5.1. Kundenübersicht



Abbildung 106 - Benutzerhandbuch-Kundenübersicht-Logo

Nr.	KundenNr	Vorname	Nachname	Email	Kunde Seit	Aktiv
1	KU1001	johannes	Zach	zach.johannes@gmx.net	2017-01-08 11:32:13	1
2	KU1002	Manuel	Sander	manuel.sander1@gmx.net		1
3	KU1003	Gernot	Krems	gernot.krems@layer8.at		1
4	KU1004	Franz	Bertl			1
5	KU1016	Richard	Williamson	rwilliamson1@wp.com	2017-01-08 14:22:29	
6	KU1019	Melissa	Vasquez	mvasquez4@who.int	2017-01-08 14:22:29	
7	KU1020	Billy	Richards	brichards5@vinaora.com	2017-01-08 14:22:29	1
8	KU1021	Irene	Hall	ihall6@zmodo.com	2017-01-08 14:22:29	1
9	KU1022	Laura	Olson	lolson7@hhs.gov	2017-01-08 14:22:29	1
10	KU1023	Michelle	Mccoy	mmccoy@bravesites.com	2017-01-08 14:22:29	1

Abbildung 107 - Benutzerhandbuch-Kundenübersicht

In der Kundenübersicht sieht der Partner alle ihm zugeordneten Kunden. Diese können beliebig in der Tabelle sortiert werden. Interessant ist vor allem die Spalte „Aktiv“. Der Wert 0 bedeutet, dass der Kunde inaktiv ist, was wiederum bedeutet, dass keine weiteren Produkte verkauft werden können. Weiters gibt es für den Partner die Möglichkeit, einen neuen Kunden über den Button „Neuer Kunde“ anzulegen, die Informationen zu einem spezifischen Kunden anzeigen zu lassen , dem ausgewählten Kunden ein Produkt hinzuzufügen und einen Kunden gegebenenfalls zu löschen .



21.4.5.1.1. Neuer Kunde

The screenshot shows a form titled "Neuen Kunden anlegen" (Create New Customer). It includes fields for personal information like title, first name, last name, address, and phone numbers, as well as professional details like VAT and country. A "Speichern" (Save) button is at the bottom.

Abbildung 108 - Benutzerhandbuch-Kunden anlegen

21.4.5.1.2. Produktinformationen

Date	Product	Type	Total
2. Mai 2017	WebHosting: startUp	WebHosting Package	€4.17
	Total		€4.17
28. April 2017	WebHosting: Large	WebHosting Package	€25.00
	WebHosting: Medium	WebHosting Package	€16.67
	Total		€41.67
26. April 2017	WebHosting: Small	WebHosting Package	€8.33
	Total		€8.33
25. April 2017	WebHosting: Shop	WebHosting Package	€29.00
	Total		€29.00
25. April 2017	WebHosting: Small	WebHosting Package	€8.33

Abbildung 109 - Benutzerhandbuch-Produktinformationen

Die Produktinformationen zeigen alle Produkte, die der Kunde aktiv im Einsatz hat.



21.4.5.1.3. Kundeninformationen

Abbildung 110 - Benutzerhandbuch-Kundeninformationen

Die Kundeninformationen lassen sich unterteilen in persönliche Kundeninformationen und Produktinformationen.

Die persönlichen Kundeninformationen zeigen im linken Teil Informationen wie Name, Telefonnummer, Geburtsdatum usw. und im rechten Teil die Adressen der Kunden an.

Im linken Teil können die Informationen der Kunden bearbeitet werden. Im rechten Teil können Adressen hinzugefügt oder gegebenenfalls vorhandene

Adressen geändert werden. Es kann nur eine Standardadresse geben. Zusätzlich können Notizen zu den Kunden gespeichert werden.

Abbildung 111- Benutzerhandbuch-Kunden bearbeiten

Abbildung 112- Benutzerhandbuch- Adresse hinzufügen

Abbildung 113- Benutzerhandbuch- Adresse ändern



21.4.5.2. Rechnungsübersicht



Abbildung 114 - Benutzerhandbuch-Kunden auswählen-Logo

Kundennummer	Name	E-Mail	Letzter Login
KU1001	Zach, Johannes	zach.johannes@gmx.net	
<input checked="" type="checkbox"/> KU1002	Sander, Manuel	manuel.sander1@gmx.net	
KU1003	Krems, Gernot	gernot.krems@layer8.at	
KU1004	Bertl, Franz		2017-01-08 11:32:13
KU1016	Williamson, Richard	rwilliamson1@wp.com	2017-01-08 14:22:29
KU1019	Vasquez, Melissa	mvasquez@who.int	2017-01-08 14:22:29
KU1020	Richards, Billy	brichards5@vinaora.com	2017-01-08 14:22:29
KU1021	Hall, Irene	ihall6@giymodo.com	2017-01-08 14:22:29

Abbildung 115 - Benutzerhandbuch-Kunden auswählen

Bevor man Module wie die Rechnungsübersicht oder Produkte ausführen kann, muss man einen Kunden auswählen. Sollte dies nicht schon im Vorhinein geschehen sein, wird man automatisch auf die Kunden auswählen Seite umgeleitet.

Rechnung	Summe	Aktion
Rechnung 1002-201701	€99.83 von € 99.83	
Rechnung 1002-201702	€99.83 von € 99.83	
Rechnung 1002-201703	€99.83 von € 99.83	
Rechnung 1002-201704	€45.66 von € 99.83	
Rechnung 1002-201705	€0 von € 544.99	

Abbildung 116 - Benutzerhandbuch-Rechnungsübersicht

Hier werden alle Rechnungen des ausgewählten Kunden angezeigt. Sie stehen als PDF zum Download zur Verfügung.



Herr Manuel Sander
Dienersdorf 15
8224 Kaindorf

Rechnung

perfect:net
Dieter Biernat, MSc

[REDACTED]
[REDACTED]
[REDACTED]
lautend auf Dieter Biernat

Rechnung Nr.: 1002-201703

Rechnungsdatum: 2017-03-01
Fälligkeit: 2017-06-01

Sehr geehrter Herr Manuel Sander!

Entsprechend unserer Vereinbarung erlaube ich mir, Ihnen untenstehende Leistungen wie folgt zu verrechnen:

Leistungsumfang	Menge	Netto
WebHosting: Shop	1	24,17 €
WebHosting: Small	3	20,83 €
WebHosting: startUp	1	3,48 €
WebHosting: Medium	1	13,89 €
WebHosting: Large	1	20,83 €
	Summe netto:	83,19 €
	USt. (20%):	16,64 €
	Endbetrag brutto:	99,83 €

Bei Fragen wenden Sie sich bitte an billing@perfectnet.at

Bitte geben Sie bei einer Telebanking-Überweisung folgende Zahl in das Kundendaten-Feld ein: KU1002

perfect:net, design:foto:hosting, IT-Dienstleistungen, Werbegrafik-Design, Foto-Design, Einzelunternehmen Dieter Biernat, MSc,

Abbildung 117 - Benutzerhandbuch-Musterrechnung



21.4.5.3. Produktübersicht



Abbildung 118 - Benutzerhandbuch-Produktübersicht-Logo

The screenshot shows a product overview page titled 'Produkte' under 'PartnerDashboard / Produkte'. At the top, there are navigation links: Dashboard, Kundensicht, Rechnungsübersicht, News, Kunde: Sander, Manuel, Warenkorb: 0, FAQ, and Sander, Manuel. A dropdown menu 'Produkttypen: Webhosting Package' is open. Below the header, there are five product categories displayed in boxes:

Webhosting: startUp	Webhosting: Small	Webhosting: Medium	Webhosting: Large
€ 4.17	€ 8.33	€ 16.67	€ 25.00
One-Page-Porfolios und kleine Websites	Websites basierend auf Joomla, Wordpress, Typo3, Drupal, usw.	Websites mit vielen Inhalten und Bildern.	Große Websites mit sehr großen Datenmengen.
[In den Warenkorb]	[In den Warenkorb]	[In den Warenkorb]	[In den Warenkorb]

A sixth category, 'Webhosting: Shop', is shown below the others, with a price of € 29.00 and a note about webshop systems like Magento, PrestaShop, etc., followed by a '[In den Warenkorb]' button. At the bottom of the page, there is a footer bar with links: © 2017 partner.dashboard Impressum AGB Download.

Abbildung 119 - Benutzerhandbuch-Produktübersicht

Hier können dem ausgewählten Kunden alle zum Kauf angebotenen Pakete zugeordnet werden.



21.4.6.News



Abbildung 120 - Benutzerhandbuch-News-Logo

The screenshot shows a list of news items on a dashboard. Each item has a date, a title, and a message body. The first item is from 2017-01-08 with title 'Title#01' and message 'Message#01'. The second item is from 2017-01-07 with title 'Title#02' and message 'Message#02'. The third item is from 2017-01-06 with title 'Title#03' and message 'Message#03'.

Date	Title	Message
2017-01-08	Title#01	Message#01
2017-01-07	Title#02	Message#02
2017-01-06	Title#03	Message#03

Abbildung 121 - Benutzerhandbuch-News

Hier werden die vom Administrator erstellten Newseinträge wie Wartungsarbeiten, Systemausfälle und sonstige News angezeigt.

21.4.7.FAQ



Abbildung 122 - Benutzerhandbuch-FAQ-Logo

The screenshot shows a FAQ section with two main categories: 'ALLGEMEIN' and 'KUNDENVERWALTUNG'. Each category contains a list of questions with a dropdown arrow to the right of each. In 'ALLGEMEIN', the questions are: 'Wie kann ich mein Passwort ändern?', 'Wie kann ich die Two-Factor-Authentication aktivieren?', 'Wie kann ich einen neuen Kunden anlegen?', and 'Wie kann ich Produkte bestellen?'. In 'KUNDENVERWALTUNG', the question is: 'Wie kann ich einen Kunden auswählen?'.

Category	Question
ALLGEMEIN	Wie kann ich mein Passwort ändern?
ALLGEMEIN	Wie kann ich die Two-Factor-Authentication aktivieren?
ALLGEMEIN	Wie kann ich einen neuen Kunden anlegen?
ALLGEMEIN	Wie kann ich Produkte bestellen?
KUNDENVERWALTUNG	Wie kann ich einen Kunden auswählen?

Abbildung 123 - Benutzerhandbuch-FAQ

Hier können Partner die Frequently Asked Questions nachlesen, um etwaige Fragen nicht immer an den Support richten zu müssen. Die FAQs werden vom Administrator angelegt.



PartnerDashboard

Administrationshandbuch





21.5. Administrationshandbuch

Das Administrationshandbuch ist ausschließlich für verwaltende Personen des PartnerDashboards gedacht. Um einen ordnungsgemäßen Betrieb gewährleisten zu können steht die Administrationsoberfläche nur berechtigten Mitarbeitern zu Verfügung.

21.5.1. Basiskonfiguration

Die Basiskonfiguration wird einmalig bei der Inbetriebnahme des Systems durchgeführt und bleibt im Betrieb weitestgehend unverändert. Eine Änderung darf nur durch ausreichend geschulte Personen durchgeführt werden.

```
1 <?php
2
3     // GLOBAL CONFIGURATION
4     define("_DEBUG", true);
5     define("_LOGFILEPATH", "/path/to/logfile.log");
6
7     // LIBRARY
8     define("_BOOTSTRAP_CSS", "/path/to/bootstrap.css");
9     define("_BOOTSTRAP_JS", "/path/to/bootstrap.js");
10    define("_JQUERY_JS", "/path/to/jquery.js");
11
12    // DATABASE
13    define("_SQL_HOSTNAME", "hostname");
14    define("_SQL_DATABASE", "database");
15    define("_SQL_USERNAME", "username");
16    define("_SQL_PASSWORD", "password");
17
18 ?>
```

Abbildung 124 - Administrationshandbuch-Basiskonfiguration

21.5.2. Dashboard

The screenshot shows the administration section of the PartnerDashboard. At the top, there is a navigation bar with links for Dashboard, Kundenübersicht, Rechnungsübersicht, News, and a user profile for 'Kunde: Sander, Manuel'. Below the navigation bar, the main content area is titled 'Dashboard' and 'PartnerDashboard'. A sidebar on the left contains sections for 'Allgemein' and 'Utilities'. The main content area is titled 'Administration' and includes links for 'Benutzerverwaltung', 'Berechtigungen', 'Requests', 'Requestanfragen', 'News Verwaltung', and 'FAQ Verwaltung'. At the bottom of the page, there is a footer with copyright information: '© 2017 partner.dashboard Impressum AGB-Download'.

Abbildung 125 - Administrationshandbuch-Dashboard

Um die administrativen Funktionen des PartnerDashboards nutzen zu können, steht hinter dem Menüpunkt „Dashboard“ der Bereich „Administration“ zur Verfügung. Dieser ist ausschließlich für Administratoren sichtbar.



21.5.3. Benutzerverwaltung



Abbildung 126 - Administrationshandbuch-Benutzerverwaltung-Logo

	Benutzername	Vorname	Aktionen
J	ADMIN	Zach, Johannes	[Edit]
A	AMANDA.GARRETT	Garrett, Amanda	[Edit]
B	BILLY.RICHARDS	Richards, Billy	[Edit]
C	CATHERINE.MONTGOMERY	Montgomery, Catherine	[Edit]
C	CATHERINE.PRICE	Price, Catherine	[Edit]
D	DEBORAH.THOMAS	Thomas, Deborah	[Edit]
G	GERNOT.KREMS	Krems, Gernot	[Edit]

Abbildung 127 - Administrationshandbuch-Benutzerverwaltung

Das Modul „Benutzerverwaltung“ gilt als Übersicht der vorhandenen Benutzerkonten. Das PartnerDashboard kann von sämtlichen Benutzern genutzt werden. Über die Schaltfläche „Neuen Benutzer anlegen“, können weitere Konten hinzugefügt werden.

Benutzername	Username
Passwort	Password
Anrede	Herr
Vorname	Vorname
Nachname	Nachname
Geburtsdatum	mm/dd/yyyy
Telefon	Telefon
Mobil	Mobil
Mail	Mail
Fax	Fax
Adresse	Adresse
PLZ	PLZ
Ort	Ort
Land	Österreich
VAT	

Abbildung 128 - Administrationshandbuch-Benutzeranlage



Weiters dient das Modul der Verwaltung sowie der Löschung von Benutzerkonten und bietet die Weiterleitung zu den dazugehörigen Kunden.

The screenshot shows a list of users in a table format:

	User	Customer Name
J	ADMIN	Zach, Johannes
A	AMANDA.GARRETT	Garrett, Amanda
B	BILLY.RICHARDS	Richards, Billy

A context menu is open over the first row (ADMIN), containing the following options:

- Konto ändern
- Konto löschen
- Kennwort ändern

At the bottom right of the menu, there is a link "Kunden anzeigen".

Abbildung 129 - Administrationshandbuch-Benutzerverwaltung 2

Zusätzlich steht in der Änderungsansicht die optionale Aktivierung der Two-Factor Authentication zur Verfügung.

The screenshot shows a configuration form for Two-Factor Authentication:

Benutzerinformation	
Benutzername	admin
Passwort	<input type="password"/> <input checked="" type="checkbox"/>
Two-Factor-Authentication	

Abbildung 130 - Administrationshandbuch-TFA

Mittels Klick auf das QR-Icon wird der dazugehörige QR-Code angezeigt, der bequem am Smartphone gescannt werden kann.



Abbildung 131 - Administrationshandbuch-QR-Code



21.5.4. Berechtigungsverwaltung



Abbildung 132 - Administrationshandbuch-Berechtigungen-Logo

The screenshot shows the 'Berechtigungen' (Permissions) section of the PartnerDashboard. At the top, there are navigation links for Dashboard, Kundenübersicht, Rechnungsübersicht, News, and user information (Kunde: Sander, Manuel, Warenkorb: 0, FAQ, Zach, Johannes). Below the header, there are three tabs: Administratoren, Berechtigungen (selected), and Mitglieder. The main content area displays a list of permissions under 'General': Verwalten von News, Verwalten von FAQs, Requests, Requestanfragen bearbeiten, Kennwörter ändern, Benutzerverwaltung, and Zugriffsverwaltung. There are also sections for 'System' and 'Kunden'. At the bottom, there are buttons for '+ Gruppe erstellen', 'Berechtigungen speichern', and 'Benutzer hinzufügen'.

Abbildung 133 - Administrationshandbuch-Berechtigungen

Innerhalb des „Berechtigungsmoduls“ ist es möglich, den Zugriff auf einzelne Bereiche des PartnerDashboards für vorhandene Benutzerkonten zu steuern. Um einen guten Überblick zu gewährleisten, wurden verfügbare Berechtigungen in eigene Kategorien unterteilt. Zur Minimierung des administrativen Aufwands werden diese Berechtigungen ausschließlich für Berechtigungsgruppen vergeben, in denen einzelne Benutzer als Mitglied hinzugefügt werden können.

The screenshot shows the 'Berechtigungen' (Permissions) section of the PartnerDashboard. The 'Berechtigungen' tab is selected. A table lists users assigned to a specific permission group: Sander, Manuel (manuel.sander), Gernot Krems (gernot.krems), and Johannes Zach (johannes.zach). Each row has a delete button ('X') on the right. The footer includes copyright information: © 2017 partner.dashboard Impressum AGB-Download.

Abbildung 134 - Administrationshandbuch-Berechtigungsgruppenmitglieder



21.5.5. Request Engine



Abbildung 135 - Administrationshandbuch-Requests-Logo

Die Request Engine arbeitet autark im Hintergrund und führt in einem festgelegten Intervall definierte Prozesse (Produktbestellung, Rechnungserstellung, ...) aus.

Produktbestellung - WebHosting: Small			
	Zach, Johannes		2017-05-02 12:22:59
InquiryApprovalTask		COMPLETED	
ApprovedTask		COMPLETED	
CreateStorageTask		COMPLETED	
EnablePhpTask		COMPLETED	
EnableMySqlTask		COMPLETED	
EnablePhpmysqladminTask		COMPLETED	
EnableSslTask		COMPLETED	
EnableMailFeaturesTask		COMPLETED	
EnableSecurityFeaturesTask		COMPLETED	
EnableWebmailTask		COMPLETED	
DeclinedTask		SKIPPED	
SendMailTask		SKIPPED	
CancelRequestTask		SKIPPED	
Finalize Order Request	Zach, Johannes	COMPLETED	2017-05-02 12:22:59
Finalize Order Request	Zach, Johannes	COMPLETED	2017-05-02 12:23:00
Produktbestellung - WebHosting: startUp	Zach, Johannes	COMPLETED	2017-05-02 12:23:00

Abbildung 136 - Administrationshandbuch-Requests

Auf der Übersichtsseite kann der Administrator den Status einzelner Requests inklusive der darin enthaltenen Aufgaben einsehen.



21.5.6. Requestanfragen



Abbildung 137 - Administrationshandbuch-Requestanfragen-Logo

Während der Abarbeitung von Requests gibt es für spezielle Aufgaben die Möglichkeit Genehmigungen eines Administrators anzufordern. Abhängig von der Entscheidung des Administrators werden die Aufgaben innerhalb des Requests durchgeführt.

Requester	Product Type	Date	Action Buttons
Zach, Johannes für Sander, Manuel	Produktbestellung - WebHosting: startUp	2017-05-02 12:47:17	Ablehnen Genehmigen
Zach, Johannes für Krems, Gernot	Produktbestellung - WebHosting: Small	2017-05-02 12:47:21	Ablehnen Genehmigen

Abbildung 138 - Administrationshandbuch-Request Inquiry



21.5.7. News Verwaltung



Abbildung 139 - Administrationshandbuch-News Verwaltung-Logo

The screenshot shows a web-based administration interface. At the top, there is a navigation bar with links for Dashboard, Kundenübersicht, Rechnungsübersicht, News, Kunde: Sander, Manuel, Warenkorb: 0, FAQ, and Zach, Johannes. Below the navigation bar, the main content area has a title "News Verwaltung" and a subtitle "PartnerDashboard / News Verwaltung". A table lists four news items with columns for Title, Start Date, End Date, and Actions. To the right of the table, a dropdown menu titled "Filter: Alle" is open, showing options: Alle, Sichtbar (selected), and Nicht sichtbar. At the bottom of the page, there is a footer with the text "© 2017 partner.dashboard Impressum AGB-Download".

Abbildung 140 - Administrationshandbuch-News

Im Modul „News Verwaltung“ können Informationen zu Ereignissen, welche für Partner interessant sind, verfasst werden. Die Informationen können mit einem Start- sowie Enddatum versehen werden, so dass diese nur in einem definierten Zeitraum sichtbar sind. Das Statusicon neben dem Titel zeigt an, ob die Information im Moment dargestellt wird. Um eine gute Übersicht gewährleisten zu können, steht im Aktionsmenü eine Filterfunktion zur Verfügung.

Durch Klick auf das Änderungsicon öffnet sich ein Bearbeitungsfenster. Es kann der Titel, die Start- sowie Endzeiten festgelegt werden und für den Inhalt steht ein Richtext-Editor zur Verfügung.

The screenshot shows a modal dialog box titled "News Eintrag editieren". It contains a form with fields for "Titel" (Title) set to "Title#01", "VON" (From) set to "2017-01-31", and "BIS" (To) set to "2018-01-18". Below these fields is a rich text editor toolbar. The main content area contains the text "Das ist eine Testnachricht !". At the bottom of the dialog, there are two buttons: "SPEICHERN" (Save) and "SCHLIESSEN" (Close).

Abbildung 141 - Administrationshandbuch-Newseinträge bearbeiten



21.5.8.FAQ Verwaltung



Abbildung 142 - Administrationshandbuch-FAQ Verwaltung-Logo

Im Modul „FAQ Verwaltung“ können wiederkehrende Fragen und dazugehörige Antworten bzw. Lösungen verfasst werden.

The screenshot shows a web-based administration interface. At the top, there's a navigation bar with links for Dashboard, Kundenübersicht, Rechnungsübersicht, News, and a user profile section (Kunde: Sander, Manuel, Warenkorb: 0, FAQ, Zach, Johannes). Below the navigation is a header for 'FAQ Verwaltung' with a sub-header 'PartnerDashboard / FAQ Verwaltung'. A toolbar at the top right includes 'Filter: Allgemein', 'Kategorien verwalten', and 'Neuen Eintrag erstellen'. The main content area displays a table of four entries, each with a category dropdown, a question, and edit/delete icons. The questions are: 'Wie kann ich mein Kennwort ändern?', 'Wie kann ich die Two-Factor-Authentication aktivieren?', 'Wie kann ich einen neuen Kunden anlegen?', and 'Wie kann ich Produkte bestellen?'. At the bottom of the page is a footer with copyright information: '© 2017 partner.dashboard Impressum AGB-Download'.

Abbildung 143 - Administrationshandbuch-FAQ

FAQs können in Kategorien unterteilt werden, welche über das Aktionsmenü durch einen Klick auf „Kategorien verwalten“ bearbeitet werden können.

This screenshot shows a modal window titled 'FAQ Kategorien'. It lists six categories: Allgemein, Kundenverwaltung, Account, Payments, Privacy, and Delivery. Each category has an edit icon (pencil) and a delete icon (cross) next to it. At the bottom of the modal are two buttons: a plus sign icon for adding new categories and a 'SCHLIESSEN' (close) button.

Abbildung 144 - Administrationshandbuch-FAQ Kategorien

Kategorien, welche in Verwendung sind, können geändert, jedoch nicht gelöscht werden.



Allgemein	Wie kann ich mein Kennwort ändern?	<input checked="" type="checkbox"/> <input type="checkbox"/>
Nicht zugeordnet	Wie kann ich die Two-Factor-Authentication aktivieren?	<input checked="" type="checkbox"/> <input type="checkbox"/>
Allgemein Kundenverwaltung Account Payments Privacy Delivery	Wie kann ich einen neuen Kunden anlegen?	<input checked="" type="checkbox"/> <input type="checkbox"/>
ung	Wie kann ich Produkte bestellen?	<input checked="" type="checkbox"/> <input type="checkbox"/>
	Wie kann ich einen Kunden auswählen?	<input checked="" type="checkbox"/> <input type="checkbox"/>

Abbildung 145 - Administrationshandbuch-FAQ Kategorie Auswahl

Die zur Verfügung stehenden Kategorien, können direkt im FAQ Eintrag durch ein DropDown-Element ausgewählt werden. Einträge, die der Kategorie „Nicht zugeordnet“ zu finden sind, werden im PartnerDashboard nicht angezeigt.

Um eine gute Übersicht gewährleisten zu können, steht eine Filterfunktion zur Verfügung.

FAQ Verwaltung		Filter: Allgemein	Kategorien verwalten	Neuen Eintrag erstellen
Allgemein	Wie kann ich mein Kennwort ändern?	<input checked="" type="checkbox"/> <input type="checkbox"/>	Alle	<input checked="" type="checkbox"/> <input type="checkbox"/>
Allgemein	Wie kann ich die Two-Factor-Authentication aktivieren?	<input checked="" type="checkbox"/> <input type="checkbox"/>	Nicht zugeordnet	<input checked="" type="checkbox"/> <input type="checkbox"/>
Allgemein	Wie kann ich einen neuen Kunden anlegen?	<input checked="" type="checkbox"/> <input type="checkbox"/>	Allgemein	<input checked="" type="checkbox"/> <input type="checkbox"/>
Allgemein	Wie kann ich Produkte bestellen?	<input checked="" type="checkbox"/> <input type="checkbox"/>	Kundenverwaltung	<input checked="" type="checkbox"/> <input type="checkbox"/>
	Wie kann ich einen Kunden auswählen?	<input checked="" type="checkbox"/> <input type="checkbox"/>	Account	<input checked="" type="checkbox"/> <input type="checkbox"/>
			Payments	<input checked="" type="checkbox"/> <input type="checkbox"/>
			Privacy	<input checked="" type="checkbox"/> <input type="checkbox"/>
			Delivery	<input checked="" type="checkbox"/> <input type="checkbox"/>

Abbildung 146 - Administrationshandbuch-FAQ Filter



PartnerDashboard

Stundennachweis





21.6. Stundennachweis

21.6.1. Gernot Krems

21.6.1.1. Gesamtübersicht

Monat	Zuhause	Team	Ist	Soll	Delta	Ist PTD	Soll PTD	Abw PTD
September	5,5	22,0	27,5	26	1,5	27,5	26	1,5
Oktober	11,5	18,0	29,5	26	3,5	57,0	52	5,0
November	13,0	9,0	22,0	26	-4,0	79,0	78	1,0
Dezember	14,5	8,5	23,0	26	-3,0	102,0	104	-2,0
Januar	27,5	1,5	29,0	26	3,0	131,0	130	1,0
Februar	21,5	12,5	34,0	26	8,0	165,0	156	9,0
März	28,5	3,5	32,0	26	6,0	197,0	182	15,0
Summe	122,0	75,0	197,0	182	15,0			

Tabelle 7 - Stundennachweis-Gernot Krems-Übersicht

21.6.1.2. PTD

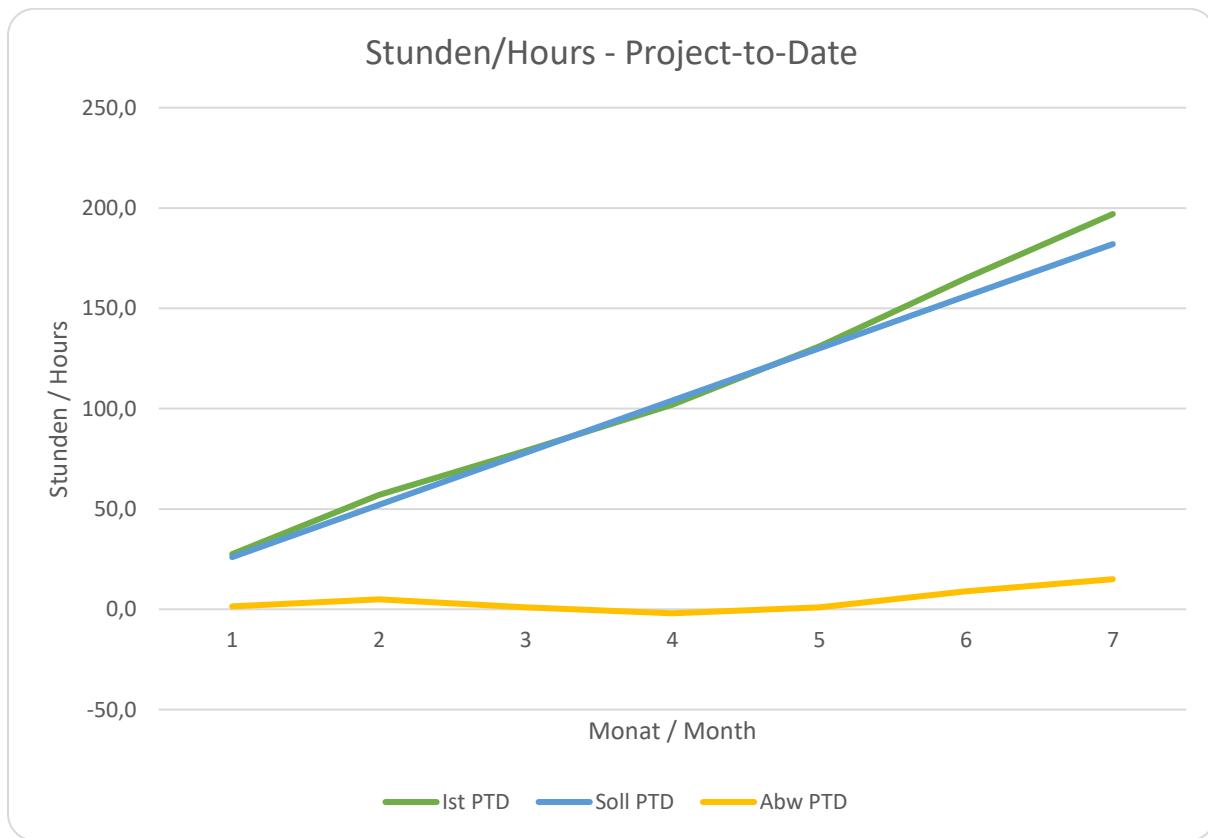


Tabelle 8 - Stundennachweis-Gernot Krems-PTD



21.6.1.3. Stunden pro Monat

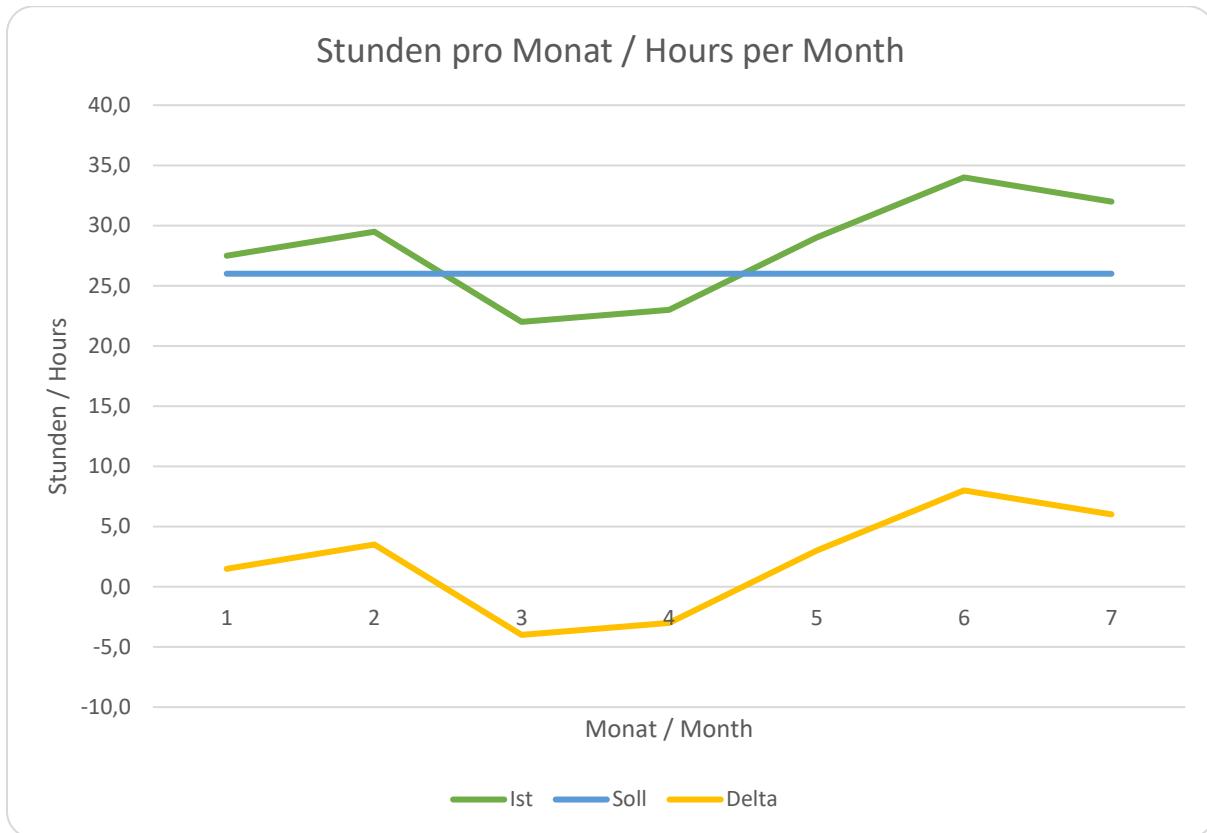


Tabelle 9 - Stundennachweis-Gernot Krems-Stunden pro Monat



21.6.2. Manuel Sander

21.6.2.1. Gesamtübersicht

Monat	Zuhause	Team	Ist	Soll	Delta	Ist PTD	Soll PTD	Abw PTD
September	9,0	22,0	31,0	26	5,0	31,0	26	5,0
Oktober	12,5	18,0	30,5	26	4,5	61,5	52	9,5
November	18,5	9,0	27,5	26	1,5	89,0	78	11,0
Dezember	21,5	8,5	30,0	26	4,0	119,0	104	15,0
Januar	33,0	1,5	34,5	26	8,5	153,5	130	23,5
Februar	22,0	12,5	34,5	26	8,5	188,0	156	32,0
März	22,0	3,5	25,5	26	-0,5	213,5	182	31,5
Summe	138,5	75,0	213,5	182	31,5			

Tabelle 10 - Stundennachweis-Manuel Sander-Übersicht

21.6.2.2. PTD

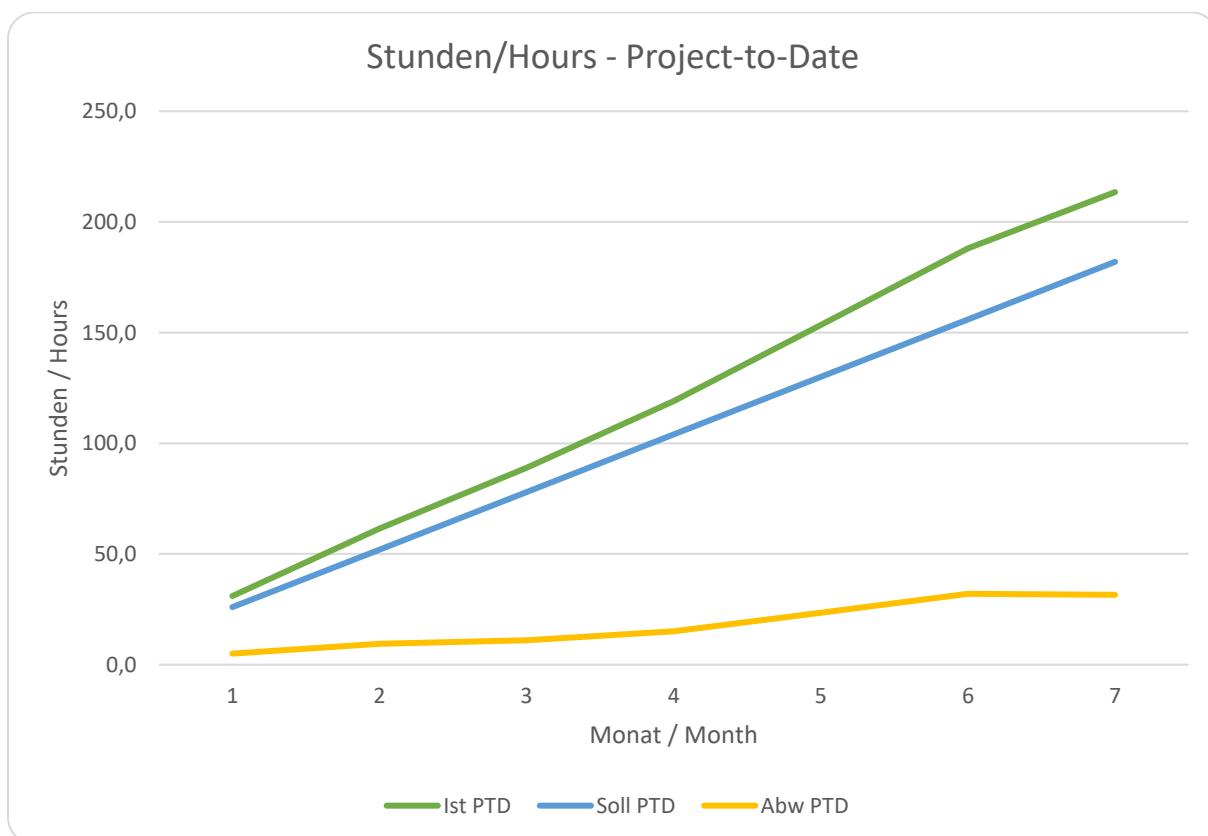


Tabelle 11 - Stundennachweis-Manuel Sander-PTD



21.6.2.3. Stunden pro Monat

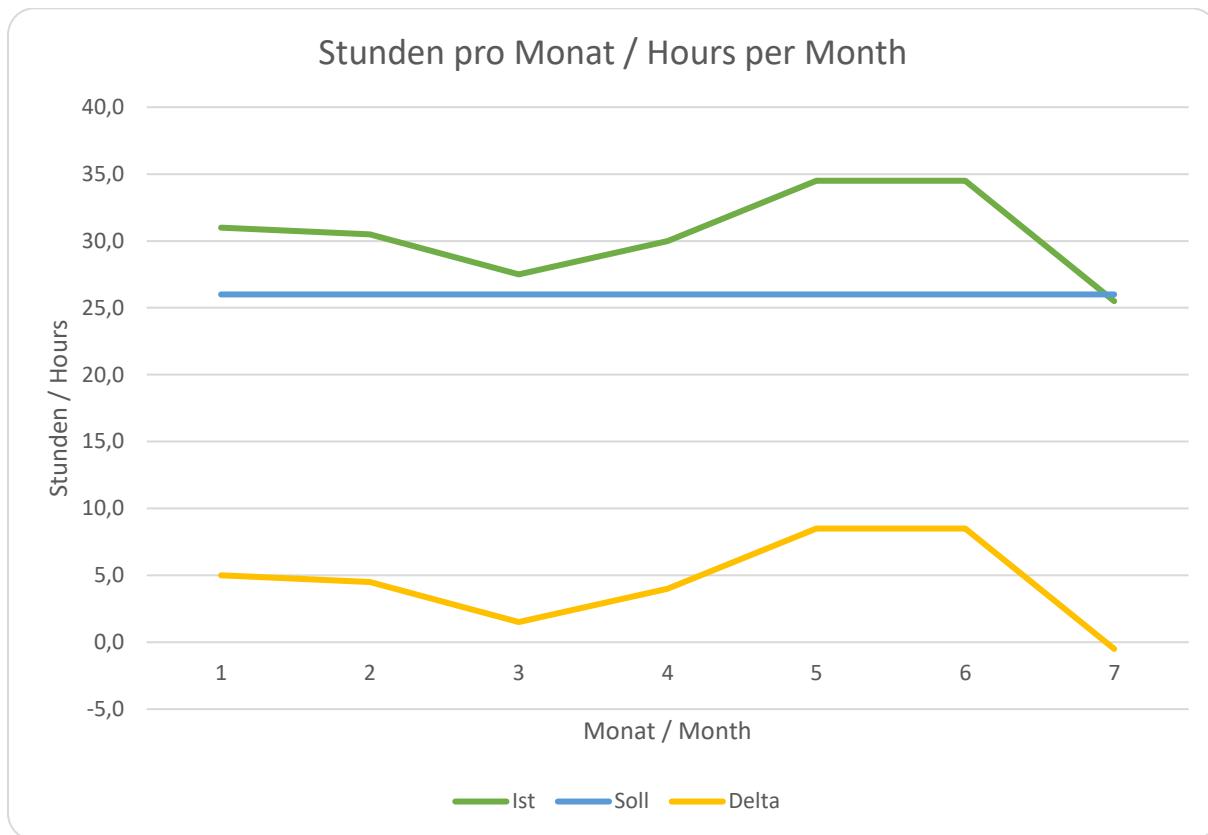


Tabelle 12 - Stundennachweis-Manuel Sander-Stunden pro Monat



21.6.3.Johannes Zach

21.6.3.1. Gesamtübersicht

Monat	Zuhause	Team	Ist	Soll	Delta	Ist PTD	Soll PTD	Abw PTD
September	15,0	22,0	37,0	26	11,0	37,0	26	11,0
Oktober	16,0	18,0	34,0	26	8,0	71,0	52	19,0
November	16,0	9,0	25,0	26	-1,0	96,0	78	18,0
Dezember	22,5	8,5	31,0	26	5,0	127,0	104	23,0
Jänner	33,0	1,5	34,5	26	8,5	161,5	130	31,5
Februar	21,0	12,5	33,5	26	7,5	195,0	156	39,0
März	22,5	3,5	26,0	26	0,0	221,0	182	39,0
Summe	146,0	75,0	221,0	182	39,0			

Tabelle 13 - Stundennachweis-Johannes Zach-Übersicht

21.6.3.2. PTD

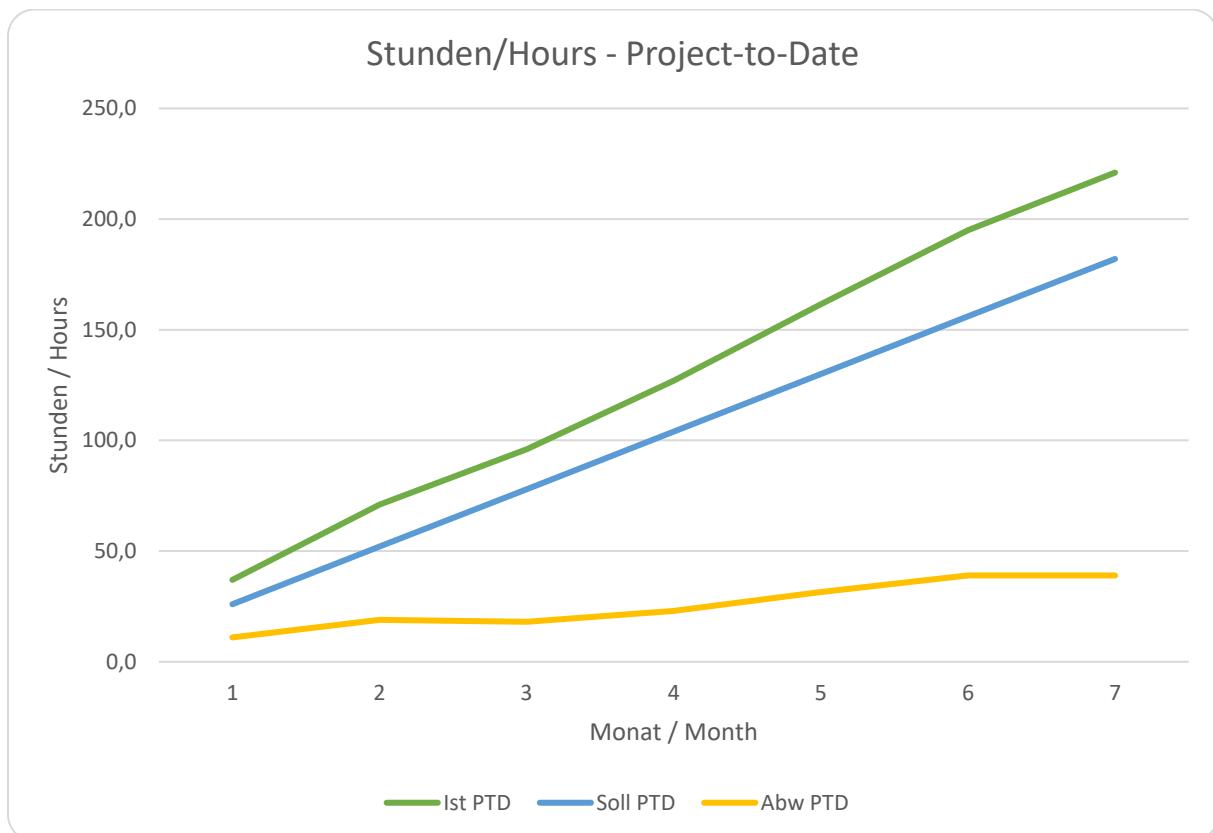


Tabelle 14 - Stundennachweis-Johannes Zach-PTD



21.6.3.3. Stunden pro Monat

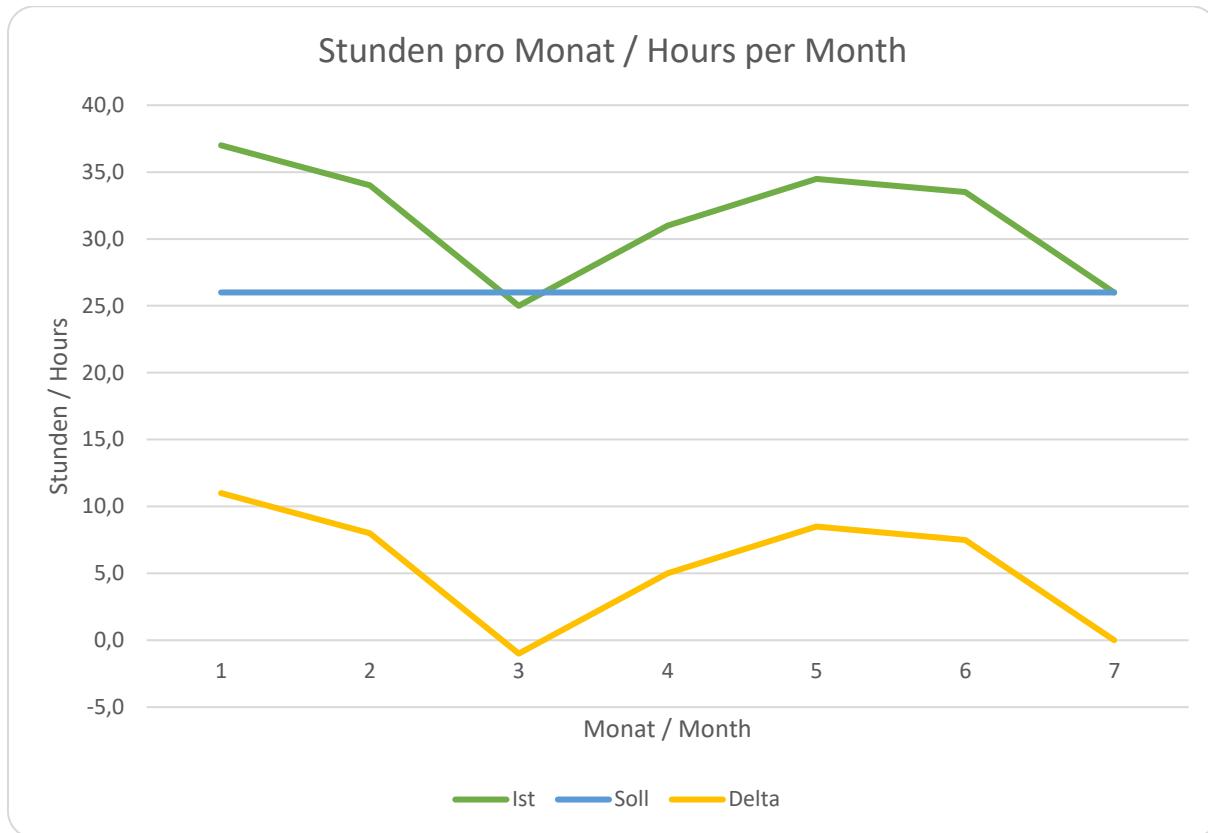


Tabelle 15 - Stundennachweis-Johannes Zach-Stunden pro Monat



PartnerDashboard

Sprintstatusberichte





21.7. Sprintstatusberichte

21.7.1. Sprint 1

1. Projektstammdaten			
Projektbezeichnung	PartnerDashboard	Bericht Nummer	1
Projektaufzeit	01.09.2016 – 05.04.2017	Bericht Zeitraum	01.09.2016 – 01.10.2016
Projektleiter: Manuel Sander			

Tabelle 16 - Sprintstatusbericht-Sprint 1.1

2. Gesamteinschätzung des Projekts aus Sicht des Teams		
Zielerreichung		Wir haben leider nicht alle Ziele erreicht, die wir uns vorgenommen haben.
Zeitplan		Konnte leider nicht komplett eingehalten werden.
Ressourcenstatus		Aktuell sind die Ressourcen gedeckt.
Rahmenbedingungen		Die Datenbankschnittstelle ist noch nicht vollständig besprochen.

Tabelle 17 - Sprintstatusbericht-Sprint 1.2

3. Retrospektive	
Was haben wir gut gemacht?	Wir haben eine gute Übersicht über das Ausmaß des Projektes gewonnen. Auch haben wir eine gute Struktur aufgebaut.
Wobei müssen wir uns noch verbessern?	Vor allem müssen wir die Arbeitsverteilung überdenken und konsequenter durchführen.
Welche Bereiche müssen überhaupt neu überdacht werden?	Keine

Tabelle 18 - Sprintstatusbericht-Sprint 1.3

4. Resümee des Teams	
Nach anfänglichen Koordinationsproblemen verlief der Sprint durchaus positiv. Wir sollten jedoch noch einige Dinge (Kommunikation, Arbeitsverteilung usw.) verbessern.	

Tabelle 19 - Sprintstatusbericht-Sprint 1.4



5. Kennzahlen

	IST / Bewertung			Erläuterungen / Hinweise	Maßnahmen / weiteres Vorgehen
	09/16				
Wurden die Sprintziele erreicht?				Nicht wie erwartet	Realistischere Sprintplanung
Wurden die Prozessschritte eingehalten?				Ja	
Wurden die Codierstandards eingehalten?				Ja, nach PHP-FIG	
Fehlen Standards/Vorgaben?				Bis jetzt nicht	
Unterstützt die verwendete Technologie noch die Anforderungen im Projekt?				Ja	
Gibt es zwischenmenschliche Reibungen?				Unterschiedliche Interpretationen von Arbeit	Aussprache
Ist die Kommunikation ausreichend?				Meetings werden eingehalten. Wir kommen öfters vom Thema ab.	Nach Lösungen wird gesucht

Tabelle 20 - Sprintstatusbericht-Sprint 1.5

6. Screenshot des Ergebnisses des Sprints

The screenshot shows a Jira board with the following details:

- PAR Sprint 1**: The sprint name at the top left.
- QUICK FILTERS: Only My Issues Recently Updated**: Filter options at the top right.
- Board View**: A button to switch between board and list view.
- To Do**: An empty column.
- In Progress**: Contains two items:
 - PAR-18**: Status: (checked), Description: Login, Assigned to: JZ
 - PAR-17**: Status: (checked), Description: DAO's, Assigned to: JZ
- Done**: Contains four items:
 - PAR-14**: Status: (checked), Description: Mockups fertigstellen, Assigned to: MS
 - PAR-15**: Status: (checked), Description: Datenbank fertigstellen, Assigned to: JZ
 - PAR-20**: Status: (checked), Description: Factorys erstellen, Assigned to: MS
 - PAR-16**: Status: (checked), Description: Value Objects, Assigned to: MS

Abbildung 147 - Sprintstatusbericht-Ergebnis

Tabelle 21 - Sprintstatusbericht-Sprint 1.6



21.7.2.Sprint 2

1. Projektstammdaten			
Projektbezeichnung	PartnerDashboard	Bericht Nummer	2
Projektaufzeit	01.09.2016 – 05.04.2017	Bericht Zeitraum	01.10.2016 – 01.11.2016
Projektleiter: Manuel Sander			

Tabelle 22 - Sprintstatusbericht-Sprint 2.1

2. Gesamteinschätzung des Projekts aus Sicht des Teams		
Zielerreichung		Alle gesetzten Ziele wurden erreicht.
Zeitplan		Wurde zur Gänze eingehalten.
Ressourcenstatus		Aktuell sind die Ressourcen gedeckt.
Rahmenbedingungen		Die Datenbankschnittstelle ist noch nicht vollständig besprochen.

Tabelle 23 - Sprintstatusbericht-Sprint 2.2

3. Retrospektive	
Was haben wir gut gemacht?	Aufgrund des Zeitmangels wurden die Ziele bewusst niedriger gesetzt.
Wobei müssen wir uns noch verbessern?	Zeitmanagement, Entscheidungsfähigkeit
Welche Bereiche müssen überhaupt neu überdacht werden?	SQL Anbindung (MySQLi vs. PDO)

Tabelle 24 - Sprintstatusbericht-Sprint 2.3

4. Resümee des Teams
In diesem Sprint wurden die Ziele aufgrund von Zeitmangel bewusst niedriger gesetzt. Wir wollten realisierbare Ziele definieren, die wir auch erreichen konnten.

Tabelle 25 - Sprintstatusbericht-Sprint 2.4



5. Kennzahlen

	IST / Bewertung			Erläuterungen / Hinweise	Maßnahmen / weiteres Vorgehen
	10/16	09/16			
Wurden die Sprintziele erreicht?				Ja	
Wurden die Prozessschritte eingehalten?				Ja	
Werden die Codierstandards eingehalten?				Ja, nach PHP-FIG	
Fehlen Standards/Vorgaben?				Bis jetzt nicht	
Unterstützt die verwendete Technologie noch die Anforderungen im Projekt?				Ja	
Gibt es zwischenmenschliche Reibungen?				Unterschiedliche Interpretationen von Arbeit	Aussprache
Ist die Kommunikation ausreichend?				Kommunikationskanäle sind ausreichend vorhanden und werden genutzt.	

Tabelle 26 - Sprintstatusbericht-Sprint 2.5

6. Screenshot des Ergebnisses des Sprints

Abbildung 148 - Sprintstatusbericht-Ergebnis

Tabelle 27 - Sprintstatusbericht-Sprint 2.6



21.7.3.Sprint 3

1. Projektstammdaten			
Projektbezeichnung	PartnerDashboard	Bericht Nummer	3
Projektaufzeit	01.09.2016 – 05.04.2017	Bericht Zeitraum	01.11.2016 – 01.12.2016
Projektleiter: Manuel Sander			

Tabelle 28 - Sprintstatusbericht-Sprint 3.1

2. Gesamteinschätzung des Projekts aus Sicht des Teams		
Zielerreichung		Alle gesetzten Ziele wurden erreicht.
Zeitplan		Wurde zur Gänze eingehalten.
Ressourcenstatus		Aktuell sind die Ressourcen gedeckt.
Rahmenbedingungen		Die Datenbankschnittstelle ist noch nicht vollständig besprochen.

Tabelle 29 - Sprintstatusbericht-Sprint 3.2

3. Retrospektive	
Was haben wir gut gemacht?	Programmstrukturen wurden eingehalten.
Wobei müssen wir uns noch verbessern?	Zeitmanagement
Welche Bereiche müssen überhaupt neu überdacht werden?	

Tabelle 30 - Sprintstatusbericht-Sprint 3.3

4. Resümee des Teams
Es sind leider noch immer keine visuellen Ergebnisse vorhanden. Dies wird jedoch im nächsten Sprint geschehen. Dieser Aspekt ist für das Team derzeit nicht sehr motivierend.

Tabelle 31 - Sprintstatusbericht-Sprint 3.4



5. Kennzahlen

	IST / Bewertung			Erläuterungen / Hinweise	Maßnahmen / weiteres Vorgehen
	11/16	10/16	09/16		
Wurden die Sprintziele erreicht?				Ja	
Wurden die Prozessschritte eingehalten?				Ja	
Werden die Codierstandards eingehalten?				Ja, nach PHP-FIG	
Fehlen Standards/Vorgaben?				Ajax Dispatcher	
Unterstützt die verwendete Technologie noch die Anforderungen im Projekt?				Ja	
Gibt es zwischenmenschliche Reibungen?				Nein	
Ist die Kommunikation ausreichend?				Kommunikationskanäle sind ausreichend vorhanden und werden genutzt.	

Tabelle 32 - Sprintstatusbericht-Sprint 3.5

6. Screenshot des Ergebnisses des Sprints

PAR Sprint 3

0 days remaining Complete Sprint Board ▾

QUICK FILTERS: Only My Issues Recently Updated

To Do	In Progress	Done
✓ PAR-25 [DONE] 6 sub-tasks Grundstruktur		<ul style="list-style-type: none"> PAR-26 Value Objects MS PAR-27 Factorys MS PAR-29 Interfaces MS PAR-30 Proxys GK PAR-31 DAOs JZ PAR-34 AJAX dispatcher JZ
▼ Other Issues 1 issue		<ul style="list-style-type: none"> PAR-3 Log-Funktion JZ

Abbildung 149 - Sprintstatusbericht-Übersicht

Tabelle 33 - Sprintstatusbericht-Sprint 3.6



21.7.4. Sprint 4

1. Projektstammdaten			
Projektbezeichnung	PartnerDashboard	Bericht Nummer	4
Projektaufzeit	01.09.2016 – 05.04.2017	Bericht Zeitraum	01.12.2016 – 01.01.2017
Projektleiter: Manuel Sander			

Tabelle 34 - Sprintstatusbericht-Sprint 4.1

2. Gesamteinschätzung des Projekts aus Sicht des Teams		
Zielerreichung		Alle gesetzten Ziele wurden erreicht.
Zeitplan		Wurde zur Gänze eingehalten.
Ressourcenstatus		Aktuell sind die Ressourcen gedeckt.
Rahmenbedingungen		Die Datenbankschnittstelle ist noch nicht vollständig besprochen.

Tabelle 35 - Sprintstatusbericht-Sprint 4.2

3. Retrospektive	
Was haben wir gut gemacht?	Programmstrukturen wurden eingehalten.
Wobei müssen wir uns noch verbessern?	Arbeitsaufteilung
Welche Bereiche müssen überhaupt neu überdacht werden?	

Tabelle 36 - Sprintstatusbericht-Sprint 4.3

4. Resümee des Teams
Durch die ersten visuellen Ergebnisse hatten die meisten von uns einen Motivationsschub, da wir erste Erfolgserlebnisse verzeichnen durften.

Tabelle 37 - Sprintstatusbericht-Sprint 4.4



5. Kennzahlen

	IST / Bewertung			Erläuterungen / Hinweise	Maßnahmen / weiteres Vorgehen
	12/16	11/16	10/16		
Wurden die Sprintziele erreicht?				Ja	
Wurden die Prozessschritte eingehalten?				Ja	
Werden die Codierstandards eingehalten?				Ja, nach PHP-FIG	
Fehlen Standards/Vorgaben?				Nein	
Unterstützt die verwendete Technologie noch die Anforderungen im Projekt?				Ja	
Gibt es zwischenmenschliche Reibungen?				Nein	
Ist die Kommunikation ausreichend?				Kommunikationskanäle sind ausreichend vorhanden und werden genutzt.	

Tabelle 38 - Sprintstatusbericht-Sprint 4.5

6. Screenshot des Ergebnisses des Sprints

PAR Sprint 4

0 days remaining Complete Sprint Board ▾

QUICK FILTERS: Only My Issues Recently Updated

To Do	In Progress	Done
▼ <input checked="" type="checkbox"/> PAR-18 IN PROGRESS 1 sub-task Login	<input checked="" type="checkbox"/> PAR-22 ↗ TestSubtask	
▼ Other Issues 4 issues	<input checked="" type="checkbox"/> PAR-3 ↗ Log-Funktion	<input checked="" type="checkbox"/> PAR-17 ↗ DAO's UMBENENNEN
	<input checked="" type="checkbox"/> PAR-32 ↗ Tests	<input checked="" type="checkbox"/> PAR-33 ↗ Frontend Basis

Abbildung 150 - Sprintstatusbericht-Übersicht

Tabelle 39 - Sprintstatusbericht-Sprint 4.6



21.7.5.Sprint 5

1. Projektstammdaten			
Projektbezeichnung	PartnerDashboard	Bericht Nummer	5
Projektaufzeit	01.09.2016 – 05.04.2017	Bericht Zeitraum	01.01.2017 – 01.02.2017
Projektleiter: Manuel Sander			

Tabelle 40 - Sprintstatusbericht-Sprint 5.1

2. Gesamteinschätzung des Projekts aus Sicht des Teams		
Zielerreichung		Es wurden nicht alle gesetzten Ziele erreicht
Zeitplan		Wurde zur nicht Gänze eingehalten.
Ressourcenstatus		Aktuell sind die Ressourcen gedeckt.
Rahmenbedingungen		Die Datenbankschnittstelle ist noch nicht vollständig besprochen.

Tabelle 41 - Sprintstatusbericht-Sprint 5.2

3. Retrospektive	
Was haben wir gut gemacht?	Programmstrukturen wurden eingehalten.
Wobei müssen wir uns noch verbessern?	Arbeitsaufteilung
Welche Bereiche müssen überhaupt neu überdacht werden?	

Tabelle 42 - Sprintstatusbericht-Sprint 5.3

4. Resümee des Teams	
Der Aufbau der grafischen Oberfläche nimmt mehr Zeit in Anspruch als geplant. Uns kommt es bei der Umsetzung des Front Ends enorm zu Gute, dass wir die Grundstruktur nach bekannten Patterns aufgebaut haben.	

Tabelle 43 - Sprintstatusbericht-Sprint 5.4



5. Kennzahlen

	IST / Bewertung			Erläuterungen / Hinweise	Maßnahmen / weiteres Vorgehen
	01/17	12/16	11/16		
Wurden die Sprintziele erreicht?				Nein	
Wurden die Prozessschritte eingehalten?				Ja	
Werden die Codierstandards eingehalten?				Ja, nach PHP-FIG	
Fehlen Standards/Vorgaben?				Nein	
Unterstützt die verwendete Technologie noch die Anforderungen im Projekt?				Ja	
Gibt es zwischenmenschliche Reibungen?				Nein	
Ist die Kommunikation ausreichend?				Kommunikationskanäle sind ausreichend vorhanden und werden genutzt.	

Tabelle 44 - Sprintstatusbericht-Sprint 5.5

6. Screenshot des Ergebnisses des Sprints

PAR Sprint 5

QUICK FILTERS: Only My Issues Recently Updated

To Do	In Progress	Done
	<ul style="list-style-type: none"> PAR-11 FAQ's PAR-3 Log-Funktion PAR-5 Responsive Design PAR-32 Tests 	<ul style="list-style-type: none"> PAR-4 Two-Factor Authentication PAR-34 Benutzer-/Berechtigungsmanagement PAR-7 Kunden anlegen und verwalten PAR-9 Übersicht über Kunden

Abbildung 151 - Sprintstatusbericht-Übersicht

Tabelle 45 - Sprintstatusbericht-Sprint 5.6



21.7.6.Sprint 6

1. Projektstammdaten			
Projektbezeichnung	PartnerDashboard	Bericht Nummer	6
Projektaufzeit	01.09.2016 – 05.04.2017	Bericht Zeitraum	01.02.2017 – 01.03.2017
Projektleiter: Manuel Sander			

Tabelle 46 - Sprintstatusbericht-Sprint 6.1

2. Gesamteinschätzung des Projekts aus Sicht des Teams		
Zielerreichung		Es wurden nicht alle gesetzten Ziele erreicht.
Zeitplan		Wurde zur nicht Gänze eingehalten.
Ressourcenstatus		Aktuell sind die Ressourcen gedeckt.
Rahmenbedingungen		Rahmenbedingungen sind geklärt.

Tabelle 47 - Sprintstatusbericht-Sprint 6.2

3. Retrospektive	
Was haben wir gut gemacht?	Frontend-Design wurde zufriedenstellend umgesetzt.
Wobei müssen wir uns noch verbessern?	Konsequenzen der Änderungen besser abschätzen können.
Welche Bereiche müssen überhaupt neu überdacht werden?	Keine

Tabelle 48 - Sprintstatusbericht-Sprint 6.3

4. Resümee des Teams
Dieser Sprint war aufgrund des enormen Drucks der Schule sowie des Berufs überaus anstrengend. Dadurch entstanden mehr und mehr Reibungen im Team und die Kommunikation war nur noch unregelmäßig möglich.

Tabelle 49 - Sprintstatusbericht-Sprint 6.4



5. Kennzahlen

	IST / Bewertung			Erläuterungen / Hinweise	Maßnahmen / weiteres Vorgehen
	02/17	01/17	12/16		
Wurden die Sprintziele erreicht?				Es konnten nicht alle Sprintziele erledigt werden	Bessere Planung
Wurden die Prozessschritte eingehalten?					
Werden die Codierstandards eingehalten?				Ja, nach PHP-FIG	
Fehlen Standards/Vorgaben?					
Unterstützt die verwendete Technologie noch die Anforderungen im Projekt?					
Gibt es zwischenmenschliche Reibungen?				Energiereserven gehen zu Ende	
Ist die Kommunikation ausreichend?				Aufgrund von Schulstress wurde die Kommunikation vernachlässigt	

Tabelle 50 - Sprintstatusbericht-Sprint 6.5

6. Screenshot des Ergebnisses des Sprints

The screenshot shows a Jira sprint status report for "PAR Sprint 6". The top navigation bar includes "QUICK FILTERS: Only My Issues Recently Updated", a timer indicating "0 days remaining Complete Sprint", and a "Board" button. The main board area has three columns: "To Do", "In Progress", and "Done".

- To Do:** Contains one issue: PAR-1 TO DO 2 sub-tasks Informationen schnell weitergeben. A "Move to Done" button is located at the bottom right of this column.
- In Progress:** Contains one issue: PAR-13 Provisionsauswertung.
- Done:** Contains six issues:
 - PAR-3 Log-Funktion
 - PAR-5 Responsive Design
 - PAR-4 FAQ's
 - PAR-8 Benachrichtigungen
 - PAR-10 Bestellungen abwickeln
 - PAR-12 Übersicht über Aktionen und Produkte
 - PAR-6 Kommunikationsschnittstelle

A sidebar on the left lists "Other Issues 8 issues" with icons for each:

- PAR-13 (grey)
- PAR-5 (orange)
- PAR-4 (blue)
- PAR-8 (green)
- PAR-10 (yellow)
- PAR-12 (purple)
- PAR-6 (red)

Abbildung 152 Sprintstatusbericht Übersicht

Tabelle 51 - Sprintstatusbericht-Sprint 6.6



21.7.7.Sprint 7

1. Projektstammdaten			
Projektbezeichnung	PartnerDashboard	Bericht Nummer	7
Projektaufzeit	01.09.2016 – 05.04.2017	Bericht Zeitraum	01.03.2017 – 24.03.2017
Projektleiter: Manuel Sander			

Tabelle 52 - Sprintstatusbericht-Sprint 7.1

2. Gesamteinschätzung des Projekts aus Sicht des Teams		
Zielerreichung		Sämtliche Ziele wurden erreicht.
Zeitplan		Der Zeitplan wurde zur Gänze eingehalten.
Ressourcenstatus		Aktuell sind die Ressourcen gedeckt.
Rahmenbedingungen		Alle Rahmenbedingungen sind geklärt.

Tabelle 53 - Sprintstatusbericht-Sprint 7.2

3. Retrospektive	
Was haben wir gut gemacht?	Tests und die letzten Teile des Frontends wurden erfolgreich umgesetzt.
Wobei müssen wir uns noch verbessern?	Fokus auf das Wesentliche setzen.
Welche Bereiche müssen überhaupt neu überdacht werden?	Keine

Tabelle 54 - Sprintstatusbericht-Sprint 7.3

4. Resümee des Teams
Wir haben bemerkt, dass wir uns mehr auf das Wesentliche fokussieren müssen und uns nicht mit Details aufhalten dürfen. Der dadurch unnötig entstandene Zeitdruck schuf zusätzliche Reibungsflächen im Team.

Tabelle 55 - Sprintstatusbericht-Sprint 7.4



5. Kennzahlen

	IST / Bewertung			Erläuterungen / Hinweise	Maßnahmen / weiteres Vorgehen
	03/17	02/17	01/17		
Wurden die Sprintziele erreicht?					
Wurden die Prozessschritte eingehalten?					
Werden die Codierstandards eingehalten?				Ja, nach PHP-FIG	
Fehlen Standards/Vorgaben?					
Unterstützt die verwendete Technologie noch die Anforderungen im Projekt?					
Gibt es zwischenmenschliche Reibungen?				Interne Abhängigkeiten erschweren die Zusammenarbeit.	
Ist die Kommunikation ausreichend?					

Tabelle 56 - Sprintstatusbericht-Sprint 7.5

6. Screenshot des Ergebnisses des Sprints

Abbildung 153 - Sprintstatusbericht-Übersicht

Tabelle 57 - Sprintstatusbericht-Sprint 7.6



PartnerDashboard

Meeting Protokolle





21.8. Meeting Protokolle

Das Kickoff-Meeting und das Meeting für die Schnittstelle der Datenbank waren die wichtigsten Meetings für den Erfolg des PartnerDashboards.

Die Abkürzung A steht für Aufgabe, B für Beschluss, F für Feststellung und I für Informativ.

21.8.1. Kickoff-Meeting

Meeting Minutes

An	Organisator	Vertraulichkeit
Teilnehmer und informierte Personen	Sander Manuel	Öffentlich
Autor		Erstellungsdatum
Sander Manuel		05. August 2016
Datum	Uhrzeit	Ort
04. August 2016	17:00 – 21:00	Graz

Tabelle 58 - Meeting Protokolle-Kickoff Meeting-1

Teilnehmer	Informierte Personen
Hr. Krems Gernot	Hr. Hochwarter Bernd
Hr. Biernat Dieter	Hr. Zach Johannes
Hr. Sander Manuel	

Tabelle 59 - Meeting Protokolle-Kickoff Meeting-2



21.8.1.1. Thema 1 - Projektantrag

Art	Beschreibung	Verantwortlich
I	Robot Funktion hinzugefügt	Sander Manuel
I	Für gut befunden	Sander Manuel

Tabelle 60 - Meeting Protokolle-Kickoff Meeting-3

21.8.1.2. Thema 2 - Vorstudie

Art	Beschreibung	Verantwortlich
A	Vorabsprachen mit Auftraggeber: Probleme in den derzeitigen Prozessen beschreiben.	Biernat Dieter
A	Soll Prozesse fertigstellen	Krems Gernot
A	Requirements on a higher level	Sander Manuel
A	Use Cases - SOLL	Sander Manuel

Tabelle 61 - Meeting Protokolle-Kickoff Meeting-4

21.8.1.3. Thema 3 - Mockups

Art	Beschreibung	Verantwortlich
A	Derzeitige Mockups abzeichnen	Krems Gernot
B	Weitere Mockups planen (Krems, Zach, Sander)	Krems Gernot

Tabelle 62 - Meeting Protokolle-Kickoff Meeting-5



21.8.1.4. Thema 4 - Datenbank/allgemeine Punkte

Art	Beschreibung	Verantwortlich
A	Screenshots – Robot	Biernat Dieter
A	Logo entwerfen (PD)	Biernat Dieter
A	Farbpalette (perfect:net)	Biernat Dieter
A	Ablaufdokumentation (Domain/Web Host)	Biernat Dieter
A	Datenbank besprechen mit Herrn Biernat	Zach Johannes
I	Domain Status für PD	Sander Manuel
I	Journaling für PD	Sander Manuel
I	Möglicherweise Dashboard Theme für Bootstrap	Sander Manuel

Tabelle 63 - Meeting Protokolle-Kickoff Meeting-6



21.8.2.Meeting Schnittstelle

Meeting Minutes

An	Organisator	Vertraulichkeit
Teilnehmer und informierte Personen	Sander Manuel	Öffentlich
Autor	Erstellungsdatum	
Sander Manuel	03. November 2016	
Datum	Uhrzeit	Ort
27. Oktober 2016	18:00 – 23:00	Rasthaus Dokl bei Gleisdorf

Tabelle 64 - Meeting Protokolle-Meeting Schnittstelle-1

Teilnehmer	Informierte Personen
Hr. Gernot Krems	Hr. Bernd Hochwarter
Hr. Manuel Sander	
Hr. Johannes Zach	
Hr. Dieter Biernat	

Tabelle 65 - Meeting Protokolle-Meeting Schnittstelle-2



21.8.2.1. Thema 1 – Schnittstelle Datenbank

Art	Beschreibung	Verantwortlich
I	Spalten der Datenbank abgeglichen	Manuel Sander
I	Mögliche Probleme mit der Datenanbindung besprochen	Manuel Sander
A	Tabelle „Rechnungsverlauf“ abbilden	Johannes Zach
A	Diverse „junction tables“ erstellen	Johannes Zach
A	Tabellen für internes Taskmanagement abbilden	Johannes Zach

Tabelle 66 - Meeting Protokolle-Meeting Schnittstelle-3



PartnerDashboard

Ordnerstruktur





21.9. Ordnerstruktur

```

1 PartnerDashboard
2   index.php
3   ReadMe.md
4   app
5     index.php
6     ajax
7       index.php
8     conf
9       pDashboard-dataTable-german.json
10    css
11      pdashboard-admin_permission.css
12      pdashboard-footer.css
13      pdashboard-jconfirm-custom.css
14      pdashboard-main.css
15      pdashboard-master.css
16      pdashboard-news.css
17    forms
18      addAddress.php
19      adm_permission_group_create.php
20      adm_permission_group_member_add.php
21      deleteAddress.php
22      editAddress.php
23      editContact.php
24      inputTwoFactor.php
25      selectCustomer.php
26    general
27      footer.php
28      navbar.php
29    img
30      access_management.png
31      agb_perfectnet.pdf
32      bg_main.jpg
33      bill_management.png
34      customer_overview.png
35      dieter_biernat.jpg
36      domainCheck.png
37      ecg.gif
38      false.png
39      faq.png
40      ic_face_black_24dp.png
41      ic_faq.png
42      ic_faq_32.png
43      ic_info_outline_black_24dp.png
44      ic_kundenebersicht.png
45      ic_kundenebersicht_32.png
46      ic_news.png
47      ic_news_32.png
48      ic_produktuebersicht.png
49      ic_produktuebersicht_32.png
50      ic_rechnunguebersicht.png
51      ic_rechnunguebersicht_32.png
52      ic_shopping_cart_black_48dp.png
53      ic_statistik.png
54      ic_statistik_32.png
55      logo.png
56      logoBackground.png
57      news.png
58      products.png
59      requests.png
60      statistics.png
61      Statistik2.jpg
62      temp.png
63      true.png
64      user_management.png
65      vcard.png
66    js
67      faq.js
68      pDashboard-dataTable.js.inc
69      pDashboard-prompt.js
70      pDashboard-serialize.js
71      pDashboard_main.js
72      newAccount.js
73    lib
74      bootstrap-3.3.7-dist
75      datatables-1.10.13
76      jquery-3.1.1
77      plugins
78        _plugins.txt
79        faq-template
80        jquery-confirm-v3.0.1
81        notify
82        toastr
83        summernote-0.8.2-dist
84    sites
85      adm_bulletin_board.php
86      adm_permission.php
87      adm_request_overview.php
88      adm_request_review.php
89      benutzerverwaltung.php
90      cart_overview.php
91      dashboard.php
92      development.php
93      dev_summernote.php
94      domainregistrieren.php
95      faq.php
96      handleaendern.php
97      handlegenerieren.php
98      handleloeschen.php
99      impressum.php
100     kundeneu.php

```

```

101    kundeninformationen.php
102    kundenuebersicht.php
103    login.php
104    mail.php
105    news.php
106    products_overview.php
107    rechnunguebersicht.php
108    select_customer.php
109    setup.php
110    statistik.php
111    verifuegbareklausurcheck.php
112    newAccount.php
113    request_engine
114      pending_inquiry.php
115    templates
116      pdashboard-default.php
117      pdashboard-login.php
118    conf
119      config-template.php
120      config.php
121    ext
122      SOURCES.md
123      FPDF
124      GoogleAuthenticator
125      PHPMailer
126    install
127      CreateDatabase.php
128      resetStoredProcedures.php
129      SqlInstall.php
130    SQL
131      CreateScripts
132        Additional
133          SP_GROUP_ADD_MEMBER.sql
134          SP_GROUP_ADD_PERMISSION.sql
135          SP_GROUP_LOAD_MEMBERS.sql
136          SP_GROUP_LOAD_PERMISSIONS.sql
137          SP_GROUP_REMOVE_MEMBER.sql
138          SP_GROUP_REMOVE_PERMISSION.sql
139        General
140          Account
141            SP_ACCOUNT_COUNT_ALL.sql
142            SP_ACCOUNT_CREATE.sql
143            SP_ACCOUNT_DELETE.sql
144            SP_ACCOUNT_DELETE_ALL.sql
145            SP_ACCOUNT_LOAD.sql
146            SP_ACCOUNT_LOAD_ALL.sql
147            SP_ACCOUNT_LOAD_BY_SECURITYIDENTIFIER.sql
148            SP_ACCOUNT_LOAD_SECURITY_TOKEN.sql
149            SP_ACCOUNT_SAVE.sql
150            SP_ACCOUNT_SEARCH_MATCHING.sql
151          Address
152            SP_ADDRESS_COUNT_ALL.sql
153            SP_ADDRESS_CREATE.sql
154            SP_ADDRESS_DELETE.sql
155            SP_ADDRESS_DELETE_ALL.sql
156            SP_ADDRESS_LOAD.sql
157            SP_ADDRESS_LOAD_ALL.sql
158            SP_ADDRESS_SAVE.sql
159            SP_ADDRESS_SEARCH_MATCHING.sql
160          BulletinBoard
161            SP_BULLETINBOARD_COUNT_ALL.sql
162            SP_BULLETINBOARD_CREATE.sql
163            SP_BULLETINBOARD_DELETE.sql
164            SP_BULLETINBOARD_DELETE_ALL.sql
165            SP_BULLETINBOARD_GET_DELIVERYTYPES.sql
166            SP_BULLETINBOARD_LOAD.sql
167            SP_BULLETINBOARD_LOAD_ALL.sql
168            SP_BULLETINBOARD_LOAD_FILTERED.sql
169            SP_BULLETINBOARD_REMOVE_DELIVERYTYPE.sql
170            SP_BULLETINBOARD_SAVE.sql
171            SP_BULLETINBOARD_SEARCH_MATCHING.sql
172          Contact
173            SP_CONTACT_COUNT_ALL.sql
174            SP_CONTACT_CREATE.sql
175            SP_CONTACT_DELETE.sql
176            SP_CONTACT_DELETE_ALL.sql
177            SP_CONTACT_LOAD.sql
178            SP_CONTACT_LOAD_ALL.sql
179            SP_CONTACT_SAVE.sql
180            SP_CONTACT_SEARCH_MATCHING.sql
181            SP_CONTACT_UPDATE_DEFAULT_ADDRESS_ID.sql
182          Contract
183            SP_CONTRACT_COUNT_ALL.sql
184            SP_CONTRACT_CREATE.sql
185            SP_CONTRACT_DELETE.sql
186            SP_CONTRACT_DELETE_ALL.sql
187            SP_CONTRACT_LOAD.sql
188            SP_CONTRACT_LOAD_ALL.sql
189            SP_CONTRACT_SAVE.sql
190            SP_CONTRACT_SEARCH_MATCHING.sql
191          Country
192            SP_COUNTRY_COUNT_ALL.sql
193            SP_COUNTRY_CREATE.sql
194            SP_COUNTRY_DELETE.sql
195            SP_COUNTRY_DELETE_ALL.sql
196            SP_COUNTRY_LOAD.sql
197            SP_COUNTRY_LOAD_ALL.sql
198            SP_COUNTRY_SAVE.sql
199            SP_COUNTRY_SEARCH_MATCHING.sql
200          Currency

```

Abbildung 154 - Ordnerstruktur 1-100

Abbildung 155 - Ordnerstruktur 101-200



```

201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
  SP_CURRENCY_COUNT_ALL.sql
  SP_CURRENCY_CREATE.sql
  SP_CURRENCY_DELETE.sql
  SP_CURRENCY_DELETE_ALL.sql
  SP_CURRENCY_LOAD.sql
  SP_CURRENCY_LOAD_ALL.sql
  SP_CURRENCY_SAVE.sql
  SP_CURRENCY_SEARCH_MATCHING.sql
  FAQ
    SP_FAQ_COUNT_ALL.sql
    SP_FAQ_CREATE.sql
    SP_FAQ_DELETE.sql
    SP_FAQ_DELETE_ALL.sql
    SP_FAQ_LOAD.sql
    SP_FAQ_LOAD_ALL.sql
    SP_FAQ_SAVE.sql
    SP_FAQ_SEARCH_MATCHING.sql
  Invoice
    SP_INVOICE_COUNT_ALL.sql
    SP_INVOICE_CREATE.sql
    SP_INVOICE_DELETE.sql
    SP_INVOICE_DELETE_ALL.sql
    SP_INVOICE_LOAD.sql
    SP_INVOICE_LOAD_ALL.sql
    SP_INVOICE_LOAD_GTE_DATE.sql
    SP_INVOICE_SAVE.sql
    SP_INVOICE_SEARCH_MATCHING.sql
  Log
    SP_LOG_COUNT_ALL.sql
    SP_LOG_CREATE.sql
    SP_LOG_DELETE.sql
    SP_LOG_DELETE_ALL.sql
    SP_LOG_LOAD.sql
    SP_LOG_LOAD_ALL.sql
    SP_LOG_SAVE.sql
    SP_LOG_SEARCH_MATCHING.sql
  Order
    SP_ORDER_ADD_PRODUCT.sql
    SP_ORDER_COUNT_ALL.sql
    SP_ORDER_CREATE.sql
    SP_ORDER_DELETE.sql
    SP_ORDER_DELETE_ALL.sql
    SP_ORDER_LOAD.sql
    SP_ORDER_LOAD_ALL.sql
    SP_ORDER_LOAD_CART.sql
    SP_ORDER_LOAD_PRODUCTS.sql
    SP_ORDER_REMOVE_PRODUCT.sql
    SP_ORDER_SAVE.sql
    SP_ORDER_SEARCH_MATCHING.sql
  Permission
    SP_PERMISSION_COUNT_ALL.sql
    SP_PERMISSION_CREATE.sql
    SP_PERMISSION_DELETE.sql
    SP_PERMISSION_DELETE_ALL.sql
    SP_PERMISSION_LOAD.sql
    SP_PERMISSION_LOAD_ALL.sql
    SP_PERMISSION_SAVE.sql
    SP_PERMISSION_SEARCH_MATCHING.sql
  Product
    SP_PRODUCT_COUNT_ALL.sql
    SP_PRODUCT_CREATE.sql
    SP_PRODUCT_DELETE.sql
    SP_PRODUCT_DELETE_ALL.sql
    SP_PRODUCT_LOAD.sql
    SP_PRODUCT_LOAD_ALL.sql
    SP_PRODUCT_SAVE.sql
    SP_PRODUCT_SEARCH_MATCHING.sql
  ProductTemplate
    SP_PRODUCTTEMPLATE_COUNT_ALL.sql
    SP_PRODUCTTEMPLATE_CREATE.sql
    SP_PRODUCTTEMPLATE_DELETE.sql
    SP_PRODUCTTEMPLATE_DELETE_ALL.sql
    SP_PRODUCTTEMPLATE_LOAD.sql
    SP_PRODUCTTEMPLATE_LOAD_ALL.sql
    SP_PRODUCTTEMPLATE_SAVE.sql
    SP_PRODUCTTEMPLATE_SEARCH_MATCHING.sql
  Request
    SP_REQUEST_COUNT_ALL.sql
    SP_REQUEST_CREATE.sql
    SP_REQUEST_DELETE.sql
    SP_REQUEST_DELETE_ALL.sql
    SP_REQUEST_LOAD.sql
    SP_REQUEST_LOAD_ALL.sql
    SP_REQUEST_SAVE.sql
    SP_REQUEST_SEARCH_MATCHING.sql
    SP_REQUEST_UPDATE_EXECUTION_STATE.sql
  RequestType
    SP_REQUESTTYPE_COUNT_ALL.sql
    SP_REQUESTTYPE_CREATE.sql
    SP_REQUESTTYPE_DELETE.sql
    SP_REQUESTTYPE_DELETE_ALL.sql
    SP_REQUESTTYPE_LOAD.sql
    SP_REQUESTTYPE_LOAD_ALL.sql
    SP_REQUESTTYPE_SAVE.sql
  Task
    SP_TASK_COUNT_ALL.sql
    SP_TASK_CREATE.sql
    SP_TASK_DELETE.sql
    SP_TASK_DELETE_ALL.sql
    SP_TASK_LOAD.sql

```

```

301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
  SP_TASK_LOAD_ALL.sql
  SP_TASK_LOAD_GLOBAL_TASK.sql
  SP_TASK_SAVE.sql
  SP_TASK_SEARCH_MATCHING.sql
  SP_TASK_UPDATE_EXECUTION_STATE.sql
  TaskInquiry
    SP_TASKINQUIRY_CREATE.sql
    SP_TASKINQUIRY_SAVE.sql
    SP_TASKINQUIRY_SEARCH_MATCHING.sql
  TaskParameter
    SP_TASKPARAMETER_CREATE.sql
    SP_TASKPARAMETER_SAVE.sql
    SP_TASKPARAMETER_SEARCH_MATCHING.sql
  TaskType
    SP_TASKTYPE_COUNT_ALL.sql
    SP_TASKTYPE_CREATE.sql
    SP_TASKTYPE_DELETE.sql
    SP_TASKTYPE_DELETE_ALL.sql
    SP_TASKTYPE_LOAD.sql
    SP_TASKTYPE_LOAD_ALL.sql
    SP_TASKTYPE_SAVE.sql
  Junction
    OrderedProducts
      SP_ORDEREDPRODUCT_COUNT_ALL.sql
      SP_ORDEREDPRODUCT_CREATE.sql
      SP_ORDEREDPRODUCT_SEARCH_MATCHING.sql
      SP_ORDEREDPRODUCT_SET_STATE.sql
    ProductDetail
      SP_PRODUCTDETAIL_CREATE.sql
    ProductTemplateDetail
      SP_PRODUCTTEMPLATEDETAIL_SEARCH_MATCHING.sql
  Metadata
    SP_GET_METADATA_BY_ID.sql
    SP_GET_ORDERSTATE_BY_ID.sql
    SP_METADATA_COUNT_ALL.sql
    SP_METADATA_CREATE.sql
    SP_METADATA_DELETE.sql
    SP_METADATA_DELETE_ALL.sql
    SP_METADATA_LOAD.sql
    SP_METADATA_LOAD_ALL.sql
    SP_METADATA_SAVE.sql
    SP_SEARCH_PRODUCTDETAILS.sql
  CreateTables
    00_00_title.sql
    01_00_country.sql
    02_00_contact.sql
    03_00_invoiceState.sql
    04_00_invoiceDeliveryType.sql
    05_00_contractState.sql
    06_00_contract.sql
    07_00_invoice.sql
    08_00_account.sql
    09_00_group.sql
    10_00_authorizationRole.sql
    11_00_groupPermission.sql
    12_00_groupMembers.sql
    13_00_logCategory.sql
    14_00_logEntryType.sql
    15_00_logEntrySeverity.sql
    16_00_log.sql
    50_00_faq.sql
  Data
    01_00_title_data.sql
    02_00_country_data.sql
    DummyData
      00_00_contact_data_dummy.sql
  log
    debug.log
  src
    PartnerDashboard.inc.php
    Communication
      Ajax.php
      Mail.php
    Core
      ClassLoader.php
      Configuration.php
      Session.php
    Data
      Dao
        MySQL
          General
            AccountDao.php
            AddressDao.php
            BulletinBoardDao.php
            ContactDao.php
            ContractDao.php
            CountryDao.php
            CurrencyDao.php
            FaqDao.php
            InvoiceDao.php
            LogDao.php
            OrderDao.php
            PermissionDao.php
            ProductDao.php
            ProductTemplateDao.php
            RequestDao.php
            RequestTypeDao.php
            SettingsDao.php
            TaskDao.php
            TaskInquiryDao.php

```

Abbildung 156 - Ordnerstruktur 201-300

Abbildung 157 - Ordnerstruktur 301-400



401				
402				
403				
404				
405				
406				
407				
408				
409				
410				
411				
412				
413				
414				
415				
416				
417				
418				
419				
420				
421				
422				
423				
424				
425				
426				
427				
428				
429				
430				
431				
432				
433				
434				
435				
436				
437				
438				
439				
440				
441				
442				
443				
444				
445				
446				
447				
448				
449				
450				
451				
452				
453				
454				
455				
456				
457				
458				
459				
460				
461				
462				
463				
464				
465				
466				
467				
468				
469				
470				
471				
472				
473				
474				
475				
476				
477				
478				
479				
480				
481				
482				
483				
484				
485				
486				
487				
488				
489				
490				
491				
492				
493				
494				
495				
496				
497				
498				
499				
500				

Abbildung 158 - Ordnerstruktur 401-500

501				
502				
503				
504				
505				
506				
507				
508				
509				
510				
511				
512				
513				
514				
515				
516				
517				
518				
519				
520				
521				
522				
523				
524				
525				
526				
527				
528				
529				
530				
531				
532				
533				
534				
535				
536				
537				
538				
539				
540				
541				
542				
543				
544				
545				
546				
547				
548				
549				
550				
551				
552				
553				
554				
555				
556				
557				
558				
559				
560				
561				
562				
563				
564				
565				
566				
567				
568				
569				
570				
571				
572				
573				
574				
575				
576				
577				
578				
579				
580				
581				
582				
583				
584				
585				
586				
587				
588				
589				
590				
591				
592				
593				
594				
595				
596				
597				
598				
599				
600				

Abbildung 159 - Ordnerstruktur 501-600



```

601           TopLevelDomain.php
602   Database
603       ConnectionManager.php
604   Enumeration
605       ContractState.php
606       ExecutionState.php
607       OrderState.php
608   Exception
609       RequestEngine
610           SkipException.php
611           WaitException.php
612   Log
613       Log.php
614       LogInterface.php
615       LogLevel.php
616   RequestEngine
617       BaseInquiryTask.php
618       BaseRequest.php
619       BaseTask.php
620   Security
621       Authentication.php
622       Permission.php
623       TwoFactorAuthentication.php
624   Utility
625       Utility.php
626   Dao
627       MySQL
628           BaseDao.php
629       XML
630           BaseDao.php
631   Interface
632       IBaseDao.php
633       IFactory.php
634       IIdentifiable.php
635       ILoggable.php
636       IMetadata.php
637       ISerializable.php
638   src-business
639       Business.inc.php
640   Business
641   Communication
642   Core
643       ClassLoader.php
644   RequestEngine
645       RequestFactory.php
646       RequestProcessor.php
647       RequestType.php
648       TaskFactory.php
649       TaskType.php
650   RequestTemplate
651       BaseRequestTemplate.php
652   Core
653       FinalizeOrderRequest.php
654   Demo
655       SendMailRequest.php
656   Product
657       OrderWebhostingPackage.php
658   Temp
659       CreateInvoicesRequest.php
660   Site
661       Site.php
662   Task
663       Communication
664           SendMailTask.php
665       Core
666           ApprovedTask.php
667           CancelRequestTask.php
668           DeclinedTask.php
669           WaitForDateTask.php
670           WaitForRequestTask.php
671   Demo
672       DemoTask1.php
673       DemoTask2.php
674   Inquiry
675       InquiryApprovalTask.php
676   Product
677       ActivateProductTask.php
678       DisableProductTask.php
679   System
680       ExecuteCommand.php
681   Invoice
682       CreateInvoicesTask.php
683   Webhosting
684       CreateStorageTask.php
685       EnableMailFeaturesTask.php
686       EnableMySqlTask.php
687       EnablePhpmyadminTask.php
688       EnablePhppgadminTask.php
689       EnableSecurityFeaturesTask.php
690       EnableSslTask.php
691       EnableWebmailTask.php
692   data
693       iconSet.pdn
694       iconSet32.pdn
695       iconSet_32.pdn
696       iconSet_48.pdn
697       pDashboard.mwb
698   script
699       launchRequestProcessor.sh
700   test

```

Abbildung 160 - Ordnerstruktur 601-700

```

701           false.png
702           qualityAssurance.php
703           true.png
704   Pattern
705       QA_IT_001_ManageCustomer.php
706       QA_IT_001_ManageCustomer.xml
707       QA_IT_002_ManageAccount.php
708       QA_IT_002_ManageAccount.xml
709       QA_IT_003_ManageFAQ.php
710       QA_IT_003_ManageFAQ.xml
711       QA_IT_004_ManageBulletinBoard.php
712       QA_IT_004_ManageBulletinBoard.xml
713       QA_IT_005_OrderProduct.php
714       QA_IT_005_OrderProduct.xml
715       QA_IT_006_AccessManagement.php
716       QA_IT_006_AccessManagement.xml
717       QA_Proxy_AccountProxy.php
718       QA_Proxy_AccountProxy.xml
719       QA_ValueObject_Account.php
720       QA_ValueObject_Account.xml
721       QA_ValueObject_Address.php
722       QA_ValueObject_Address.xml
723       QA_ValueObject_BulletinBoard.php
724       QA_ValueObject_BulletinBoard.xml
725       QA_ValueObject_BulletinBoardDelivery.php
726       QA_ValueObject_BulletinBoardDelivery.xml
727       QA_ValueObject_Contract.php
728       QA_ValueObject_Contract.xml
729       QA_ValueObject_Contract.php
730       QA_ValueObject_Contract.xml
731       QA_ValueObject_Currency.php
732       QA_ValueObject_Currency.xml
733       QA_ValueObject_Faq.php
734       QA_ValueObject_Faq.xml
735       QA_ValueObject_Invoice.php
736       QA_ValueObject_Invoice.xml
737       QA_ValueObject_Log.php
738       QA_ValueObject_Log.xml
739       QA_ValueObject_Order.php
740       QA_ValueObject_Order.xml
741       QA_ValueObject_Product.php
742       QA_ValueObject_Product.xml
743       QA_ValueObject_ProductDetail.php
744       QA_ValueObject_ProductDetail.xml
745       QA_ValueObject_Settings.php
746       QA_ValueObject_Settings.xml
747
748
749
750
751

```

Abbildung 161 - Ordnerstruktur 701-751