



Schriftliche Ausarbeitung

Dokumentation Anwendung „Digitaler Briefkasten“
im Rahmen des Modul „AWE1“

Prüfer:

Christian Heuermann

Erstellt von:

Jonathan Brockhausen, Phillip Röring, Julius Figge

Studiengang:

Angewandte Informatik B.Sc.

Eingereicht am:

23. Mai 2020

Inhaltsverzeichnis

Abbildungsverzeichnis	IV
Tabellenverzeichnis	V
Listingverzeichnis	VI
1 Installation	1
2 Anleitung	2
3 Technologien	3
4 Entity-Relationship Diagramm	4
5 Security	6
6 Test	8
6.1 manuelle-„Klicktests“	8
7 Use-Cases	12
8 GUI-Konzept	14
9 Konzepte	17
9.1 MVC-Pattern	17
9.2 Teamarbeit und Tests	17
10 Projektplanung	18
10.1 Projektstrukturplan	18
10.2 Projekt	18
Anhang	19
Quellenverzeichnis	34

Abbildungsverzeichnis

Abbildung 1: ERD des Projekts	4
Abbildung 2: Use-Case Diagramm	12
Abbildung 3: Farben Konzept	14
Abbildung 4: Ideen Konzept	15
Abbildung 5: Rechtsklick Umsetzung	16
Abbildung 6: Dropdown Umsetzung	16
Abbildung 7: Administrator - Use-Case Diagramm	20
Abbildung 8: Kontaktformular - Use-Case Diagramm	21
Abbildung 9: GUI-Konzept - Login	22
Abbildung 10: GUI-Konzept - Registrierung	23
Abbildung 11: GUI-Konzept - Willkommen	23
Abbildung 12: GUI-Konzept - Idee erstellen	24
Abbildung 13: GUI-Umsetzung - Login	25
Abbildung 14: GUI-Umsetzung - Registrierung	26
Abbildung 15: GUI-Umsetzung - Ideen	27
Abbildung 16: GUI-Umsetzung - Idee erstellen	28
Abbildung 17: GUI-Umsetzung - Idee ansehen	29
Abbildung 18: GUI-Umsetzung - Admin Ansicht	30
Abbildung 19: GUI-Umsetzung - Spezialist Ansicht	31
Abbildung 20: Projektstrukturplan	32

Tabellenverzeichnis

Tabelle 1: GUI-Testdurchführung	9
---	---

Listingverzeichnis

1 Installation

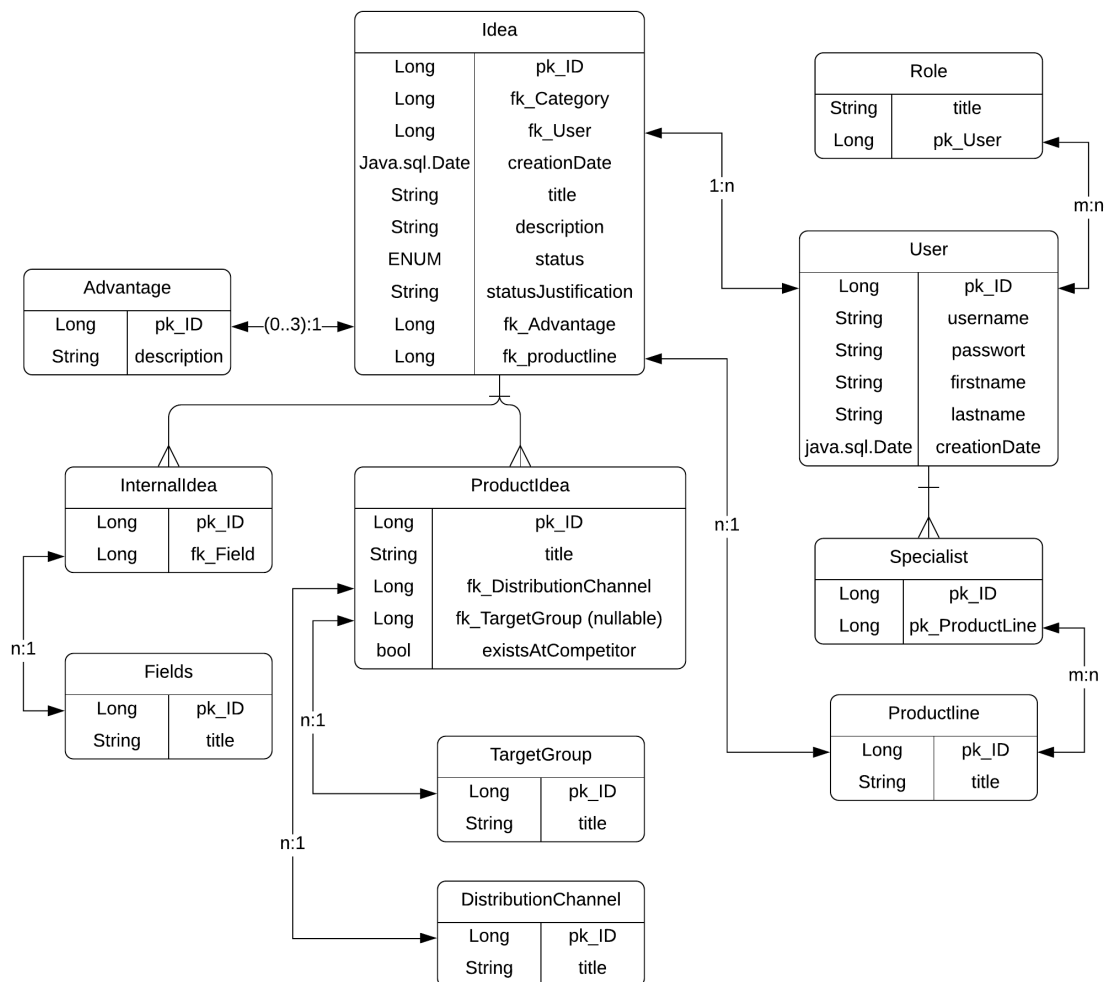
2 Anleitung

3 Technologien

4 Entity-Relationship Diagramm

In Abb. 1 ist das zugrundeliegende Entity-Relationship Diagramm dargestellt.

Abbildung 1: ERD des Projekts



Quelle: Eigene Darstellung

Dieses Diagramm wurde im Programm in der Datenbank umgesetzt. Im Folgenden werden einige Design-Entscheidungen erläutert.

Zuordnung von Fachspezialisten und Ideen

Um die automatische Zuordnung von Fachspezialisten zu Ideen umzusetzen, haben wir die Produktparte als Zuordnungskriterium herangezogen. Die Klasse des Fachspezialisten erbt vom der Klasse des Benutzers mit der zusätzlichen Eigenschaft, dass ihm eine oder mehrere Produktlinien zugewiesen sind. Einer Produktparte können mehrere Fachspezialisten zuge-

ordnet sein. Mit dieser m:n-Beziehung erreicht das Programm die größtmögliche Flexibilität. Jeder Idee (Intern und Produkt) ist eine Produktparte zugeordnet. Aus den Projektanforderungen ergeben sich mehrere Produktparten, die bei der Auslieferung bereits vorhanden sind. Da interne Ideen gemäß der Anforderungen keine Produktparte besitzen, bekommen Sie die dem Benutzer verborgene Produktparte „INTERNAL“ zugewiesen. Diese ermöglicht für interne Ideen dieselbe Logik zu verwenden. Durch diese Umsetzung ist auch eine Erweiterung um weitere Ideenkategorien ohne Änderungen am übrigen Programm möglich.

Umsetzung von Status

Wir haben uns dagegen entschieden, den Status in eine eigene Entität im Sinne des ERD auszulagern. Erweiterungen und Änderungen der Status im Echtbetrieb erfordern dann zwar unter Umständen an einigen Stellen Änderungen in der Programmlogik aber die enum bieten insgesamt Performancevorteile gegenüber der Auslagerung als vollwertige Entität und sind leichter im Code umzusetzen.

Vererbung von Ideen zu interne und Produktidee

Das Anlegen von Klassen mit Vererbung (wie im Projekt bei Ideen und Usern) kann in relationalen Datenbanken zu zwei Schwierigkeiten führen:

1. Lese- und Speicherzugriffe betreffen mehrere Tabellen und erfordern Joins. Das kann zu unübersichtlichen Strukturen und SQL-Kommandos sowie zu geringerer Performance führen
2. Nicht-polymorphe Abfragen (z.B. Namen/Preise nur der Getraenke) sind umständlich (z.B. Unterscheidung per Diskriminator)¹

Durch die Verwendung von Hibernate mit seiner nativen Java-Integration und die Vermeidung von hardcodeten SQL-Statements im Code, fallen diese Punkte kaum ins Gewicht. Die Struktur der Vererbung ermöglicht es darüber hinaus sogar weitere Kategorien von Ideen anzulegen.

¹vgl. Horn, Torsten (2007)

5 Security

Die Security unseres Programmes wird durch mehrere Bestandteile sichergestellt.

Dazu gehören zum einen aktiv absichernde Elemente.² Diese bestehen aus mehreren Komponenten.

Zuerst ist der Login sowie die Registrierung abgesichert. Nutzer müssen ihr Passwort mit mindestens 8 Zeichen wählen. Dieses wird im Backend verschlüsselt gespeichert. Hierzu benutzen wir BCrypt als Passwort Encoder. Diesen verwenden wir mit einer Stärke von 10, diese bietet für uns die beste Balance zwischen Sicherheit und Performance. Mit dem Login erhalten Nutzer eine JSession ID zugewiesen mit der sie sich in weiteren Requests authentifizieren und über die sie identifiziert werden können.

Die nächste ist die URL-Zugriffskontrolle in der Klasse „SecurityConfig“ im Package „config“. In dieser wird festgelegt welche Requests durch Spring Security zugelassen werden. Nicht authentifizierte Nutzer haben hier nur Zugriff auf statische Elemente (wie z.B. Grafiken, Javascript und CSS), die Registrierung und der Ideenansicht. Authentifizierte Nutzer werden anhand ihrer Rolle unterschieden welche im Backend überprüft wird. Nutzer, Spezialisten und Administrator können nur auf die jeweils für sie relevanten Seiten zugreifen. Der durch Spring Security erstellte JSession-Cookie wird beim ausloggen an dieser Stelle invalidiert und gelöscht.

Darüber hinaus ist die Anwendung so konfiguriert, dass ein automatischer Session Timeout nach 15 Minuten erfolgt, auch hierbei wird die Session invalidiert.

Darüber hinaus werden alle Abfragen durch das Backend geprüft. An relevanten Stellen wird in den jeweiligen Controllern bereits vor der Bearbeitung des Requests die Rolle des aktuellen Users überprüft. Damit wird sichergestellt, dass Funktionen die insbesondere dem Administrator oder Spezialisten vorbehalten sind, nur durch diese durchgeführt werden können.

Außerdem werden übertragene Informationen in den bearbeitenden Services um die Berechtigung diese anzufragen, zu verändern oder zu speichern geprüft.

Durch diese Kontrolle an mehreren Stellen erreichen wir es zu kontrollieren und sicherzustellen welche Art von Requests (un-, authentifiziert), welcher User, welcher Rolle, welche Daten wie verwenden (lesen, bearbeiten, schreiben) dürfen.

Zum anderen wird die Sicherheit im weiteren durch passive Sicherheitselemente unterstützt. Hierzu gehört das Loggen von (versuchte-) anmelden, registrieren, und abmelden vom System.³ Dies wird durch mehrere Klassen im Package „log“ sichergestellt. Diese implementieren einen jeweiligen Application-Listener, beispielhaft für den fehlerhaften Login der ApplicationListener

²Zu beachten ist, dass wir das Programm unter der Prämisse entwickelt haben, dass im Livebetrieb eine zusätzliche SSL-Verschlüsselung für den Traffic genutzt wird.

³Das Loggen von Session Timeouts konnte aufgrund von Komplikationen zum Abgabzeitpunkt nicht fertiggestellt werden.

AuthenticationFailureBadCredentialsEvent. Beim auftreten eines passenden ApplicationEvents wird mit Hilfe eines Loggers, den „slf4j“ bereitgestellt wird ein Zeitstempel sowie Nutzernamen und IP-Adresse geloggt. Hierbei ist anzumerken, dass die Logs zusätzlich außerhalb der Konsole in eine Datei geschrieben werden. Diese ist auf 5Mb begrenzt und rotiert oberhalb dieser Grenze automatisch. Zudem sind die zu schreibenden Logs eingeschränkt. Damit stellen wir sicher, dass nur relevante Informationen festgehalten werden und diese auch unabhängig vom Programm zur Auswertung zur Verfügung stehen. Desweiteren sind Fehlermeldungen eingeschränkt um nicht aus versehen Informationen durchsickern zu lassen. Beispielhaft zeigt der Login ausschließlich eine Fehlermeldung bei fehlerhaften Daten an - jedoch nicht ob der Nutzernamen oder das Passwort falsch war. **Darüber hinaus werden Exceptions gefiltert und nur ausgewählte (respektive unsere eigenen) auf der Error-Seite angezeigt. Damit stellen wir sicher, dass nicht aus versehen Exceptions, Stacktraces oder Debug-Logs an das Frontend gelangen und für den Nutzer sichtbar sein könnten.**

6 Test

6.1 manuelle-„Klicktests“

Zur Überprüfung der „GUI“ sollen manuelle Klicktests durchgeführt werden. Diese sollen dokumentiert werden um Fehler möglichst gezielt beheben zu können.

Zu notierende Informationen

Zu den notierenden Informationen gehören zum einen die Programmrevision (Git Commit Hash, Datum) sowie der verwendete Branch. Darüber hinaus ist das genutzte Betriebssystem sowie der genutzte Browser (inklusive Build zu notieren). Bei Darstellungsfehlern ist es sinnvoll zudem Screenshots zu hinterlegen sowie die Bildschirmauflösung zu notieren. Diese Informationen sammeln wir gezielt sehr detailliert um Fehler besser eingrenzen zu können.

Testvorbereitung

1. Zum Testen wird der neueste Stand des master-Branches verwendet.
2. Hierzu ist zunächst die Datenbank zu löschen und mit Hilfe der in „HelperScriptsNoTests“ vorhandenen Tests zu füllen.
3. Der Code soll kompiliert werden und die entstandene „Jar“-Datei ausgeführt werden.
4. Nach Möglichkeit soll der Test auf mehreren Browsern ausgeführt werden. Hierbei ist zu beachten, dass alle Addons zu deaktivieren sind, um eventuelle Komplikationen auszuschließen.
5. Nachdem diese Voraussetzung geschaffen ist, sind die Tests durchzuführen und die obigen Informationen zu notieren.

Testdurchführung**Tabelle 1:** GUI-Testdurchführung

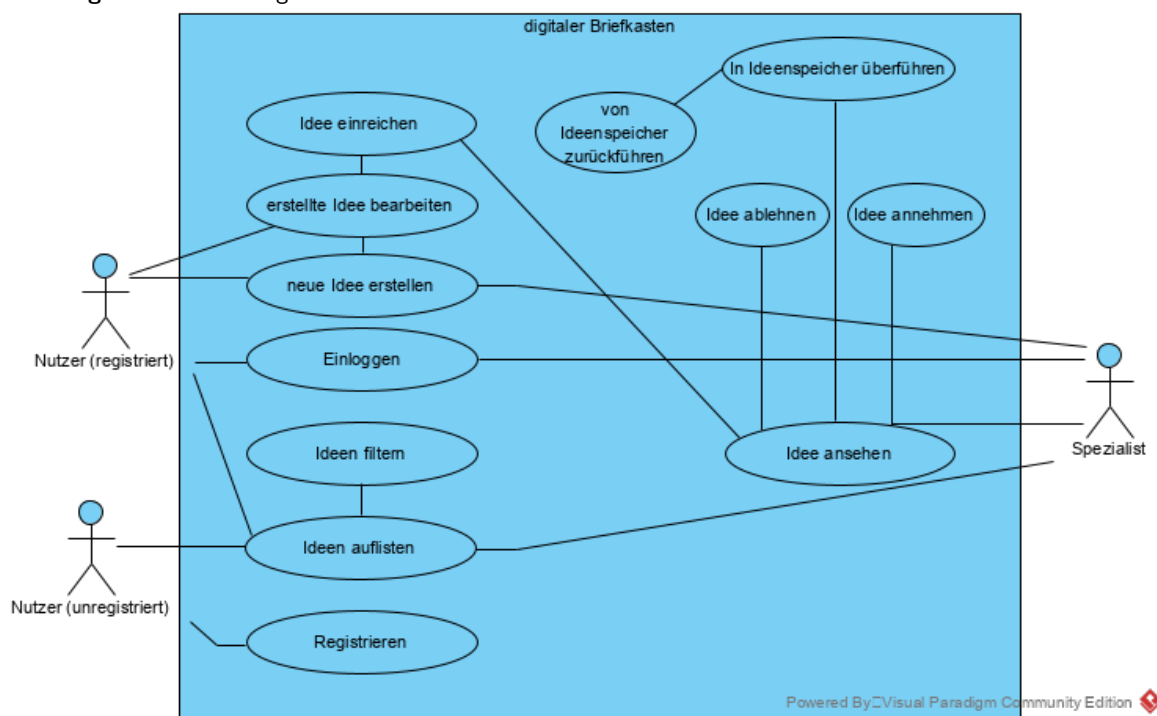
Aktion	erwartetes Ergebnis	Reaktion
Registrieren eines neuen Nutzers		
Bereits bestehenden Nutzernamen verwenden (admin)	Fehlermeldung - Nutzernamen existiert bereits	
zu kurzer Nutzernamen (<3)	Fehlermeldung - Daten falsch	
zu kurzes Passwort (<=7)	Fehlermeldung - zu kurzes Passwort	
nicht übereinstimmende Passwörter	Fehlermeldung - nicht stimmende Passwort	
mit korrekten Daten	eingeloggt sein	
Ausloggen aus dem Account	ausgeloggt sein	
Einloggen in erstellten Account		
mit falschem Passwort	Fehlermeldung	
mit falschem Nutzernamen	Fehlermeldung	
mit richtigem Passwort	eingeloggt sein	
Erstellen von beispielhaften Ideen		
Erstellen einer „internen Idee“	Idee erscheint in Tabelle nicht eingereicher Ideen	
Erstellen einer „Produkt-Idee“	Idee erscheint in Tabelle nicht eingereicher Ideen	
Erstellen einer beliebigen Idee mit fehlerhaften Werten	Fehlerhafte Attribute werden hervorgehoben	
Erstellen einer Idee von der bereits selber Name bei selbem Typ vorhanden	Fehlermeldung über Duplikat	
Bearbeitend der internen Idee	Änderungen werden übernommen	
Bearbeitend der Produkt-Idee	Änderungen werden übernommen	
Ideenübersicht		
Filtern der nicht eingereichten Ideen nach Attributen	nur Ideen mit passenden Attributen werden angezeigt	

Einreichen der erstellten Ideen	erfolgreicher Transfer in jeweilige Tabelle	
Ausloggen aus dem Account	ausgeloggt sein	
Idee Übersicht als nicht eingeloggtter Nutzer		
Filtern der Ideen in beiden Tabellen	nur Ideen mit passenden Attributen werden angezeigt	
Spezialist für „internen Idee“		
Einloggen als passender (Ideen sollten ihm zugewiesen sein) Spezialist (Zugangsdaten siehe <code>Manual.md</code>)	eingeloggt sein	
Übersicht zu entscheidender Ideen filtern	nur Ideen mit passenden Attributen werden angezeigt	
Entscheiden ohne Begründung	fehlendes Attribut wird hervorgehoben	
Idee in Ideenspeicher verschieben	Idee liegt in Ideenspeicher	
Spezialist für „Produkt-Idee“		
Account zu anderem Spezialist wechseln	Idee liegt in Ideenspeicher	
Entscheiden über Idee aus Ideenspeicher mit Auswahl „zur Entscheidung freigegeben“	Idee liegt in eigenen zu entscheidenden Ideen	
Idee aus Entscheidungsübersicht bewerten	Idee erscheint auf passender Tabelle in Ideenübersicht	
Administrator		
Account zu Administrator wechseln (Zugangsdaten siehe <code>Manual.md</code>)		
Existierende User prüfen	registrierter Account sowie alle Spezialisten werden aufgelistet	
Alle möglichen anzulegenden Felder durchgehen, bereits bestehenden Namen eingeben	bei jedem Feld wird ein Fehler angezeigt	
Alle möglichen anzulegenden Felder durchgehen	Feld wird angelegt	
Ausloggen	FERTIG!	

7 Use-Cases

Im nachfolgenden sind die Use-Cases des Programm dargestellt (Siehe Abb. 2). Diese sind den Projektvorgaben entnommen.⁴

Abbildung 2: Use-Case Diagramm



Quelle: Eigene Darstellung

Für das Use-Case Diagramm sind drei Rollen von Relevanz. Zuerst der **unregistrierte Nutzer**, welche die Sicht des Programmes für die Öffentlichkeit repräsentiert. Desweiteren der **eingeloggte Nutzer** der mehr Möglichkeiten hat, hierzu gehört auch der **Administrator**. Dieser hat über die Möglichkeiten des Nutzers weitere administrative Rechte.⁵ Jedoch besitzt er nicht die Rechte der dritten Rolle des **Spezialisten**.

Die Use-Cases lassen sich in zwei „Kern“-Kategorien unterteilen. Das sind zum einen die **Account** bezogenen Use-Cases.

Hierzu gehören der Vorgang des Einloggens sowie der Registrierung. Zu diesen ist anzumerken, dass Spezialisten sich lediglich einloggen können. Durch ihre administrative Rolle werden diese durch den Administrator angelegt.

Zum anderen ist der zweite Use-Case die **Erstellung und Bewertung von Ideen**.

⁴Vgl. Heuermann, Christian (2020)

⁵Der Admin ist als eigener Use-Case im Anhang dargestellt. Siehe Anhang 1.1 auf S.20

Eingereichte Ideen lassen sich durch alle Nutzer jeder Rolle einsehen und filtern. Darüber hinaus haben alle eingeloggten Nutzer die Möglichkeit Ideen zu erstellen, zu bearbeiten und zur Bewertung einzureichen.

Diese eingereichten Ideen werden durch Spezialisten bewertet oder gespeichert.

Desweiteren existiert die Möglichkeit für alle Nutzer dem Administrator der Plattform über ein Kontaktformular Nachrichten zu senden.⁶

⁶Das zugehörige Use-Case Diagramm findet sich im Anhang 1.2 auf S.20

8 GUI-Konzept

Wir haben uns entschieden, statt eines GUI-Mockups unser GUI-Konzept direkt im Prototypen mit auszuliefern. Das lässt sich durch mehrere Punkte begründen. Zuerst hatten wir zum Zeitpunkt der ersten Präsentation bereits einen funktionierenden Prototypen und konnten diesen direkt mit dem GUI-Konzept ausstatten. Dadurch hatten wir nicht nur ein Mockup sondern konnten bereits mit der GUI interagieren. Desweiteren hatten wir dadurch die Möglichkeit die Zeit für die Erstellung eines Konzeptes direkt in die Entwicklung funktionierender GUI zu stecken.

Die GUI wurde unter Nutzung von Bootstrap 4 in Kombination mit Font-Awesome für die Icons entwickelt, dadurch war es uns möglich eine konsistente, verständliche und klare Oberfläche zu entwickeln. Hierbei haben wir uns darauf konzentriert „eine klare Linie zu fahren“. Alle Seiten werden auf weissem Hintergrund dargestellt. Buttons und Informationen sind generell in grau (beziehungsweise Schwarz) gehalten. Abweichend hiervon treten Farben nur auf um die Aufmerksamkeit des Nutzers auf sich zu ziehen oder um Hinweise hervorzuheben. Diese Farben sind in Abb. 3 abgebildert. Die Verwendung wird im weiteren näher erläutert.

Abbildung 3: Farben Konzept



Quelle: Eigene Darstellung

Grundlegend sind die Elemente der Anwendung zentriert wie im weiteren zu sehen. Damit erreichen wir in Kombination mit der Nutzung von Bootstrap eine nahezu 100 prozentige Kompatibilität zu mobilen Endgeräten.⁷










Grundlegende Elemente dieses Konzeptes sind zum einen der Login-Screen (Siehe Abb. 9 im Anhang auf S.22). In diesem Screenshot ist auch das Logo der Anwendung zu sehen welches ebenfalls in den typischen Farben gehalten wurde. Dieses soll der Anwendung einen Wiedererkennungswert geben durch seine gleichzeitig humorvolle als auch simple Darstellung.

Die zweite zentrale Komponente des Konzeptes ist die Übersicht aller Ideen (Siehe Abb. 4). Auffallend ist hier die Gliederung der Ideen in Tabellen. Bereits zu diesem Zeitpunkt war

⁷Hierbei ist jedoch anzumerken, dass dieses Feature nicht gefordert war und somit auch nicht weitergehend getestet wurde. Allerdings sind bereits die Voraussetzungen für eine mögliche Erweiterung der Anwendung geschaffen.

geplant die Ideen nach Typ zu gliedern und in einer Übersicht mit ihren wichtigsten Eigenschaften darzustellen. Zu diesem Zeitpunkt noch per Klick⁸ sollte es möglich sein die Idee im Detail inklusive aller Informationen darzustellen. Diese Entscheidung begründet sich damit das Gleichgewicht zwischen der verfügbaren Information auf einer Seitenansicht und der Übersichtlichkeit zu wahren. Darüber hinaus findet sich auch hier die Farbgestaltung wieder, Grundsätzlich ist die Oberfläche Monochrom gehalten. Icons dienen der schnelleren Identifikation der verschiedenen Tabellen und der Übersichtlichkeit. Farbakzente sind zum einen zur Führung der Nutzer gedacht, siehe beispielhaft in dem Hyperlink auf den Ersteller der Ideen⁹. Zum anderen sind diese in den Status der Ideen mit einbezogen, hierdurch lässt sich erheblich schneller ein Überblick verschaffen.

Abbildung 4: Ideen Konzept

 Digitaler Briefkasten Home Create Idea List Ideas Logout				
 Not submitted Ideas				
title	description	creator	creation date	status
konzept 1	Testbeschreibung	konzept	2020-05-20	 not submitted
 Product-Ideas				
title	description	creator	creation date	status
weitere Idee	Testbeschreibung	konzept	2020-05-20	 pending
Geht nicht junge	geheime Nachricht	konzept	2020-05-20	 accepted
 Internal-Ideas				
title	description	creator	creation date	status
weitere Idee	Testbeschreibung	konzept	2020-05-20	 pending
Geht nicht junge	geheime Nachricht	konzept	2020-05-20	 accepted

Quelle: Eigene Darstellung

Ein weiterer relevanter Punkt der sich beispielhaft in dieser Abbildung (Siehe Abb. 4) findet ist die Navigationsleiste. Diese ist im Konzept nur nach dem Login vorhanden. Im fertigen Produkt wurde diese aber auf jeder Seite inkludiert.¹⁰ Diese ist zentrales Steuerelement der Anwendung. Auf der linken Seite findet sich das Logo dauerhaft präsent wieder. Daneben

⁸Diese Funktionalität wurde im weiteren durch ein Rechtsklick Menü erweitert (Siehe 2.2 S.25).

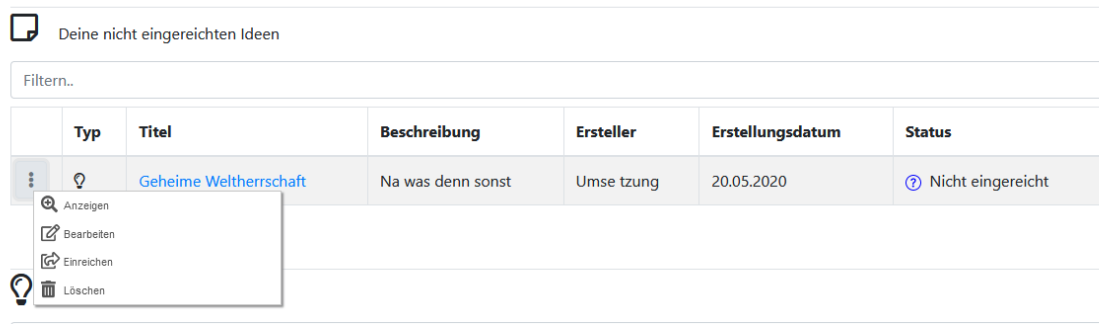
⁹Dieser Hyperlink stand beispielhaft für die Weiterleitung auf eine Detailseite auf der Tabellensicht.

¹⁰Ebenso wurde ein Footer eingefügt. Vgl. 2.2 S.25

werden zur Verfügung stehende Seiten angezeigt, wobei die aktuelle hervorgehoben ist.¹¹ Auf der rechten Seite findet sich der Logout Button, auch dieser ist hervorgehoben um vom Nutzer wahrgenommen zu werden.

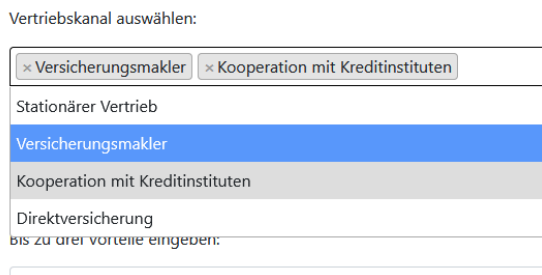
Die weiteren Konzeptteile der GUI finden sich im Anhang 2.1 auf S.22.

Abbildung 5: Rechtsklick Umsetzung



Quelle: Eigene Darstellung

Abbildung 6: Dropdown Umsetzung



Quelle: Eigene Darstellung

Hervorzuheben ist, dass gegenüber des Konzeptes in der Umsetzung¹² einige Elemente hinzugekommen sind. Die wichtigsten hierbei sind das bereits genannte Rechtsklick Menü (Siehe Abb. 5). Sowie intelligente Dropdown Menüs (Siehe Abb. 6). Diese sollen dem Nutzer die Möglichkeit geben intuitiv die Anwendung zu bedienen und erweitern diese durch dynamische Menüs welche sich in die Oberfläche einpassen.

¹¹Im fertigen Produkt ist diese Sicht abhängig von den verschiedenen Rollen. Vgl. 2.2 S.25

¹²Siehe Anhang 2.2 auf S.25

9 Konzepte

9.1 MVC-Pattern

In unserer Anwendung benutzen wir das Architekturmuster Model View Controller. Dieses Muster haben wir explizit ausgewählt, da Springboot zusammen mit Thymeleaf als Frontend hierfür sehr gut geeignet ist.

9.2 Teamarbeit und Tests

Unsere Team Arbeit haben wir auf den Austausch untereinander ausgerichtet. So konnten wir uns unsere unterschiedlichen Kompetenzen zu Nutze machen und beispielhaft im Mob-Programming¹³ Wissen vermittelt und uns gegenseitig unterstützt.

Darüber hinaus haben wir nach dem „All hands on Deck“-Prinzip¹⁴ gearbeitet.

Das haben wir zum einen aufbauend auf Teaminterne Kommunikation (beispielhaft über Telegram), aber insbesondere auch über die „Github CI“ erreicht. Diese war konfiguriert bei jedem Commit alle Tests durchzuführen und bei Problemen Email-Benachrichtigungen zu versenden.

Ausserdem haben wir „Sonarlint“ eingesetzt um unsere Code-Qualität zu überprüfen und stetig zu verbessern. Dadurch haben wir nicht nur unsere Code Qualität sichergestellt, sondern konnten auch auftretenden Probleme möglichst schnell erkennen und beheben.

Als netter Nebeneffekt lässt sich festhalten, dass durch das gemeinsame arbeiten Wissensilos effektiv aufgebrochen wurden und der Lerneffekt im Zuge des Projekts für alle beteiligten maximiert wurde.

¹³Hiermit ist das gemeinse Programmieren über ein Videotelefonat gemeint, bei dem abwechselnd eine Person den Bildschirm teilt.

¹⁴Dieses bezeichnet den Ansatz, bei auftretenden Problemen und Fehlern sich zuerst auf die Behebung dieser zu konzentrieren, bevor weitergehende Aufgaben bearbeitet werden.

10 Projektplanung

10.1 Projektstrukturplan

Das Projekt wurde von uns in vier Phasen aufgeteilt, *Vorbereitung, Implementierung, Dokumentation & Tests* und *Abschluss*. Der Projektstrukturplan ist im Anhang in ?? 3.1 dargestellt.

10.2 Projekt

Anhang

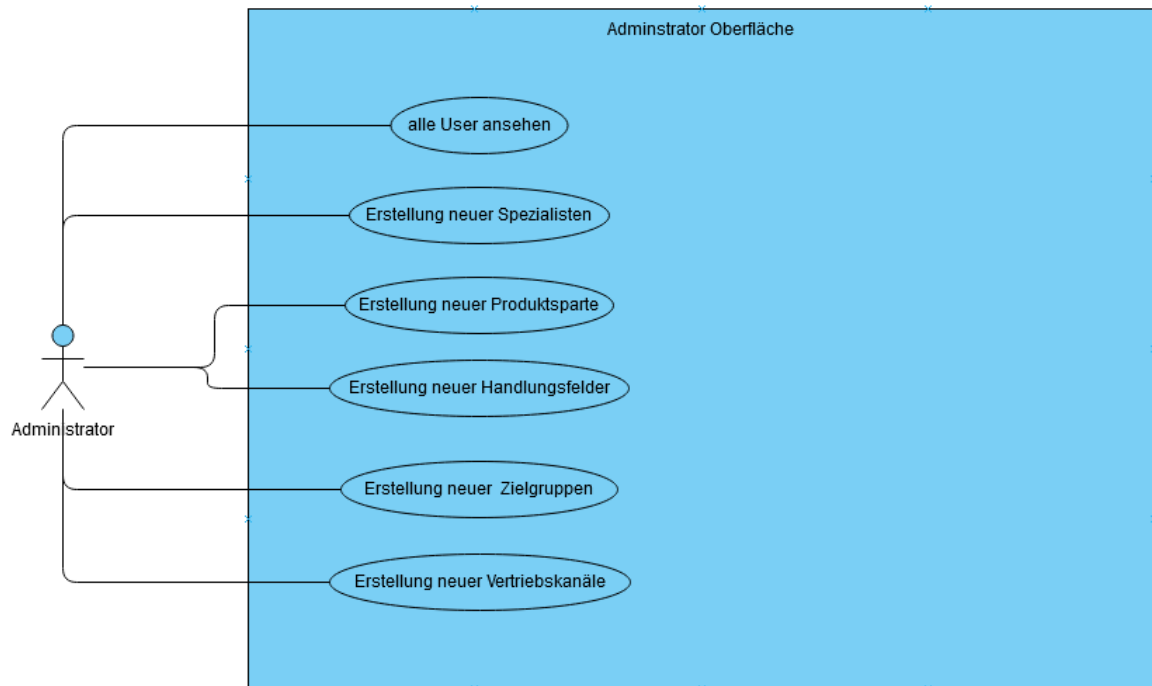
Anhangsverzeichnis

Anhang 1: Weitere Use-Cases	20
Anhang 1.1: Administrator	20
Anhang 1.2: Kontaktformular	20
Anhang 2: GUI-Konzept	22
Anhang 2.1: Konzept	22
Anhang 2.2: Umsetzung	25
Anhang 3: Projektplanung	32
Anhang 3.1: Projektstrukturplan	32

Anhang 1 Weitere Use-Cases

Anhang 1.1 Administrator

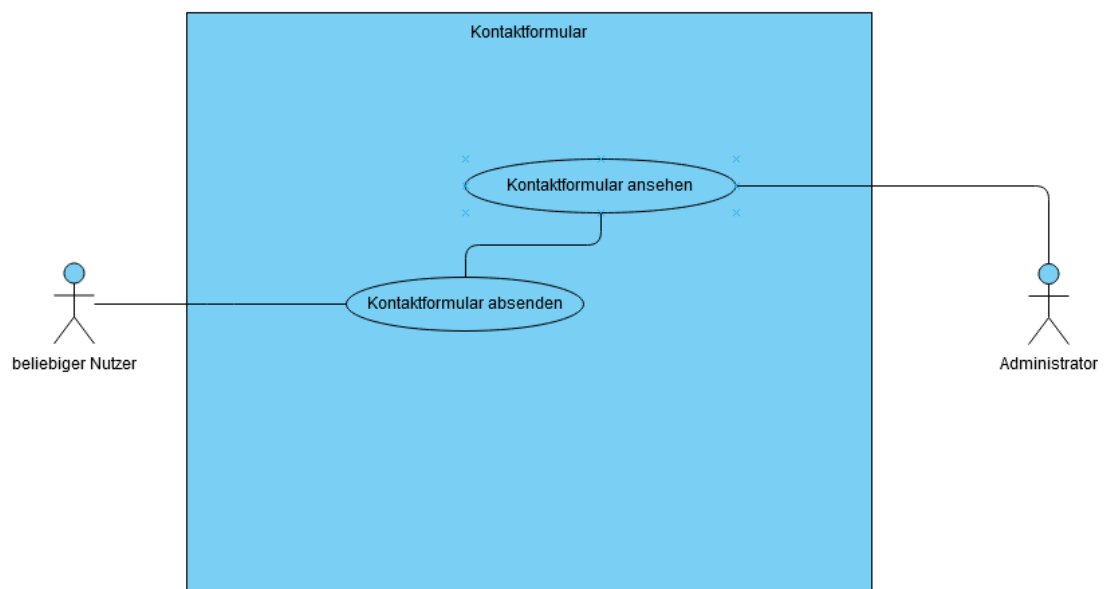
Abbildung 7: Administrator - Use-Case Diagramm



Quelle: Eigene Darstellung

Anhang 1.2 Kontaktformular

Abbildung 8: Kontaktformular - Use-Case Diagramm



Quelle: Eigene Darstellung

Anhang 2 GUI-Konzept

Anhang 2.1 Konzept

Abbildung 9: GUI-Konzept - Login



Digitaler
Briefkasten

Username

Password

Log in

Create an account

Quelle: Eigene Darstellung

Abbildung 10: GUI-Konzept - Registrierung

The registration form features a logo at the top consisting of a stylized mailbox with three curved lines above it representing signal waves. Below the logo, the text 'Digitaler Briefkasten' is centered. The form contains three input fields: 'Username', 'Password', and 'Password Confirmation', each with a light gray border. At the bottom is a dark gray button with the text 'Register' in white.

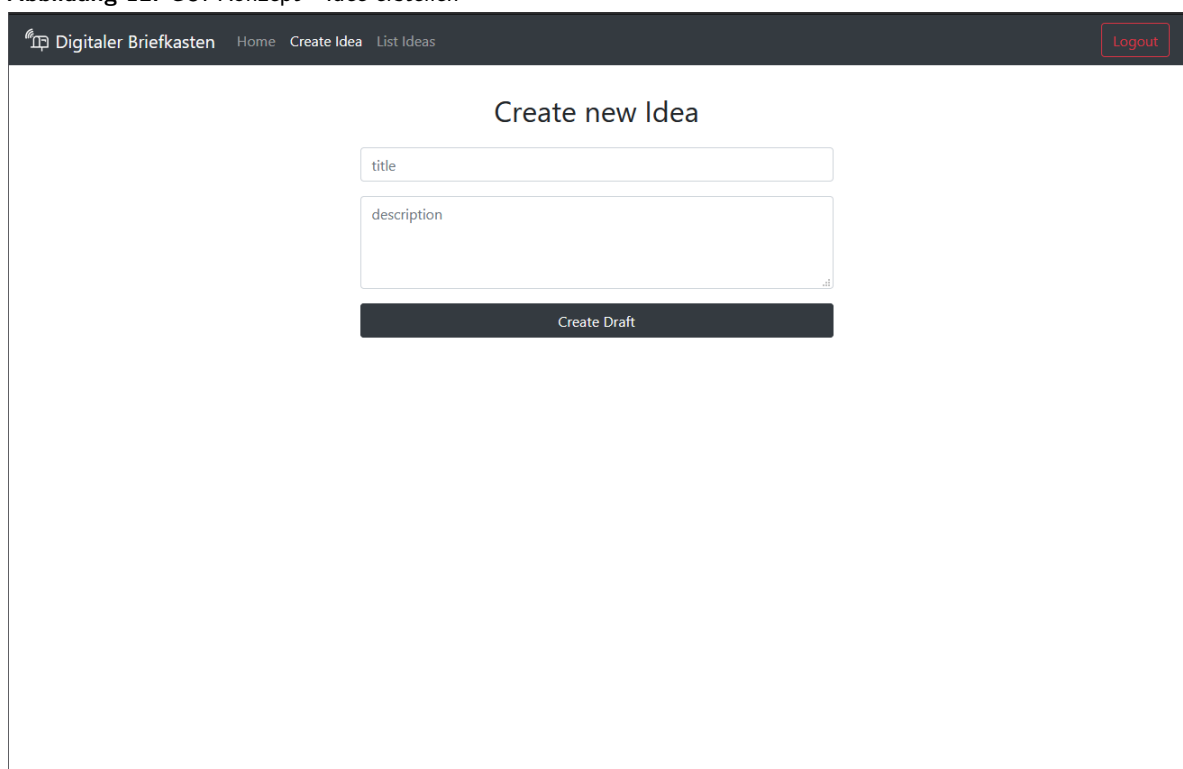
Quelle: Eigene Darstellung

Abbildung 11: GUI-Konzept - Willkommen

The welcome screen has a dark gray header bar. On the left, it contains a small icon and the text 'Digitaler Briefkasten'. In the center of the header are the links 'Home', 'Create Idea', and 'List Ideas'. On the right is a red-outlined button labeled 'Logout'. The main content area has a light gray background with the text 'Welcome konzept!' in a large, bold font. Below this, in a smaller font, is the text 'This is just a Prototype - so please be aware of that.' The rest of the page is white.

Quelle: Eigene Darstellung

Abbildung 12: GUI-Konzept - Idee erstellen



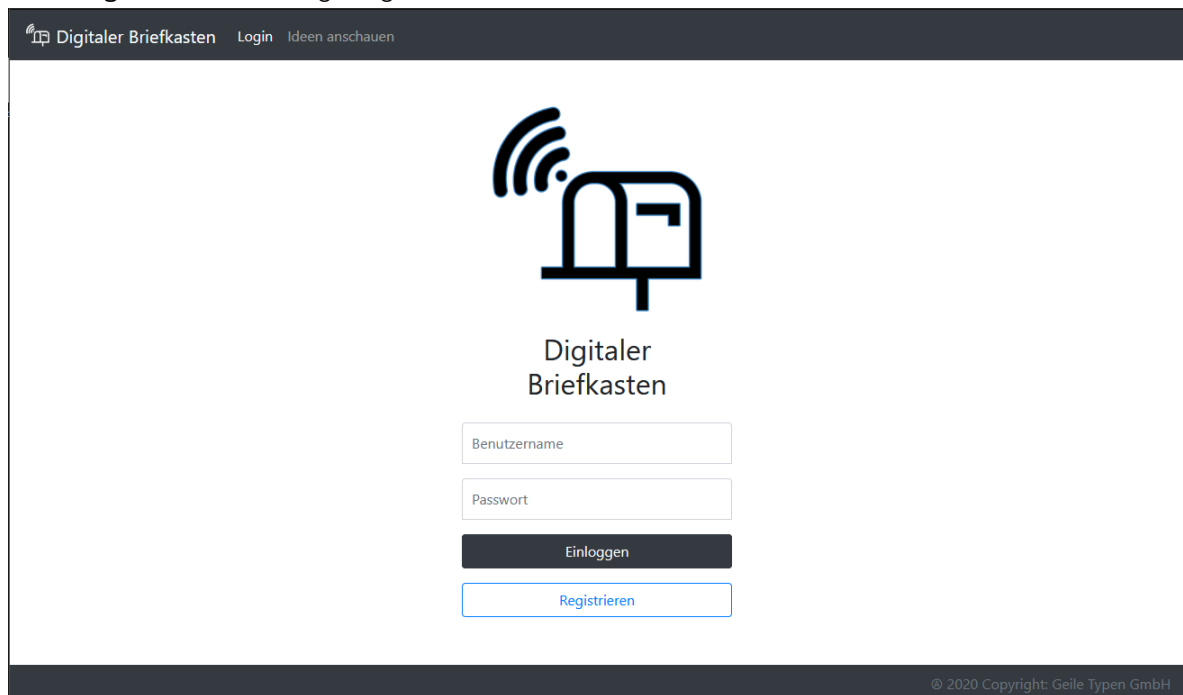
The image shows a web application interface for creating a new idea. At the top, there is a dark navigation bar with the text 'Digitaler Briefkasten' and a logo, followed by links 'Home', 'Create Idea', and 'List Ideas'. A 'Logout' button is located on the right side of the navigation bar. The main content area is titled 'Create new Idea'. It contains two input fields: 'title' and 'description'. Below these fields is a dark button labeled 'Create Draft'.

Quelle: Eigene Darstellung

Anhang 2.2 Umsetzung

Die im folgenden dargestellten GUI Bestandteile stellen die wichtigsten Teile der Oberfläche dar. Auf die Abbildung aller Bestandteile wurde aufgrund der zu großen Menge, zur Wahrung der Übersichtlichkeit, verzichtet.

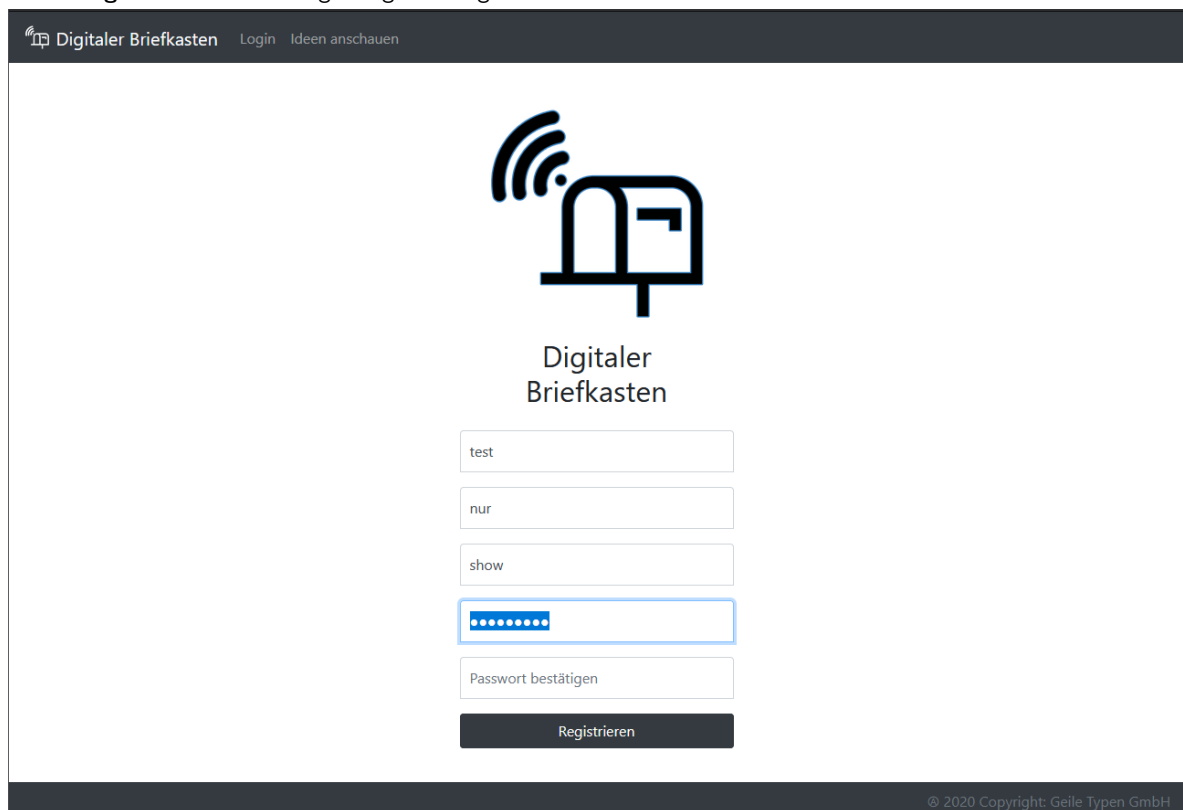
Abbildung 13: GUI-Umsetzung - Login



The screenshot displays a web application interface for 'Digitaler Briefkasten'. At the top, a dark navigation bar contains a home icon, the text 'Digitaler Briefkasten', and links for 'Login' and 'Ideen anschauen'. The main content area features a large icon of a mailbox with signal waves above it, followed by the title 'Digitaler Briefkasten'. Below the title are two input fields labeled 'Benutzername' and 'Passwort'. Underneath these fields are two buttons: a dark 'Einloggen' button and a light 'Registrieren' button. The footer of the page shows the copyright notice '© 2020 Copyright: Geile Typen GmbH'.

Quelle: Eigene Darstellung

Abbildung 14: GUI-Umsetzung - Registrierung



The screenshot shows a web application interface for 'Digitaler Briefkasten'. At the top, a dark navigation bar contains a logo, the text 'Digitaler Briefkasten', and links for 'Login' and 'Ideen anschauen'. The main content area features a large icon of a mailbox with signal waves. Below the icon, the text 'Digitaler Briefkasten' is displayed. The registration form consists of several input fields: a text field with 'test', a text field with 'nur', a text field with 'show', a password field with a blue border and a blue dot mask, a text field with 'Passwort bestätigen', and a dark 'Registrieren' button. A dark footer bar at the bottom right contains the copyright notice '© 2020 Copyright: Geile Typen GmbH'.

Digitaler Briefkasten

test

nur

show

.....

Passwort bestätigen

Registrieren

© 2020 Copyright: Geile Typen GmbH

Quelle: Eigene Darstellung

Abbildung 15: GUI-Umsetzung - Ideen

Digitaler Briefkasten
 Home Idee erstellen ▾ Ideen anschauen
 Logout

Deine nicht eingereichten Ideen

	Typ	Titel	Beschreibung	Ersteller	Erstellungsdatum	Status
⋮	💡	Geheime Weltherrschaft	Na was denn sonst	Umse tzung	20.05.2020	❓ Nicht eingereicht

Produkt-Ideen

	Titel	Beschreibung	Ersteller	Erstellungsdatum	Status	Status Begründung
⋮	Eine weitere	Geheimer Test	Umse tzung	20.05.2020	🕒 auf Bewertung wartend	

Interne-Ideen

	Titel	Beschreibung	Ersteller	Erstellungsdatum	Status	Status Begründung
⋮	Beispiel	Test	Umse tzung	20.05.2020	🕒 auf Bewertung wartend	

[Nutzen, Umsetzung, Rolle: Nutzer] [Kontaktformular](#)
© 2020 Copyright: Geile Typen GmbH

Quelle: Eigene Darstellung

Abbildung 16: GUI-Umsetzung - Idee erstellen

The screenshot shows a web application interface for creating a new product idea. The header bar is dark grey and contains the logo 'Digitaler Briefkasten', navigation links 'Home', 'Idee erstellen', and 'Ideen anschauen', and a 'Logout' button. The main content area is white and titled 'Neue Produktidee erstellen'. It contains several input fields: 'Titel', 'Beschreibung', 'Zielgruppe auswählen:', 'Vertriebskanal auswählen:', 'Produktsparte auswählen:' (with a dropdown menu showing 'KFZ'), a checkbox 'Existiert bereits Vergleichbares (bei der Konkurrenz)?', and three 'Vorteil' input fields under the heading 'Bis zu drei Vorteile eingeben:'. A dark grey button labeled 'Produkt-Idee erstellen' is at the bottom. The footer is dark grey and contains user information '[Nutzer: Umsetzung , Rolle: Nutzer]', a 'Kontaktformular' link, and a copyright notice '© 2020 Copyright: Geile Typen GmbH'.

Digitaler Briefkasten Home Idee erstellen Ideen anschauen Logout

Neue Produktidee erstellen

Titel

Beschreibung

Zielgruppe auswählen:

Vertriebskanal auswählen:

Produktsparte auswählen:

KFZ

☐ Existiert bereits Vergleichbares (bei der Konkurrenz)?

Bis zu drei Vorteile eingeben:

Vorteil

Vorteil

Vorteil

Produkt-Idee erstellen

[Nutzer: Umsetzung , Rolle: Nutzer] [Kontaktformular](#) © 2020 Copyright: Geile Typen GmbH

Quelle: Eigene Darstellung

Abbildung 17: GUI-Umsetzung - Idee ansehen

Digitaler Briefkasten Home Idee erstellen Ideen anschauen Logout

Geheime Weltherrschaft

Titel	Geheime Weltherrschaft
Beschreibung	Na was denn sonst
Produktlinie	KFZ
Vorteile	Nenene
Zielgruppen	Paare
Vertriebskanäle	Versicherungsmakler
Existiert bereits Vergleichbares?	<input checked="" type="checkbox"/>
Ersteller	Umsetzung
Erstellungsdatum	20.05.2020
Status	Nicht eingereicht
Status-Begründung	

[Nutzer: Umsetzung , Rolle: Nutzer] [Kontaktformular](#) © 2020 Copyright: Geile Typen GmbH

Quelle: Eigene Darstellung

Abbildung 18: GUI-Umsetzung - Admin Ansicht

The screenshot shows the Admin View of the 'Digitaler Briefkasten' application. The top navigation bar includes a home icon, the application name 'Digitaler Briefkasten', and links for 'Adminpanel', 'Idee erstellen', 'Ideen anschauen', and a 'Logout' button. The main content area features a large 'Willkommen admin!' greeting. Below this, there are several interactive elements: a link to 'Alle registrierten User ansehen', a section for 'Fachspezialisten anlegen' which contains input fields for 'Username', 'Vorname', 'Nachname', 'Passwort', and 'Passwort bestätigen', a 'Fachgebiet auswählen' dropdown, and a 'Spezialist anlegen' button. At the bottom of the main area, there are links to 'Produktparte anlegen', 'Handlungsfeld anlegen', 'Zielgruppe anlegen', and 'Vertriebskanal anlegen'. The footer displays user information '[Nutzer: admin , Rolle: Administrator]', a 'Kontaktformular' link, and copyright information '© 2020 Copyright: Geile Typen GmbH'.

Digitaler Briefkasten Adminpanel Idee erstellen Ideen anschauen Logout

Willkommen admin!

Alle registrierten User ansehen

Fachspezialisten anlegen

Username

Vorname

Nachname

Passwort

Passwort bestätigen

Fachgebiet auswählen:

Spezialist anlegen

Produktparte anlegen

Handlungsfeld anlegen

Zielgruppe anlegen

Vertriebskanal anlegen

[Nutzer: admin , Rolle: Administrator] Kontaktformular © 2020 Copyright: Geile Typen GmbH

Quelle: Eigene Darstellung

Abbildung 19: GUI-Umsetzung - Spezialist Ansicht

The screenshot displays a web application interface for a 'Spezialist' (Specialist) view. At the top, a dark navigation bar contains the text 'Digitaler Briefkasten' with a mail icon, followed by links: 'Ideen bewerten', 'Idee erstellen' (with a dropdown arrow), and 'Ideen anschauen'. A red 'Logout' button is in the top right corner. Below the navigation bar, a large light gray banner displays the welcome message 'Willkommen SpeziusMaximus_KFZ!'. The main content area is divided into two sections. The first section, 'Anstehende Entscheidungen', is marked with a key icon and contains a table with one row of data. The second section, 'Ideen im Speicher', is marked with a folder icon and contains an empty table. The footer of the page shows the user's role as 'Spezialist' and a copyright notice for 'Geile Typen GmbH'.

Navigation: Digitaler Briefkasten | Ideen bewerten | Idee erstellen ▾ | Ideen anschauen | Logout

Willkommen SpeziusMaximus_KFZ!

Anstehende Entscheidungen

	Typ	Titel	Beschreibung	Ersteller	Erstellungsdatum	Status
⋮	💡	Eine weitere	Geheimer Test	Umse tzung	20.05.2020	🕒 auf Bewertung wartend

Ideen im Speicher

	Typ	Titel	Beschreibung	Ersteller	Erstellungsdatum	Status
--	-----	-------	--------------	-----------	------------------	--------

[Nutzen: SpeziusMaximus_KFZ, Rolle: Spezialist] | [Kontaktformular](#) | © 2020 Copyright: Geile Typen GmbH

Quelle: Eigene Darstellung

Anhang 3 Projektplanung

Anhang 3.1 Projektstrukturplan

Abbildung 20: Projektstrukturplan



Quelle: Eigene Darstellung

	Anforderung	Umsetzung
Muss	Noch nicht registrierte Mitarbeiter können sich am System registrieren	Umgesetzt
Muss	Registrierte Mitarbeiter können sich am System anmelden	Umgesetzt
Muss	Registrierte Mitarbeiter können neue Ideen erfassen	Umgesetzt
Muss	Registrierte Mitarbeiter können sich eine Liste ihrer eingereichten Ideen anzeigen lassen	Umgesetzt
Muss	Registrierte Mitarbeiter können ihre Ideen solange bearbeiten oder auch löschen solange dieses noch nicht zur Bewertung an einen Fachspezialisten übergeben wurden.	Umgesetzt
Muss	Nicht registrierte Mitarbeiter können vorhandene Ideen lesen, sich eine Übersicht der Ideen anzeigen lassen und die Übersicht filtern	Umgesetzt
Muss	Diese Funktionen stehen auch registrierten Mitarbeitern zur Verfügung	Umgesetzt
Muss	Neue Ideen werden Fachspezialisten zur Bewertung zugeordnet	Umgesetzt
Muss	Die Zuordnung erfolgt automatisch sobald die Idee vom registrierten Mitarbeiter zur Bewertung eingereicht wurde	Umgesetzt
Muss	Fachspezialisten können eine Idee entweder annehmen, ablehnen oder für einen späteren Zeitpunkt in einen sog. Ideenspeicher überführen / sie aus dem Ideenspeicher zurückholen	Umgesetzt
Muss	Fachspezialisten begründen ihre Entscheidung transparent und für alle sichtbar in der Anwendung	Umgesetzt
Muss	Fachspezialisten können ihnen zugewiesene Ideen in einer Liste sehen und diese Liste filtern	Umgesetzt

Quellenverzeichnis

Internetquellen

Heuermann, Christian (2020). *Projektvorgaben Softwareprojekt - Meine Idee Initiative*.

Horn, Torsten (2007). *Vererbung und Polymorphie mit relationalen Datenbanken*. URL: <https://www.torsten-horn.de/techdocs/sql-vererbung.htm> (besucht am 21. Mai 2020).