IPA 2024  
Abschlussarbeit

OVWEB – Stellvertreter definieren

## 3. Mai 2024

|  |  |
| --- | --- |
| Kandidatin & Autorin: | Mia Justine Rascher |
| Firma: | Generali Versicherungen AG |
| Abteilung: | IT Apprentices |
| Berufsfachschule: | Technische Berufsschule Zürich |
| Verantwortliche Fachkraft: | Roman Babenko |
| Hauptexperte: | Janes Thomas |
| Nebenexperte: | Livio Ambrogini |
| Fachrichtung: | Applikationsentwicklung |
| Projektvorgehensmodell: | IPERKA |

# Dokumentinformationen

## Versionsverwaltung

### Vorlage = Basisversion der Dokumentationsvorlage

i.A. = in Arbeit

V = Freigegebene (geprüfte) Version

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Datum | Autorin | Beschreibung |
| Vorlage | 18. April 2024 | Mia Rascher | Erstellen der Dokumentationsvorlage nach den Generali Vorgaben. |
| i.A. | 22. April 2024 | Mia Rascher | Hinzufügen des Administrativen Teils. Start der Phase Informieren |
| i.A. | 23. April 2024 | Mia Rascher | Weiterarbeit an der Phase Informieren.  Start der Phase Planen. |
| i.A. | 24. April 2024 | Mia Rascher | Weiterarbeit an der Phase Planen. Letzte Arbeit am Administrationsteil. |
| i.A. | 25. April 2024 | Mia Rascher | Abschliessen der Phase Planen und Entscheiden. Einstieg in die Realisationsphase. |
| i.A. | 26. April 2024 | Mia Rascher | Arbeitsjournal |
| i.A. | 29. April 2024 | Mia Rascher | Arbeitsjournal |
| i.A. | 30. April 2024 | Mia Rascher | Arbeitsjournal |
| i.A. | 2. Mai 2024 | Mia Rascher |  |
| i.A. | 3. Mai 2024 | Mia Rascher |  |
| V | 6. Mai 2024 | Mia Rascher |  |

## Informationen zum Verzeichnis

### Glossar = Die Begriffe werden beim ersten Vorkommnis in folgender Farbe (#0000FF) dargestellt.

Inhalt

[24. April 2024 1](#_Toc164864706)

[Dokumentinformationen 2](#_Toc164864707)

[Versionsverwaltung 2](#_Toc164864708)

[Vorlage = Basisversion der Dokumentationsvorlage 2](#_Toc164864709)

[i.A. = in Arbeit 2](#_Toc164864710)

[V = Freigegebene (geprüfte) Version 2](#_Toc164864711)

[Version 2](#_Toc164864712)

[Datum 2](#_Toc164864713)

[Autorin 2](#_Toc164864714)

[Beschreibung 2](#_Toc164864715)

[Informationen zum Verzeichnis 2](#_Toc164864716)

[Glossar = Die Begriffe werden beim ersten Vorkommnis in folgender Farbe (#0000FF) dargestellt. 2](#_Toc164864717)

[Teil 1 – Administrativer Teil 6](#_Toc164864718)

[1. IPA-Projektorganisation 6](#_Toc164864719)

[2. Organisation der IPA-Ergebnisse 7](#_Toc164864720)

[2.1 Datensicherung 7](#_Toc164864721)

[3. Deklaration der benützten Standards 7](#_Toc164864722)

[3.1 Frontend 7](#_Toc164864723)

[Name 7](#_Toc164864724)

[Einsatz 7](#_Toc164864725)

[3.2 Backend 7](#_Toc164864726)

[Name 7](#_Toc164864727)

[Einsatz 7](#_Toc164864728)

[3.3 Daten Layer 7](#_Toc164864729)

[Name 7](#_Toc164864730)

[Einsatz 7](#_Toc164864731)

[3.4 Identity Provider 7](#_Toc164864732)

[Name 7](#_Toc164864733)

[Einsatz 7](#_Toc164864734)

[3.5 Entwicklungsumgebung 8](#_Toc164864735)

[Name 8](#_Toc164864736)

[Einsatz 8](#_Toc164864737)

[3.6 UI/UX Design 8](#_Toc164864738)

[Name 8](#_Toc164864739)

[Einsatz 8](#_Toc164864740)

[3.7 Dokumentation und Administration 8](#_Toc164864741)

[Name 8](#_Toc164864742)

[Einsatz 8](#_Toc164864743)

[3.8 Projektmanagment 8](#_Toc164864744)

[Name 8](#_Toc164864745)

[Einsatz 8](#_Toc164864746)

[4. Zeitplan 8](#_Toc164864747)

[5. Arbeitsjournal 8](#_Toc164864748)

[5.1 Tag 1: Montag, 22. April 2024 8](#_Toc164864749)

[**Tag** 8](#_Toc164864750)

[5.2 Tag 2: Dienstag, 23. April 2024 9](#_Toc164864751)

[**Tag** 9](#_Toc164864752)

[5.3 Tag 3: Mittwoch, 24. April 2024 9](#_Toc164864753)

[**Tag** 10](#_Toc164864754)

[5.4 Tag 4: Donnerstag, 25. April 2024 10](#_Toc164864755)

[**Tag** 10](#_Toc164864756)

[5.5 Tag 5: Freitag, 26. April 2024 10](#_Toc164864757)

[**Tag** 10](#_Toc164864758)

[5.6 Tag 6: Montag, 29. April 2024 11](#_Toc164864759)

[**Tag** 11](#_Toc164864760)

[5.7 Tag 7: Dienstag, 30. April 2024 11](#_Toc164864761)

[**Tag** 11](#_Toc164864762)

[5.8 Tag 8: Donnerstag, 2. Mai 2024 12](#_Toc164864763)

[**Tag** 12](#_Toc164864764)

[5.9 Tag 9: Freitag, 3. Mai 2024 12](#_Toc164864765)

[**Tag** 12](#_Toc164864766)

[5.10 Tag 10: Montag, 6. Mai 2024 13](#_Toc164864767)

[**Tag** 13](#_Toc164864768)

[Teil 2 – Individueller praktischer Teil 14](#_Toc164864769)

[6. Kurzfassung 14](#_Toc164864770)

[7. Projektvorgehen 14](#_Toc164864771)

[7.1 Beschreibungen der Projektmethoden 14](#_Toc164864772)

[7.2 Entscheidung 14](#_Toc164864773)

[**Kriterium** 14](#_Toc164864774)

[**IPERKA** 14](#_Toc164864775)

[**Scrum** 14](#_Toc164864776)

[7.1 Informieren 14](#_Toc164864777)

[7.2 Planen 14](#_Toc164864778)

[7.3 Entscheiden 15](#_Toc164864779)

[7.4 Realisieren 15](#_Toc164864780)

[7.5 Kontrollieren 15](#_Toc164864781)

[7.6 Auswerten 15](#_Toc164864782)

[8. Einleitung 16](#_Toc164864783)

[9. Informieren 16](#_Toc164864784)

[9.1 Ziele der Aufgabenstellung 16](#_Toc164864785)

[9.2 Vorgaben und Anforderungen 16](#_Toc164864786)

[9.3 Ist-Analyse 16](#_Toc164864787)

[9.4 User Stories und Akzeptanzkriterien 19](#_Toc164864788)

[**Nummer** 19](#_Toc164864789)

[**Story** 19](#_Toc164864790)

[**Akzeptanzkriterien** 19](#_Toc164864791)

[10. Planen 20](#_Toc164864792)

[10.1 Realisierungskonzept 20](#_Toc164864793)

[10.2 GUI-Design 22](#_Toc164864794)

[10.3 Aktivitätsdiagramme 24](#_Toc164864795)

[10.5 Klassendiagramm 24](#_Toc164864796)

[10.6 Datenmodell 24](#_Toc164864797)

[10.7 Testkonzept 24](#_Toc164864798)

[11. Entscheiden 24](#_Toc164864799)

[11.1 Entscheidungsmatrix 24](#_Toc164864800)

[12. Realisieren 24](#_Toc164864801)

[13. Kontrollieren 24](#_Toc164864802)

[13.1 Testkonzept 24](#_Toc164864803)

[13.2 Testprotokoll 24](#_Toc164864804)

[14. Auswerten 24](#_Toc164864805)

[14.1 Vergleich Ist / Soll 24](#_Toc164864806)

[14.2 Fazit 24](#_Toc164864807)

[14.3 Schlussreflexion 24](#_Toc164864808)

[15. Danksagung 24](#_Toc164864809)

[16. Glossar 25](#_Toc164864810)

[Begriff 25](#_Toc164864811)

[Definition 25](#_Toc164864812)

[17. Literaturverzeichnis 25](#_Toc164864813)

# Teil 1 – Administrativer Teil

## IPA-Projektorganisation

**Lehrbetrieb und Durchführungsort:**  
Generali Personenversicherungen AG  
Soodmattenstrasse 10  
8134 Adliswil  
058 472 44 44

**Kandidatin:**  
Mia Justine Rascher  
Juchstrasse 29  
8173 Neerach  
079 229 14 85  
mia.rascher@generali.com  
raschermia@outlook.com

**Auftraggeber:**Daniel Sager  
Generali Personenversicherungen AG  
Soodmattenstrasse 10  
8134 Adliswil  
daniel.sager@generali.com

**Berufsbildner / Lehrfirma:**Roger Sorg  
Generali Personenversicherungen AG  
Soodmattenstrasse 10  
8134 Adliswil  
079 779 53 69  
roger.sorg@generali.com

**Verantwortliche Fachkraft:**Roman Babenko  
Generali Personenversicherungen AG  
Soodmattenstrasse 10  
8134 Adliswil  
roman.babenko@generali.com

**Hauptexperte:**  
Janes Thomas  
janes@janesthomas.ch  
079 295 82 88

**Nebenexperte:**  
Livio Ambrogini  
l.ambrogini@hotmail.com  
078 976 99 52

## Organisation der IPA-Ergebnisse

### 2.1 Datensicherung

Um die vollständige Datensicherung zu gewährleisten, verwende ich bei meiner IPA unterschiedliche GitHub Repositorien.

#### 2.1.1 Code

Der gesamte Code wurde auf einem eigenständigen Branch «ipa-stellvertreter-definieren» auf dem Monorepo «insurance-offer-management» auf GitHub gepusht.

Link zum Branch: [IPA-Stellvertreter-Definieren](https://github.com/IT-Apprentices/insurance-offer-management/tree/ipa-stellvertreter-definieren)

Folgend findet sich ein Verlauf der Git Commits während der 10 Tage (Commit Logs):

#### 2.1.2 Dokumentation

Meine Dokumentation habe ich auf einem persönlichen GitHub Repository täglich gesichert mit einer aktuellen Version, sowie einer Versionierungs-Version, welche in einem separaten Ordner gespeichert wurde.

Link zum Repository: [IPA Stellverterter Definieren](https://github.com/meluetta/IPA.git)

Folgend findet sich ein Verlauf der Git Commits während der 10 Tage (Commit Logs):

## Deklaration der benützten Standards

[Begründung wieso in der Generali diese Technologien verwendet werden.]

Im Entwickler-Team werden die folgenden Technologien verwendet:

### 3.1 Frontend

|  |  |
| --- | --- |
| Name | Einsatz |
| NextJS 14 | Frontend Technologie |
| Shadcn | Komponenten Design Bibliothek |
| Typescript | Frontendsprache |
| Jest | Frontend Testing |
| Storybook | Frontend Dokumentation |

### 3.2 Backend

|  |  |
| --- | --- |
| Name | Einsatz |
| Springboot | Backend Technologie |
| Java | Backendsprache |
| Lombok | Annotations für Automatisierung |
| JUnit | Testing |

### 3.3 Daten Layer

|  |  |
| --- | --- |
| Name | Einsatz |
| PostgreSQL | Datenbank |
| Flyway | Datenbankversionierungstool |

### 3.4 Identity Provider

|  |  |
| --- | --- |
| Name | Einsatz |
| OpenID Connect | Identitätsprotokoll |
| Keycloak | Identitätsmanagment |

### 3.5 Entwicklungsumgebung

|  |  |
| --- | --- |
| Name | Einsatz |
| JetBrains IntelliJ IDEA | Entwicklungsumgebung |
| Docker | Containerisierungsplattform |
| Windows 11 | Operating System |

### 3.6 UI/UX Design

|  |  |
| --- | --- |
| Name | Einsatz |
| Figma | Design Tool |
| Generali Styleguide | Styleguide |

### 3.7 Dokumentation und Administration

|  |  |
| --- | --- |
| Name | Einsatz |
| Microsoft Office | Dokumentation |
| Draw.io und IntelliJ IDEA | Diagramme & Skizzen |
| Github | Versionierung der Dokumentation und Backups |

### 3.8 Projektmanagment

|  |  |
| --- | --- |
| Name | Einsatz |
| Jira | Projektmanagment-Tool |
| Github | Versionskontroll- und Kollaborationsplattform |

## Zeitplan

## Arbeitsjournal

### 5.1 Tag 1: Montag, 22. April 2024

|  |  |
| --- | --- |
| Tag | 1 |
| Datum | 22.04.2024 |
| Aktivität | Anforderungen studieren, Zeitplan erstellen, Ist-Zustand Analyse |
| Erfolge | Zeitplan fertig gestellt, grosser Fortschritt im ersten Teil der Dokumentation. |
| Misserfolge | Etwas Mühe mit der Ist-Analyse und der Korrekten Dokumentation, Arbeitsjournal etwas knapp |
| Hilfestellungen | - |
| Reflexion | Ich habe zu spät mit dem Arbeitsjournal angefangen und muss mich morgen früher daran erinnern und zeitiger damit anfangen. |
| Weitere Lösungsvarianten | Eine Erinnerung im Kalender setzen für die eingeplanten 30 Minuten Arbeitsjournal. |
| Sollzeit | 8,2h |
| Istzeit | 8,1h |

### 5.2 Tag 2: Dienstag, 23. April 2024

|  |  |
| --- | --- |
| Tag | 2 |
| Datum | 23.04.2024 |
| Aktivität | Ist-Zustand Analyse, User Stories und Akzeptanzkriterien definieren, Soll-Zustand definieren, Diagramme erstellen |
| Erfolge | Ich bin heute einigermassen gut vorangekommen. Wir hatten ein gutes erstes Expertengespräch. |
| Misserfolge | Ich habe leider etwas wenig Zeit für «Soll-Zustand definieren» eingeplant. Ich werde morgen noch etwas mehr daran weiterarbeiten, da meine Diagramme auch noch nicht fertig sind. |
| Hilfestellungen | Meine Fragen an Herr Janes Thomas wurden heute aufgeklärt. |
| Reflexion | Die Dokumentation erfordert sehr viel Arbeit, da mir manchmal die Worte fehlen, jedoch bin ich guter Dinge und komme gut voran. Nach dem Expertengespräch hatte ich keine Zeit mehr für die Diagramme, da ich noch einmal den Zeitplan überarbeitet habe, bevor ich ihn Herrn Thomas via E-Mail zugestellt habe. Ausserdem hatte ich heute mehr Zeit für das Arbeitsjournal und habe dieses zu meiner Zufriedenheit fertig gestellt. |
| Weitere Lösungsvarianten | Es gibt keine weiteren Lösungsvarianten für aktuelle Probleme. |
| Sollzeit | 8,2h |
| Istzeit | 8,2h |

### 5.3 Tag 3: Mittwoch, 24. April 2024

|  |  |
| --- | --- |
| Tag | 3 |
| Datum | 24.04.2024 |
| Aktivität | Diagramme erstellen, Mockups erstellen, Dokumentation administrativer Teil, Dokumentation Planen |
| Erfolge | Ich habe meine Mockups sowie Sequenzdiagramme fertiggestellt und die Entscheidungsmatrix für die Projektmethode erstellt, vervollständigt und hinzugefügt. |
| Misserfolge | Ich habe es nicht geschafft mein Datenbank Modell zu vervollständigen, die Testkonzepte zu erstellen und die Entscheidungsphase zu dokumentieren und abzuschliessen. Ich habe somit beide Meilensteine des heutigen Tages nicht erreicht und abgeschlossen. Mir fehlt noch einiges an erklärenden Beschreibungen für bestimmte Diagramme & Grafiken. |
| Hilfestellungen | Einer meiner Mitlernenden hat mir den Tipp gegeben für meine Sequenzdiagramme ChatGPT und PlantUML zu verwenden. Dies habe ich getan mich mit beiden Tools etwas herumprobiert. Ich habe ChatGPT nach dem Text für den PlantUML Editor gefragt und somit herausgefunden, wie es funktioniert.  Roman, mein Fachexperte, hat mich beim Sequenzdiagramm unterstützt, da ich unsicher war, wie genau ich die einzelnen Prozesse aufteilen und richtig darstellen kann. Dies hat mich gut vorangebracht und ich habe schnell verstanden, worauf ich achten muss und wie ich schnell Fortschritte mache. |
| Reflexion | Ich habe heute leider nicht alles geschafft, was ich geplant hatte. Dadurch komme ich bereits jetzt in einen Zeitdruck. Ich bin noch unsicher, wie genau ich dies wieder aufholen kann, aber ich bin dennoch mit meiner Arbeit zufrieden. Als ich bei einer Arbeit nicht weitergekommen bin, habe ich an einem anderen Ort weitergemacht und habe somit weniger Zeit verloren, wie wenn ich versucht hätte «den Knopf» sofort zu lösen. Ich habe zudem mit meinen Mockups ebenfalls für die Entscheidungsphase eine «Vorarbeit» geleistet. Da ich in der Phase «Entscheiden» der IPERKA-Projektmethode zwischen zwei Mockup-Varianten entscheide. Diese Varianten sind vorhanden und ich habe meine Entscheidung getroffen, jedoch fehlt noch die ausführliche Dokumentation davon. |
| Weitere Lösungsvarianten | Um meinen Zeitdruck etwas zu lösen, möchte ich morgen erneut mit Roman zusammensitzen und an meiner Planungsphase arbeiten. Ich habe eine «Blockade» in meinem Kopf und brauche einen Ratschlag für die Planung für das Testkonzept und das Datenmodell. |
| Sollzeit | 8,2h |
| Istzeit | 8,2h |

### 5.4 Tag 4: Donnerstag, 25. April 2024

|  |  |
| --- | --- |
| Tag | 4 |
| Datum | 25.04.2024 |
| Aktivität | Datenbank Modell erstellen, Testkonzept erstellen, Lösungsvariante festlegen, Backend Datenbank anpassen, Modell Anpassung, E-Mail-Benachrichtigung erstellen |
| Erfolge | Ich konnte heute einiges an Zeit aufholen. Morgens habe ich meine Testkonzepte geschrieben und bin relativ gut vorangekommen. Ich habe mein Datenmodell bzw. die Datenbankänderungen mit einem Text beschrieben. Vor dem Mittag habe ich meine Entscheidung in der Entscheidungsphase dokumentiert. Nach dem Mittag bin ich dann gestartet in die Realisationsphase. Ich bin gut vorangekommen und obwohl ich laut Zeitplan vier Stunden im Verzug bin. Ich konnte die Datenbank Anpassungen, inklusive Modell Anpassungen fliessend zusammen umsetzen und ich bin sehr glücklich mit dem Ergebnis. Dadurch, dass ich schnell und effizient arbeiten konnte, konnte ich eine Stunde an Arbeit wieder aufholen.  Zusätzlich hatte ich bei meinen Überlegungen für meine Datenbank eine Blockade, weswegen ich dann eine Tabelle auf dem Whiteboard bei uns im Büro gezeichnet habe. Dies hat mir sehr geholfen zu visualisieren, was genau ich beachten muss. |
| Misserfolge | Nachdem ich ca. eine Stunde an meinem Datenmodell gearbeitet habe, habe ich aus Versehen meinen Browser neugeladen, ohne zu speichern. Das hat mich sehr frustriert und ich habe überlegt, wie ich dies nun löse. Schliesslich habe ich mich für eine Datenkbankänderung in Textform entschieden, anstatt erneut 40 Minuten damit zu verbringen ein neues ERD zu machen. Nachdem ich die ersten beiden Teile des Realisierens «fertig» hatte, war noch nicht einmal eine Stunde vergangen. Jedoch hatte ich kleinere Syntax-Fehler und in meiner IDE und meinem Browser gab es einige kleine Probleme, durch welche ich 1,5 Stunden debuggen musste. Zusätzlich habe ich noch mit den E-Mail-Benachrichtigungen angefangen, wobei ich aber ebenfalls ein Problem mit den Credentials für die API hatte. |
| Hilfestellungen | Roman hat mir sehr beim Debuggen geholfen. Vier Augen sehen ein fehlendes Komma schneller als Zwei und ich wollte nicht noch mehr Zeit verlieren, wenn ich schon hinter meinem Zeitplan bin. Zusätzlich hat mich Roman bei meinen Problemen bezüglich des E-Mail Services unterstützt. Er hat getestet, ob das Problem bei mir lokal liegt oder ob bei ihm dasselbe Problem herrscht. Dadurch konnten wir nach einiger Zeit feststellen, dass das Problem bei den API-Credentials liegt und ein Tippfehler in diesen war. |
| Reflexion | Das mein Datenbankmodell nicht gespeichert wurde ärgert mich sehr, jedoch wurde mir dadurch direkt wieder bewusst, wie wichtig es ist, wichtige Dinge sofort und immer wieder zu speichern. Heute war ein sehr erfolgreicher Tag für mich und ich gehe mit einem guten Gefühl nachhause, weil ich stolz auf meine Arbeit sein kann. |
| Weitere Lösungsvarianten | Ich bin sehr zufrieden mit meinen heutigen Lösungsansätzen für die heutigen Schwierigkeiten. Ich bin gut vorangekommen und habe einige Fortschritte gemacht.  Etwas visuell vor mir zu haben und es mir nicht die ganze Zeit im Kopf merken zu müssen, wenn ich sonst schon im Stress bin, hilft mir sehr. Anbei mein beschriebenes Whiteboard -> |
| Sollzeit | 8,2h |
| Istzeit | 8,2h |

### 5.5 Tag 5: Freitag, 26. April 2024

|  |  |
| --- | --- |
| Tag | 5 |
| Datum | 26.04.2024 |
| Aktivität | E-Mail-Benachrichtigung erstellen, Rollenerweiterung, Tests erstellen |
| Erfolge | Ich bin gut vorangekommen und konnte die E-Mail-Benachrichtigungen, sowie die Rollenerweiterung abschliessen. Ebenfalls hat Roman mir dabei geholfen einen JUnit-Test für die E-Mail-Benachrichtigung zu schreiben und zu testen. |
| Misserfolge | Bei der Rollenerweiterung hatte ich einige Probleme und bin nicht so schnell vorangekommen, wie ich eigentlich wollte. Damit bin ich immer noch 3 Stunden hinter meinem Zeitplan. |
| Hilfestellungen | Roman hat mir gut bei der Rollenerweiterung geholfen und mir auch nochmal genau erklärt, wie die Details im JUnit Test funktionieren. |
| Reflexion | Im Großen und Ganzen war es ein erfolgreicher Tag, allerdings war ich heute etwas weniger produktiv als an Tag 4. Zum einen hatte ich etwas wenig Schlaf und zum anderen hatte ich dadurch lästige Kopfschmerzen. Ich hoffe ich kann am Wochenende etwas Energie tanken und am Montag wieder mit mehr Motivation weiterarbeiten. |
| Weitere Lösungsvarianten | Zu den Problemen die ich heute hatte gibt es keine weiteren Lösungsvarianten. |
| Sollzeit | 8,2h |
| Istzeit | 8,2h |

### 5.6 Tag 6: Montag, 29. April 2024

|  |  |
| --- | --- |
| Tag | 6 |
| Datum | 29.04.2024 |
| Aktivität | Anpassung Pendenzenliste, Tests erstellen |
| Erfolge | Ich bin heute einen guten Schritt weiter gekommen mit den Anpassungen in der Pendenzenliste. Zusätzlich existiert nun ein Test, welcher die Datenbank mit Daten befüllt, damit man meinen Task einfach und schnell testen kann. |
| Misserfolge | Es fehlen noch das Feature, bei welchem die Knöpfe verschwinden, wenn man sie einmal gedrückt hat. Zudem stimmen die Datensätze in der Pendenzenliste noch nicht, da noch nicht der Befugnis Geber noch der Stellvertreter sichtbar aufgeführt sind. Ich habe bereits mehr als doppelt so lange wie eingeplant an diesem Task gearbeitet und bin immer noch nicht fertig. |
| Hilfestellungen | Roman hat mir bei der Erkennung des Tasks im Frontend sehr viel geholfen, somit sind zum Beispiel die «Accept» und «Decline» Knöpfe nur noch bei meinem Task und nicht mehr bei den anderen sichtbar. |
| Reflexion | Ich wüsste nicht, wie ich es hätte schneller oder besser lösen können. Gerade dieser Teil meiner Aufgabe hat sehr viele knifflige Verschachtelungen zwischen Frontend, Backend und Datenbank und erfordert unglaublich viel Arbeit. Diese habe ich eindeutig unterschätzt und ich werde meinen Zeitplan vermutlich neu machen müssen. |
| Weitere Lösungsvarianten | Ein neuer Zeitplan ist die einzige mögliche neue Lösungsvariante. |
| Sollzeit | 8,2h |
| Istzeit | 8,2h |

### 5.7 Tag 7: Dienstag, 30. April 2024

|  |  |
| --- | --- |
| Tag | 7 |
| Datum | 30.04.2024 |
| Aktivität | Anpassung Pendenzenliste, Stellvertreter Verwaltung UI(StatusBox) |
| Erfolge | Ich konnte heute die Anpassung der Pendenzenliste abschliessen und habe das Grunddesign der Statusbox für die Stellvertreter Verwaltung abgeschlossen. |
| Misserfolge | Heute bin ich immer noch im Rückstand und hinter meiner Zeitplanung. |
| Hilfestellungen | Roman hatte mir geholfen die Funktionalität der Pendenzenliste zu finalisieren und hat mir einen «optimaleren» Weg für das Anzeigen der Knöpfe gezeigt. Dadurch sind die Knöpfe nur beim Stellvertreter sichtbar und nachdem der Status von «Pending» gewechselt wurde, sind sie nicht mehr sichtbar. |
| Reflexion | Ich bin gut vorangekommen, jedoch mache ich etwas langsam Fortschritte und mir fehlt noch einiges an der Dokumentation. Ich hoffe ich schaffe es in den nächsten 3 Tagen noch alles aufzubauen. |
| Weitere Lösungsvarianten | Aktuell gibt es keine anderen Lösungsvarianten. |
| Sollzeit | 8,2h |
| Istzeit | 8,2h |

### 5.8 Tag 8: Donnerstag, 2. Mai 2024

|  |  |
| --- | --- |
| Tag | 8 |
| Datum | 02.05.2024 |
| Aktivität | Stellvertreter Verwaltung UI (Statusbox und PopUp), Expertenbesuch |
| Erfolge | Ich konnte die Grundfunktion der Statusbox heute abschliessen und habe ebenfalls mit dem Pop-up angefangen. Für das Pop-up konnte ich einen Teil von Jonas (ein Mitlernender) verwenden und an meine Aufgabe anpassen. Dadurch konnte ich etwas Zeit sparen. |
| Misserfolge | Ich habe den Meilenstein des Realisierens heute nicht erreicht. Daher muss ich morgen noch schauen, wie viele Tests ich erstellen kann, bevor ich dann mein Testprotokoll sowie die Unit Tests durchführen kann. |
| Hilfestellungen | Roman hat mir den Link zum File von Jonas geschickt und gesagt ich kann mich an seinem Beispiel inspirieren lassen. |
| Reflexion | Ich habe nun nur noch 2 Tage vor mir und bin aktuell sehr im Zeitdruck. Ich darf mich nicht zu sehr stressen lassen und muss das Ziel im Auge behalten. Ansonsten schaffe ich meine Dokumentation nicht fertig. |
| Weitere Lösungsvarianten | Für den heutigen Tag kenne ich keine weiteren Lösungsvarianten. |
| Sollzeit | 8,2h |
| Istzeit | 8,2h |

### 5.9 Tag 9: Freitag, 3. Mai 2024

|  |  |
| --- | --- |
| Tag | 9 |
| Datum | 03.05.2024 |
| Aktivität | Stellvertreter UI Pop-up, Dokumentation Realisieren |
| Erfolge | Ich konnte heute die Hauptfunktionen meiner Aufgabe abschliessen und habe fast alles im Kapitel 12 Realisieren dokumentieren können. |
| Misserfolge | Mir fehlen immernoch die Punkte Kontrollieren und Auswerten, allerdings bin ich zuversichtlich, dass ich dies bis am Montagabend um 17Uhr schaffen werde. |
| Hilfestellungen | Roman hat mir geholfen, da ich beim Pop-up eine Komponente verwendet habe, welche wir bereits in einer anderen Form verwendet haben (DatePicker). Diese hatte als Anforderung eine Variabel mit dem Namen «birthdate» welche ich nicht habe und somit mussten wir dies anpassen. |
| Reflexion | Ich konnte heute einige Fortschritte machen und bin zuversichtlich, dass ich die IPA rechtzeitig fertigstellen und abgeben kann. |
| Weitere Lösungsvarianten | Aktuell gibt es keine anderen Lösungsvarianten. |
| Sollzeit | 8,2h |
| Istzeit | 8,2h |

### 5.10 Tag 10: Montag, 6. Mai 2024

|  |  |
| --- | --- |
| Tag | 10 |
| Datum |  |
| Aktivität |  |
| Erfolge |  |
| Misserfolge |  |
| Hilfestellungen |  |
| Reflexion |  |
| Weitere Lösungsvarianten |  |
| Sollzeit |  |
| Istzeit |  |

# Teil 2 – Individueller praktischer Teil

## Kurzfassung

## Projektvorgehen

Im folgenden Kapitel erkläre und begründe ich wieso ich mich für die Projektmethode IPERKA entschieden habe. Um dies zu veranschaulichen habe ich eine Entscheidungsmatrix gestaltet und zwischen IPERKA und Scrum entschieden.

### 7.1 Beschreibungen der Projektmethoden

Bei der IPERKA-Projektmethode geht es darum einen strukturierten Rahmen für eine saubere Durchführung zu schaffen. Sie gliedert sich in sechs Phasen: Informieren, Planen, Entscheiden, Realisieren, Kontrollieren und Auswerten. Es ist ein sehr flexibles Modell und findet Anwendung in unterschiedlichen Bereichen, von der Bildung bis zum Projektmanagement. Es wird besonders Wert auf eine systematische Herangehensweise gelegt und die Bedeutung einer sorgfältigen Vorbereitung, sowie einer abschliessenden Reflexion und Analyse des Endergebnisses.

Scrum ist ein agiles Framework, das Teams dabei unterstützt, komplexe Projekte in kurzen, iterativen Schritten umzusetzen. Es basiert auf regelmäßigen Meetings, klaren Rollen und einem transparenten Arbeitsprozess, um die Zusammenarbeit und Effizienz zu maximieren. Durch eine kontinuierliche Anpassung an Feedback und Veränderungen ermöglicht Scrum eine flexible und schnelle Reaktion auf Anforderungen während des gesamten Projektverlaufs.

### 7.2 Entscheidung

|  |  |  |
| --- | --- | --- |
| **Kriterium** | **IPERKA** | **Scrum** |
| Erfahrung | = | = |
| Kompatibilität | + | - |
| **Bewertung:** | **+** | **-** |

In den folgenden Unterkapiteln wird weiter erläutert, wie die Bewertungen begründet wurden.

#### 7.2.1 Erfahrung

In den letzten 3 Jahren im Betrieb haben wir mit Scrum gearbeitet. Ich konnte viel Erfahrung in der Organisation und Durchführung dieses Projektmanagements sammeln. IPERKA habe ich bereits in der Oberstufe kennengelernt und verwende es oft bei wichtigen Arbeiten, welche eine ausführliche Dokumentation erfordern.

#### 7.2.2 Kompatibilität

IPERKA ist einfacher umzusetzen in einem Projekt, in dem ich hauptsächlich allein arbeite. Mit Sprints und mehreren Meetings innerhalb dieser 10 Tage zu arbeiten wäre unpraktisch und würde mich in meinem Arbeitsprozess unterbrechen. IPERKA hilft mir in der Strukturierung meiner Dokumentation und meines Vorgehens und bietet mir einen klaren roten Faden.

### 7.3 IPERKA

Im folgenden Kapitel erkläre ich die einzelnen Bestandteile von IPERKA erneut im Detail.

#### 7.3.1 Informieren

In der Phase des Informierens geht es darum, das Projekt zu verstehen und mögliche Missverständnisse und Unklarheiten aufzuklären. Hierbei werden alle relevanten Informationen gesammelt, analysiert und strukturiert, um ein klares Bild der Aufgabenstellung der Projektziele zu erhalten.

#### 7.3.2 Planen

Wenn alle Informationen vorhanden und gesammelt sind, geht es in die nächste Phase über, die Planungsphase. Diese Phase dreht sich um das Entwickeln verschiedener Strategien für die komplexen Aufgaben und die Erstellung konkreter Konzepte und Herangehensweisen für die Umsetzung.

#### 7.3.3 Entscheiden

Nachdem alle Pläne, Konzepte und Strategien entwickelt wurden geht es in die Entscheidungsphase, in welcher die Arbeit der Planungsphase verglichen und abgewogen wird. Basierend auf einer sorgfältigen Bewertung wird der optimale Lösungsweg für die Projektumsetzung bestimmt.

#### 7.3.4 Realisieren

Der nächste Schritt ist die Realisierungsphase, in welcher die klaren Vorgaben und Entscheidungen umgesetzt werden. Während dieser Phase werden die geplanten Schritte ausgeführt und kontinuierlich dokumentiert, um den Fortschritt nachvollziehbar zu gestalten.

#### 7.3.5 Kontrollieren

Nach dem Abschluss der Umsetzung erfolgt die Überprüfung der Ergebnisse. Anhand der Zielsetzung und dem zuvor definierten Testkonzept, wird sichergestellt, ob alles wie geplant funktioniert hat und alle Ziele erreicht wurden.

#### 7.3.6 Auswerten

Anschliessend wird der gesamte Prozess ausgewertet. Diese Reflexion bezieht sich auf die Bewertung des erreichten Ergebnisses im Vergleich zum gesetzten Ziel. Die Erfahrungen, Probleme und das Verbesserungspotential des Projekts werden ebenfalls einbezogen.

## Einleitung

## Informieren

### 9.1 Ziele der Aufgabenstellung

Das Ziel der Aufgabe besteht darin, dem Berater, die Funktionalität der Bestimmung eines Stellvertreters zu bieten für den Fall, dass der Berater abwesend sein sollte. Dies inkludiert den Datenstamm-Kundendaten, mit welcher der Stellvertreter arbeiten können muss, ohne dass diese Daten mit seinen eigenen vermischt werden.

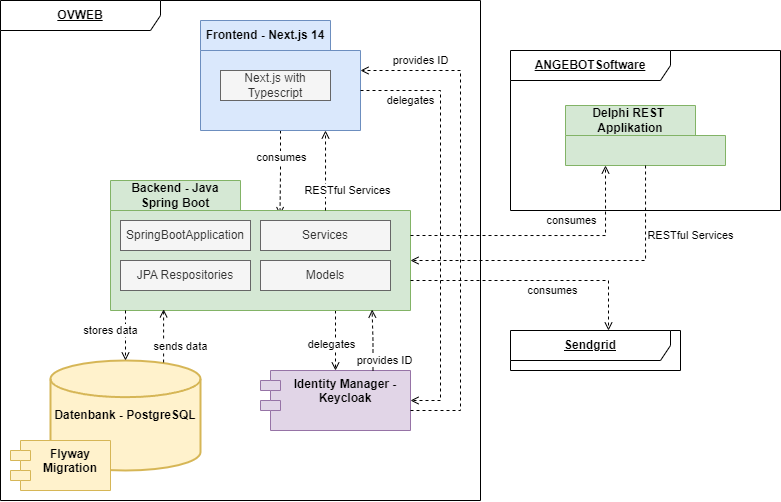
### 9.2 Vorgaben und Anforderungen

Wenn der Berater eine Anfrage an einen Stellvertreter schickt, muss dieser eine Benachrichtigung mit einem Link erhalten und diese Anfrage annehmen oder ablehnen können. Der Berater muss den Status dieser Anfrage auf einer Pendenzenliste einsehen können. Auf dieser ist ebenfalls ersichtlich, wer die Anfrage gestellt hat und wie lang diese dauert. Der Stellvertreter kann zwischen seinen eigenen Daten und denen des Beraters, welchen er vertritt, unterscheiden. Der Berater kann seinem Stellvertreter die Befugnis jederzeit wieder entziehen.

### 9.3 Ist-Analyse

#### 9.3.1 Überblick

Im untenstehenden Diagramm ist die OVWEB-Anwendung inklusive ihrer Abhängigkeiten klar und übersichtlich dargestellt.



#### 9.3.2 Frontend

Kommunikation zum Backend

Die Kommunikation zwischen dem Frontend und dem Backend erfolgt unter Benutzung des Client-API-Packages, welches automatisch von OpenAPI Maven Plugin basierend auf der Endpunktannotation im Backend generiert wird. Somit wird die typensichere Integration zwischen dem OVWEB Frontend und dem OVWEB Backend sichergestellt.

Server-Actions

Um mit der OVWEB-Backend-API zu interagieren, werden Server Actions im OVWEB Frontend verwendet. Diese ermöglichen es die serverseitige Programmierlogik direkt im Frontend zu verwenden. Dies macht die Entwicklung effizienter und flexibler. Ebenfalls können durch diese Technik Daten abgerufen, verarbeitet und an das Backend gesendet werden, ohne dass der Browser zwischenspeichern muss, was die Performance und die Sicherheit der Plattform verbessert.

Storybook

Storybook wird als eine zentrale Bibliothek für die UI-Komponenten und Seiten verwendet. In das Storybook werden sämtliche entwickelten Komponenten und Seiten eingetragen, sodass sie unabhängig vom Hauptsystem visualisiert werden können. Dies ermöglicht es, dass auch Personen ohne technischen Hintergrund die UI-Elemente betrachten und beurteilen können, ohne dass die gesamte Applikation gestartet werden muss.

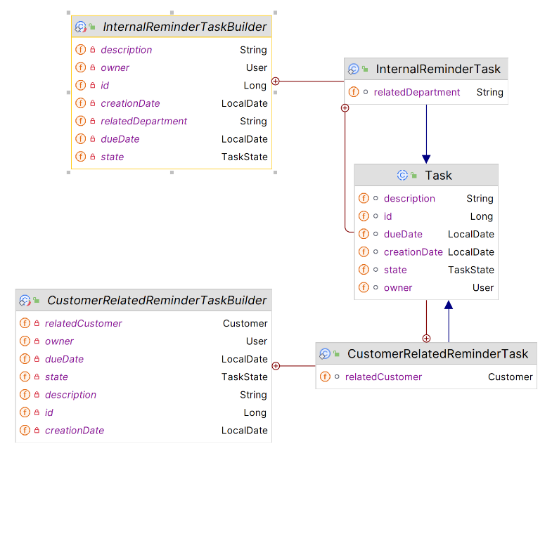
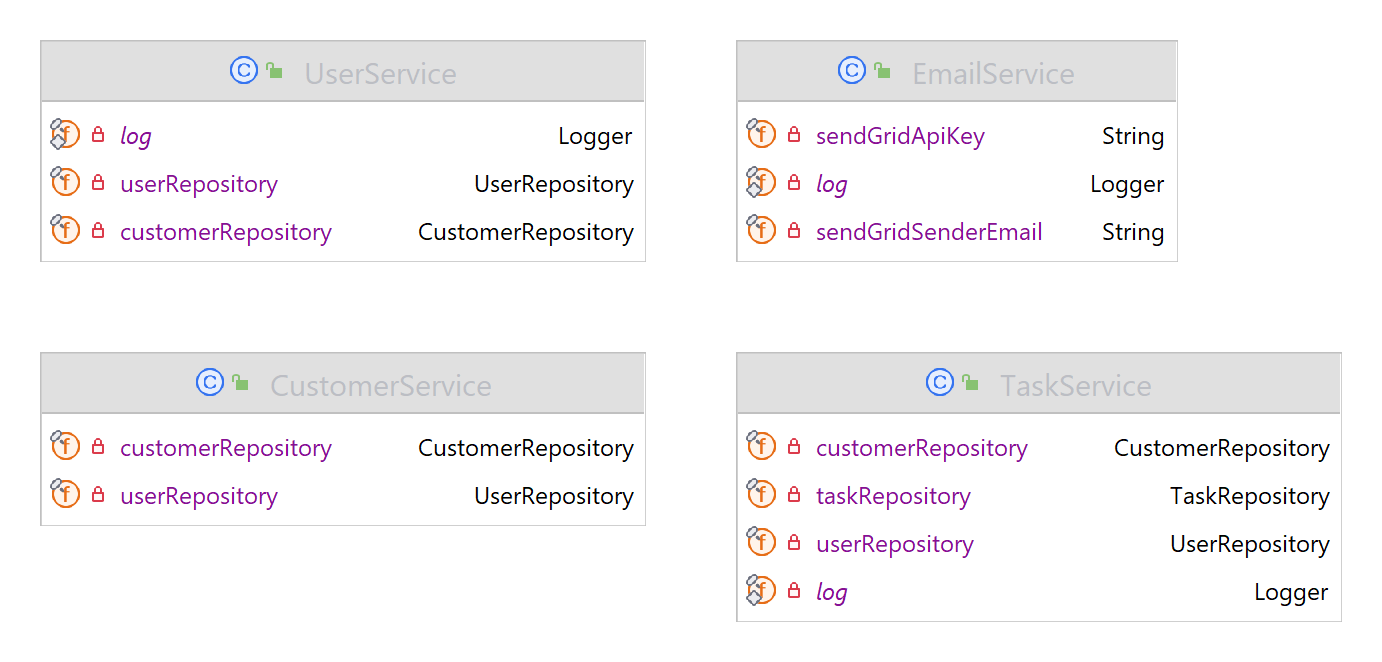
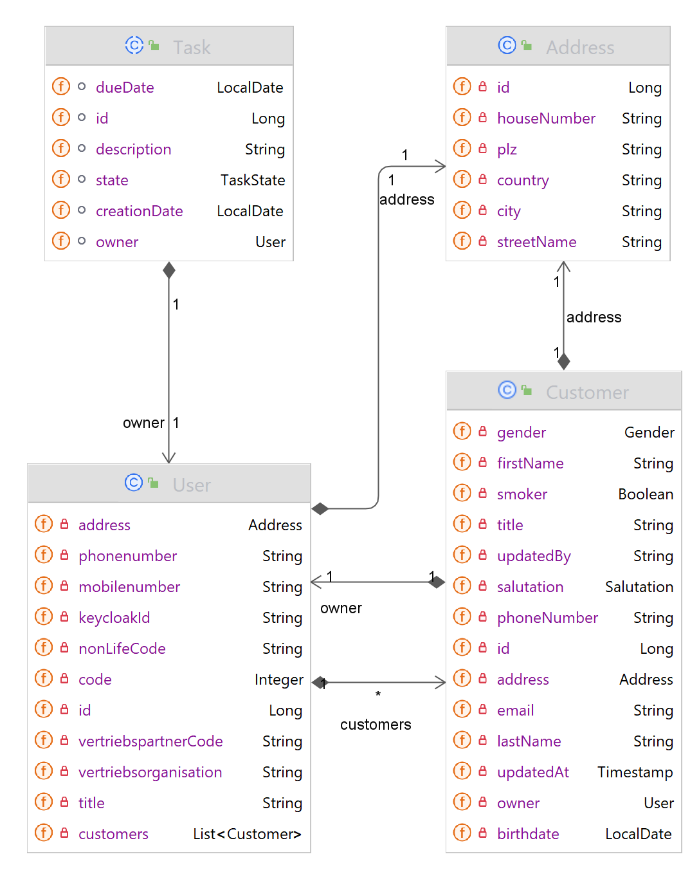
Ein Bild, das Text, Screenshot, Zahl, Software enthält.

Automatisch generierte Beschreibung

#### 9.3.3 Backend

Das Backend wurde mit dem Jave-Framework Springboot realisiert. User, Customer, Taks und Adresse sind im Model beinhaltet. Für die Datenbank wird PostgreSQL als Query-Sprache verwendet. Die Anbindung dieser Datenbank läuft mit der Hilfe von einem lokalen Docker-Container und wird durch Flyway versioniert.

#### 9.3.4 Klassendiagramme

Die folgenden Darstellungen zeigen das Klassendiagramm mit den einzelnen Teilen der Applikation. Ebenfalls dargestellt wird die Struktur rund um die Task-Tabelle, da diese für meine funktionalen Anforderungen essentiell sind.

#### 9.3.5 Datenbankdiagramm

Die Datenbank wird durch Flyway verwaltet und versioniert. Sie beinhaltet die gleichen Entitäten wie die einzelnen Models im Backend.



### 9.4 User Stories und Akzeptanzkriterien

#### 9.4.1 Rollen

#### 9.4.2 Anforderungen

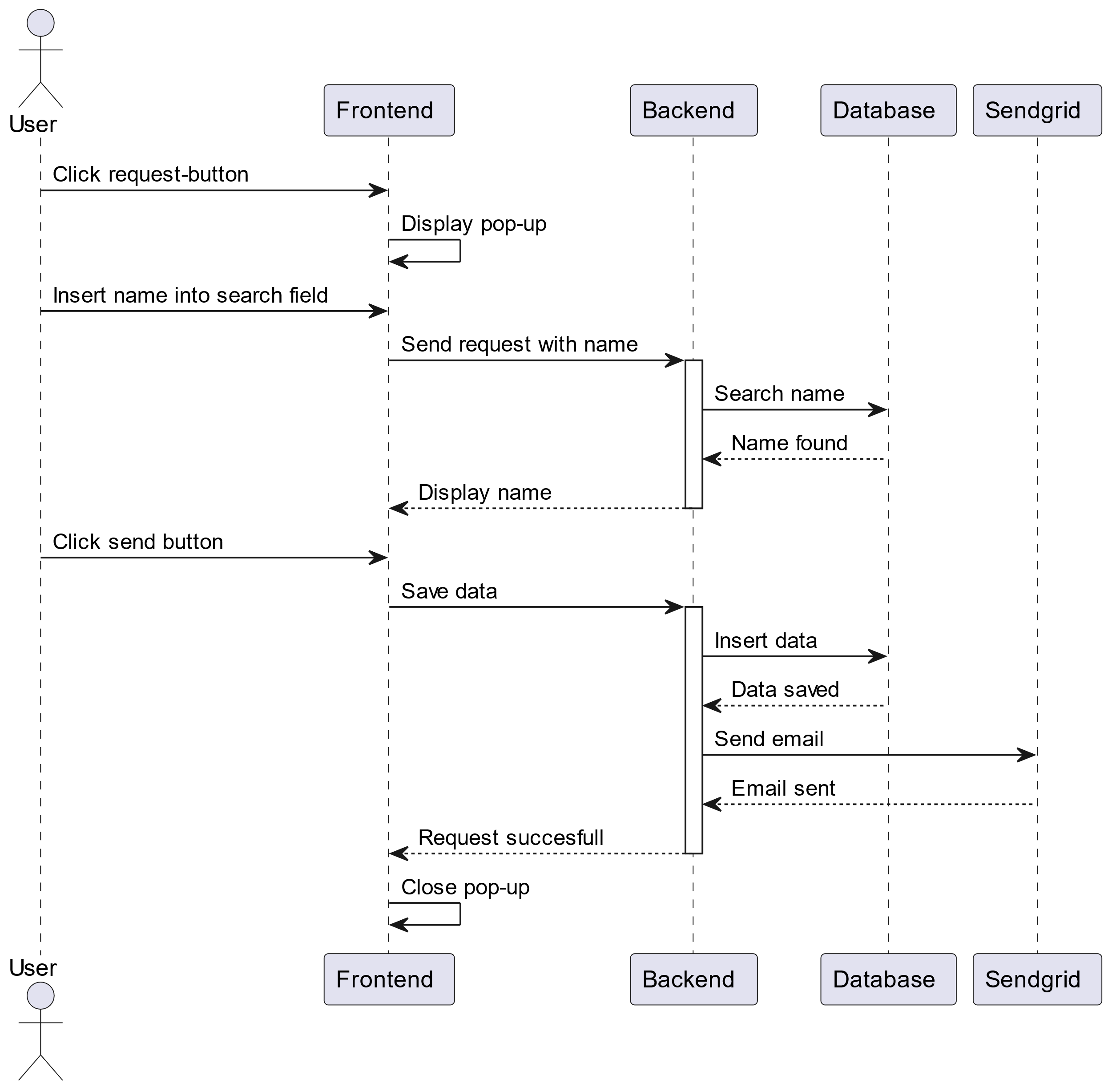
|  |  |  |
| --- | --- | --- |
| **Nummer** | **Story** | **Akzeptanzkriterien** |
| User Story – 01 | **Als** Versicherungsberater **möchte ich** die Möglichkeit haben, einen Stellvertreter anzufragen, **damit** mich jemand vertreten kann. | AC 1.1: Eingeloggt hat der Berater die Möglichkeit einen Knopf zu drücken und es öffnet sich ein Pop-up-Fenster. AC 1.2: Wenn er auf senden klickt, wird die Anfrage an das Backend weitergeleitet und gespeichert. |
| User Story – 02 | **Als** Versicherungsberater **möchte ich** sehen welchen Status meine Anfrage an einen Stellvertreter hat, **damit** ich weiss, ob ich jemand anderen suchen muss. | AC 2.1: In der Pendenzenliste wird der Status der Anfrage angezeigt AC 2.2: Es gibt die 3 Status «Pending», «Accepted» und «Declined». |
| User Story – 03 | **Als** Versicherungsberater **möchte ich** eine E-Mail mit der Antwort des möglichen Stellvertreters erhalten, **damit** ich weiss, ob er sie angenommen oder abgelehnt hat. | AC 3.1: Wenn der Stellvertreter annimmt oder ablehnt, wird dem Versicherungsberater eine Email mit der korrekten Antwort zugestellt. |
| User Story – 04 | **Als** Versicherungsberater **möchte ich** ein Pop-up erhalten, **damit** ich in diesem den richtigen Berater suchen und das richtige Datum für die Anfrage eingeben kann. | AC 4.1: Wenn der Berater auf den «Stv. Request»-Knopf drückt soll sich das Pop-up öffnen. AC 4.2: Das Pop-up besteht aus mind. drei Feldern und zwei Knöpfen (Name des Beraters, Startdatum, Enddatum, Abbrechen und Senden) |
| User Story – 05 | **Als** Stellvertreter **möchte ich** meine Anfragen in der Pendenzenliste sehen, **damit** ich diese dort einfach annehmen oder ablehnen kann. | AC 5.1: Die Anfragen müssen in der Pendenzenliste angezeigt werden.  AC 5.2: Wenn der Stellvertreter ablehnt oder annimmt, verschwinden die Knöpfe und die Pendenz wird als «declined» angezeigt. AC 5.3: Wenn der Stellvertreter annimmt, wir die Pendenz als «accepted» angezeigt. |
| User Story – 06 | **Als** Stellvertreter **möchte ich** zwischen meinem und den Personen, die ich Stellvertrete einfach wechseln können, **damit** die Kundendaten nicht vermischt werden. | AC 6.1: Wenn der Stellvertreter auf den Knopf drückt, wechselt sich das Profil AC 6.2: Der Knopf ist sichtbar ab dem Startdatum der Stellvertretung. AC 6.2: Der Knopf verschwindet, wenn das Enddatum erreicht ist. |
| User Story – 07 | **Als** Stellvertreter **möchte ich** eine E-Mail erhalten, wenn ich eine Anfrage als Stellvertreter für jemanden bekommen habe, **damit** ich die Anfrage sehe, obwohl ich die Applikation nicht offen habe. | AC 7.1: Wenn eine Anfrage an einen Stellvertreter geschickt wird, soll die E-Mail abgeschickt werden, mit den Informationen der Person, welche die Anfrage geschickt hat. |
| User Story – 08 | **Als** Stellvertreter **möchte ich** die Möglichkeit haben die Stellvertretung jederzeit beenden zu können. | AC8.1: Der Befugnis Geber muss einen Knopf haben, mit dem er die Stellvertretung jederzeit beenden kann. |

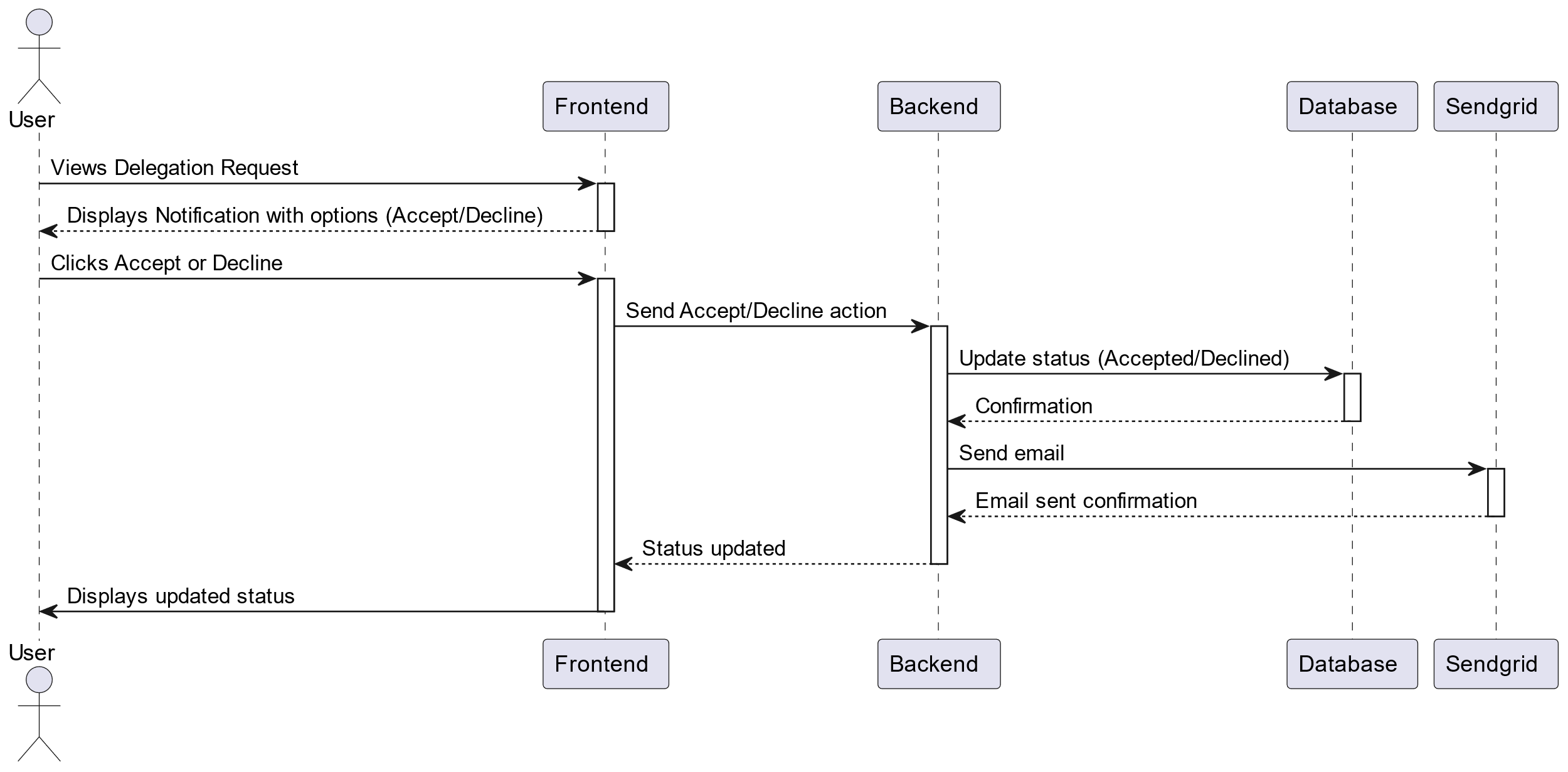
## Planen

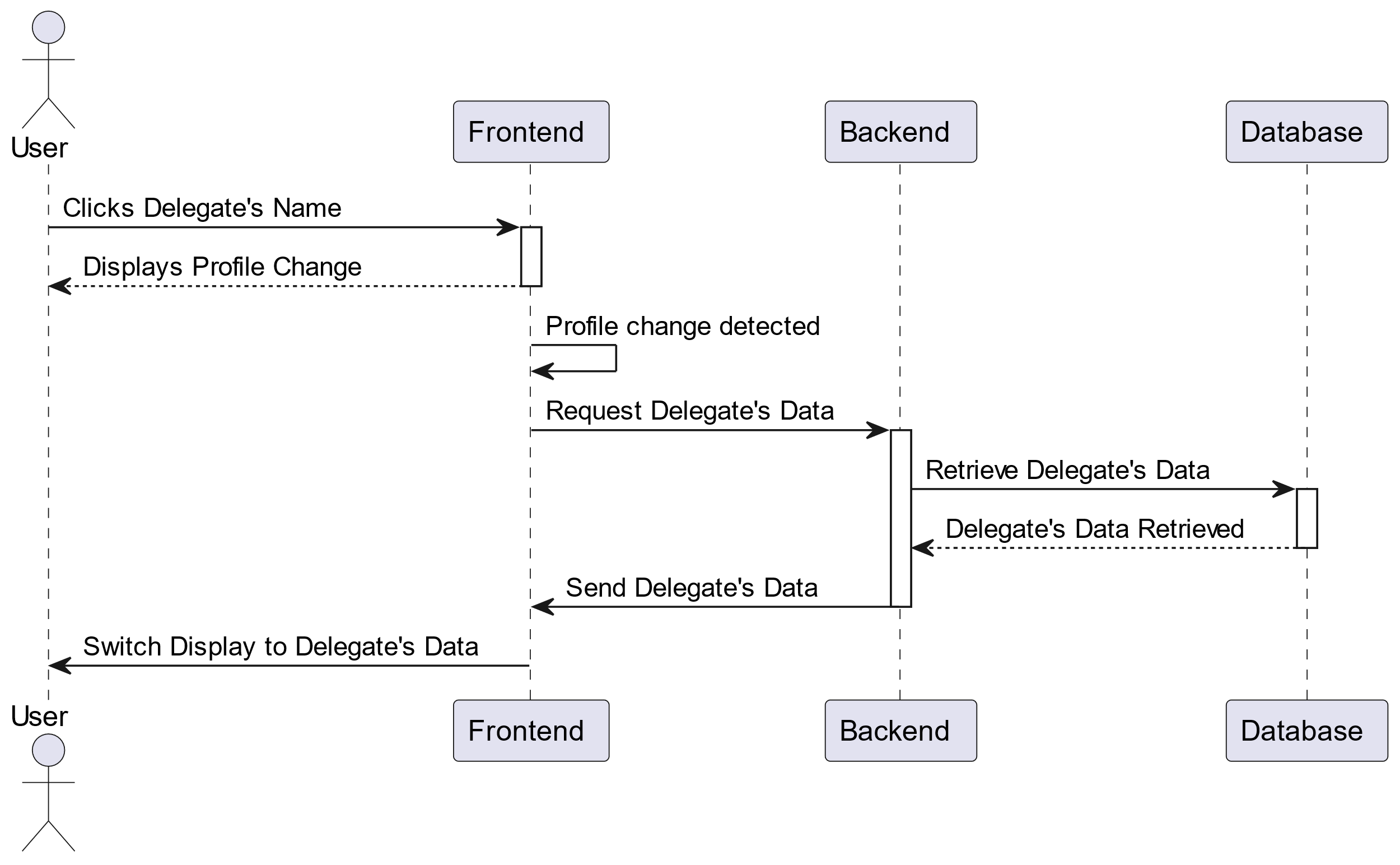
### 10.1 Realisierungskonzept

#### 10.1.1 Soll-Zustand

Der Soll-Zustand soll Versicherungsberatern ermöglichen einen Stellvertreter zu bestimmen, falls sie einmal abwesend sein sollten.







#### 10.1.2 Frontendänderungen

Im Frontend wird eine Profilübersicht auf der rechten Seite hinzugefügt. Der Berater soll einfach und schnell zwischen seinem Profil und den Profilen der Berater, welche er stellvertritt, wechseln können.   
Neue Komponente müssen im Storybook festgehalten werden.   
Für jede Request wird eine Tracing-ID generiert.

#### 10.1.3 Backendänderungen

Im Backend wird ein Sub-Task hinzugefügt um die Stellvertretungsanfragen zu managen und den Zugriff als Stellvertreter auf die Daten des Zustellvertretenden zu ermöglichen.

### 10.2 GUI-Design

Ein Bild, das Text, Screenshot, Software, Diagramm enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Zahl, Schrift enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Zahl, Schrift enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Zahl, Diagramm enthält.

Automatisch generierte Beschreibung

#### 10.2.1 Variante 1

Die Variante 1 zeigt das ausgewählte Profil in roter Farbe an, es entsteht ein deutlich visueller Kontrast und man sieht auf den ersten Blick, welches Profil hervorgehoben wird.

#### 10.2.2 Variante 2

Die Variante 2 ist deutlich schlichter und simpler, ein einfacher Radiobutton symbolisiert welches Profil das aktuell ausgewählte Profil ist.

### 10.3 Klassendiagramm

### 10.4 Datenmodell

Für die Datenbank brauche ich einen weiteren Subtyp für die Tabelle «Task». Der Tabellenname lautet «task$delegation\_request\_task». Diese Tabelle braucht natürlich die Id der Task Tabelle, sowie ein Start- und Enddatum, ebenfalls muss der Name des Auftraggebers mitgegeben werden sowie derjenige der die Stellvertreter Position einnehmen soll.

### 10.5 Testkonzept

Das Testkonzept besteht aus automatisierten und manuellen Tests, damit wird sichergestellt, dass das Projekt umfassend abgedeckt ist. Die Implementierung und Durchführung der unten aufgeführten Tests findet in der Phase «Kontrollieren» statt. Dadurch wird nach dem Abschluss der «Realisierungsphase» eine einheitliche Kontrolle und Evaluation ermöglicht und überprüft ob die Anforderungen gemässe der Aufgabenstellung umgesetzt wurden.

#### 10.5.1 Testziele

Das Ziel, welches mit dem Testprozess erreicht werden soll, ist die Überprüfung und Sicherstellung der Funktionsfähigkeit, Sicherheit, Benutzerfreundlichkeit und Zuverlässigkeit der neuen Funktionen, Features und Implementierungen. Dabei wird überprüft, ob die Software, die zuvor festgelegten User Stories und Akzeptanzkriterien zu allen Anforderungen erfüllt. Zusätzlich soll sichergestellt werden, dass die Inbetriebnahme ohne Fehler und Sicherheitslücken durchgeführt werden kann.

Was bewusst nicht getestet wurde:

#### 10.5.2 Testumgebung

|  |  |
| --- | --- |
| **Gerät:** | Lenovo ThinkBook 15 G4 ABA |
| **Betriebssystem:** | Windows 11 Pro |
| **Systemtyp:** | 64-Bit |
| **Prozessor:** | AMD Ryzen 7 5825U with Radeon Graphics (2.00 GHz) |
| **RAM:** | 16 GB 3200 MHz |
| **Browser** | Chrome Version 124.0.6367.79 (Offizieller Build) (64-Bit) |
| **URL:** | Frontend: <http://localhost:3000>  Backend: <http://localhost:8080> |

#### 10.5.3 Testfälle

|  |  |
| --- | --- |
|  | Test-Case 1 |
| Anforderung: | User Story – 01, AC 1.1 |
| Beschreibung: | Überprüfung, ob das Pop-up-Fenster ordnungsgemäß erscheint, wenn der Benutzer den dafür vorgesehenen Knopf drückt. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsvertreter ohne aktiven Stellvertreter. |
| Schritte: | 1. Navigiere zu der Seite oder dem Bereich, wo der Knopf für die Stellvertreter Anfrage verfügbar ist. 2. Klicke auf den «Stellvertreter anfragen» Knopf. 3. Überprüfe, ob sich ein Pop-up Fenster öffnet. |
| Erwartetes Resultat: | Das Pop-up Fenster öffnet sich ohne Probleme in der Mitte der Anwendung. |

|  |  |
| --- | --- |
|  | Test-Case 2 |
| Anforderung: | User Story – 01, AC 1.2 |
| Beschreibung: | Überprüfung, ob die Anfrage erfolgreich an das Backend gesendet und gespeichert wird, nachdem der Benutzer das Pop-up-Fenster ausgefüllt und auf "senden" geklickt hat. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsberater ohne aktiven Stellvertreter |
| Schritte: | 1. Navigiere zu der Seite oder dem Bereich, wo der Knopf für die Stellvertreter-Anfrage verfügbar ist. 2. Klicke auf den "Stellvertreter anfragen" Knopf. 3. Fülle alle erforderlichen Felder im Pop-up-Fenster aus. 4. Klicke auf "senden". 5. Überprüfe, ob die Anfrage erfolgreich an das Backend weitergeleitet wird und im System gespeichert wird. |
| Erwartetes Resultat: | In der Datenbank sollte ein Eintrag zu finden sein. |

|  |  |
| --- | --- |
|  | Test-Case 2 |
| Anforderung: | User Story – 02, AC 1.1 |
| Beschreibung: | Überprüfen, ob der Status der Anfrage an einen Stellvertreter in der Pendenzenliste für den Versicherungsberater korrekt angezeigt wird. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsberater, welcher eine Anfrage versendet hat. |
| Schritte: | 1. Navigiere zur Pendenzenliste. 2. Suche nach der Anfrage, für die du den Status überprüfen möchtest. 3. Überprüfe, ob der Status der Anfrage (Pending, Accepted oder Declined) neben der entsprechenden Anfrage korrekt angezeigt wird. 4. Wiederhole den Vorgang für verschiedene Anfragen, um sicherzustellen, dass alle drei Statusoptionen korrekt angezeigt werden. |
| Erwartetes Resultat: | In der Pendenzenliste wird der richtige Status angezeigt. |

|  |  |
| --- | --- |
|  | Test-Case 4 |
| Anforderung: | User Story – 05, AC |
| Beschreibung: | Überprüfen, ob die Anfragen des Stellvertreters korrekt in der Pendenzenliste angezeigt werden. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsberater. |
| Schritte: | 1. Melde dich als Stellvertreter an. 2. Navigiere zur Pendenzenliste oder einem Bereich, wo die Anfragen angezeigt werden. 3. Überprüfe, ob alle Anfragen des Stellvertreters in der Liste angezeigt werden. |
| Erwartetes Resultat: | In der Pendenzenliste werden alle Anfragen angezeigt. |

|  |  |
| --- | --- |
|  | Test-Case 5 |
| Anforderung: | User Story – 05, AC 5.1 – 5.2 |
| Beschreibung: | Überprüfen, ob die Knöpfe zum Annehmen oder Ablehnen einer Anfrage verschwinden und die Anfrage als "declined" in der Pendenzenliste angezeigt wird, nachdem der Stellvertreter die Anfrage abgelehnt hat. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsberater mit einer Anfrage als Stellvertreter. |
| Schritte: | 1. Suche in der Pendenzenliste nach einer Anfrage, die du ablehnen möchtest. 2. Klicke auf den Knopf zum Ablehnen der Anfrage. 3. Überprüfe, ob die Knöpfe zum Annehmen oder Ablehnen verschwinden. 4. Überprüfe, ob die abgelehnte Anfrage in der Pendenzenliste als "declined" angezeigt wird. |
| Erwartetes Resultat: | Die Anfrage wird als «Declined» angezeigt und die Knöpfe sind verschwunden. |

|  |  |
| --- | --- |
|  | Test-Case 6 |
| Anforderung: | User Story – 05, AC 5.1 & 5.3 |
| Beschreibung: | Überprüfen, ob die Knöpfe zum Annehmen oder Ablehnen einer Anfrage verschwinden und die Anfrage als "declined" in der Pendenzenliste angezeigt wird, nachdem der Stellvertreter die Anfrage abgelehnt hat. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsberater mit einer Anfrage als Stellvertreter. |
| Schritte: | 1. Suche in der Pendenzenliste nach einer Anfrage, die du annehmen möchtest. 2. Klicke auf den Knopf zum Annehmen der Anfrage. 3. Überprüfe, ob die Knöpfe zum Annehmen oder Ablehnen verschwinden. 4. Überprüfe, ob die abgelehnte Anfrage in der Pendenzenliste als "accepted" angezeigt wird. |
| Erwartetes Resultat: | Die Anfrage wird als «Accepted» angezeigt und die Knöpfe sind verschwunden. |

|  |  |
| --- | --- |
|  | Test-Case 7 |
| Anforderung: | User Story – 06 |
| Beschreibung: | Überprüfen, ob das Profil des Stellvertreters erfolgreich gewechselt wird, wenn der Stellvertreter auf den dafür vorgesehenen Knopf drückt. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsberater mit einer Rolle als Stellvertreter. |
| Schritte: | 1. Melde dich als Stellvertreter an. 2. Suche nach dem Knopf, der den Profilwechsel ermöglicht. 3. Klicke auf den Knopf, um das Profil zu wechseln. 4. Überprüfe, ob das Profil erfolgreich gewechselt wurde. |
| Erwartetes Resultat: |  |

|  |  |
| --- | --- |
|  | Test-Case 8 |
| Anforderung: | User Story – 06 |
| Beschreibung: | Überprüfen, ob der Knopf zum Wechseln des Profils ab dem Startdatum der Stellvertretung sichtbar ist. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsberater mit einer Rolle als Stellvertreter. |
| Schritte: | 1. Stelle sicher, dass das heutige Datum vor dem Startdatum der Stellvertretung liegt. 2. Überprüfe, ob der Knopf sichtbar ist. |
| Erwartetes Resultat: | Der Knopf ist sichtbar |

|  |  |
| --- | --- |
|  | Test-Case 9 |
| Anforderung: | User Story – 06 |
| Beschreibung: | Überprüfen, ob der Knopf zum Wechseln des Profils verschwindet, wenn das Enddatum der Stellvertretung erreicht ist. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsberater mit einer Rolle als Stellvertreter. |
| Schritte: | 1. Stelle sicher, dass das heutige Datum nach dem Enddatum der Stellvertretung liegt. 2. Überprüfe, ob der Knopf sichtbar ist. |
| Erwartetes Resultat: | Der Knopf ist nicht mehr sichtbar. |

|  |  |
| --- | --- |
|  | Test-Case 10 |
| Anforderung: | User Story – 07 |
| Beschreibung: | Überprüfen, ob der Stellvertreter eine E-Mail erhält, wenn eine Anfrage an ihn als Stellvertreter gesendet wird, mit den erforderlichen Informationen über die Person, die die Anfrage gesendet hat. |
| Voraussetzung: |  |
| Schritte: | 1. Überprüfe, ob der Stellvertreter eine E-Mail erhält. 2. Überprüfe den Inhalt der E-Mail, um sicherzustellen, dass sie die erforderlichen Informationen über die Person enthält, die die Anfrage gesendet hat. |
| Erwartetes Resultat: | Die Email wurde erhalten und beinhaltet Start-, Enddatum sowie den Namen des Absenders. |

|  |  |
| --- | --- |
|  | Test-Case 11 |
| Anforderung: | User Story – 08 |
| Beschreibung: | Überprüfen, ob der Befugnis Geber die Möglichkeit hat, die Stellvertretung jederzeit zu beenden, indem er auf den dafür vorgesehenen Knopf klickt. |
| Voraussetzung: | Bereits eingeloggt als Versicherungsberater mit einer Rolle als Stellvertreter. |
| Schritte: | 1. Melde dich als Befugnis Geber an. 2. Suche nach dem Knopf, der es ermöglicht, die Stellvertretung zu beenden. 3. Klicke auf den Knopf, um die Stellvertretung zu beenden. 4. Überprüfe, ob die Stellvertretung erfolgreich beendet wurde. |
| Erwartetes Resultat: | Die Stellvertretung wurde erfolgreich beendet und der Stellvertreter hat keinen Zugriff mehr auf das Profil des Befugnis Gebers. |

## Entscheiden

Die Entscheidung fällt zwischen zwei Varianten im Design bezüglich der Anzeige des aktiven Profils. Der Unterschied wird in den Punkten 10.2.1 & 10.2.2 beschrieben.

### 11.1 Entscheidungsmatrix

|  |  |  |
| --- | --- | --- |
| **Kriterium** | **Variante 1** | **Variante 2** |
| Benutzererfahrung | + | - |
| Komplexität | = | = |
| Ästhetik | + | + |
| **Bewertung:** | **++** | **--** |

### 11.2 Begründung

#### 11.2.1 Benutzererfahrung

Die Variante 1 ist auf den ersten Blick deutlich einfacher und schneller zu erkennen, als Variante 2. Da es in Variante 2 keinen aktiven Farbunterschied gibt.

#### 11.2.2 Komplexität

Beide Varianten sind relativ einfach umzusetzen, da es sich nur um eine Anpassung im Frontend handelt. Bei Variante 1 muss man die Schrift- und Hintergrundfarbe ändern und in Variante 2 ein weiteres Objekt im Frontend hinzufügen. Die Funktion hinter beiden ist dennoch gleich.

#### 11.2.3 Ästhetik

Variante 2 sieht zwar schlichter und neutraler aus, doch Variante 2 bringt die Gewisse Menge an Farbe hinein, welche es ins Auge stechen lässt.

### 11.3 Entscheidung

Durch das Vergleichen von beiden Varianten in den drei Kriterien bin ich zum Schluss gekommen, dass Variante 1 die Variante sein wird, welche ich umsetzen werde.

## Realisieren

### 12.1 Datenbank

Es wurden einige Änderungen an der Datenbank vorgenommen. Wie im vorherigen Kapitel geplant musste eine Sub-Tabelle für die «Task»-Tabelle erstellt werden. Diese habe ich «task$delegation\_request\_task» genannt. Diese brauchte vier Felder, «id», «start\_date», «end\_date», «delegate\_id» & «delegator\_id». Wobei «id» auf die «Task»-Tabelle referenziert. «delegate\_id» und «delegator\_id» fungieren ebenfalls als Fremdschlüssel und referenzieren beide auf die Tabelle «user». Zusätzlich musste ich in der «Task»-Tabelle «state» anpassen. Es gab eine Erweiterung der verfügbaren Status. Die Status «ACCEPTED» und «DECLINED» wurden hinzugefügt.

Zusätzlich ist mir im Verlaufe der Arbeit aufgefallen, dass ich in der «user»-Tabelle das Feld «email» ebenfalls brauche. Deshalb habe ich diese Tabelle um das Feld erweitert.

Für das Update der Datenbank wurde eine Flyway Migration benutzt. Das File musste wie folgend bennant werden:

<Prefix><Version>\_\_<Description>.sql

In diesen Fällen wurden die beiden Files «V1\_0\_17\_\_add\_subtask\_delegation.sql» und «V1\_0\_18\_\_extend\_user\_entity.sql» genannt.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Schrift, Screenshot, weiß enthält.

Automatisch generierte Beschreibung

### 12.2 Backend

#### 12.2.1 Änderungen an Models, DTOs und Mappers

Task Model

Dieser Dto entspricht der Tabelle in der Datenbank, wo alle Daten, die für die Anfrage einer Stellvertretung nötig sind, gespeichert sind.

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte Beschreibung

TaskGetDto und TaskPostDto

Werden ebenfalls, wie das obige Model erweitert. Die GetDto-Klasse wird für GET-Request gebraucht, die an Task requestet werden. Genauso wird die PostDto-Klasse für die POST-Requests benötigt. Der Hauptunterschied der beiden Klassen liegt darin, dass «TaskGetDto» noch eine Id braucht. In beiden Klassen musste «DelegationRequestTaskGetDto» bzw. «DelegationRequestTaskPostdto» hinzugefügt werden. Beide beinhalten die gleichen Daten wie das Model.

Da die Antwort als JSON zurückkommt und es schon zwei andere Task-Typen gibt, muss man zwischen ihnen unterscheiden können. Dafür wurde mit Jackson gearbeitet, einer Library für Java, welche für JSON-Handling zuständig ist. Dies ist sichtbar bei der Annotation «@DiscriminatorMapping(…)». Damit kann man die Typen Eigenschaft als JSON-Response korrekt mappen und somit sicherstellen, dass die Datensätze in der richtigen Dto serialisiert werden.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungEin Bild, das Text, Screenshot, Schrift, Dokument enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungEin Bild, das Text, Screenshot, Dokument, Schrift enthält.

Automatisch generierte Beschreibung

DtoMapper

Ein Bild, das Text, Screenshot, Schrift, Dokument enthält.

Automatisch generierte BeschreibungEin Bild, das Text, Screenshot, Dokument, Schrift enthält.

Automatisch generierte BeschreibungDa mit der Einführung eines neuen Task natürlich ebenfalls der Mapper erweitert werden muss, wurde in dieser Klasse ein weiteres «Case» hinzugefügt.

#### 12.2.2 Controller und Services

Damit das Backend mit dem Frontend reibungslos kommunizieren kann, müssen die Controller und Services angepasst werden. Ein Controller ist dafür zuständig, die einkommenden Requests zu verarbeiten, er liest die Daten im Request, kontrolliert den Ablauf der Applikation und schickt eine Antwort zurück. Dafür konsumiert er Methoden, welche vom Service bereitgestellt werden. Diese beihnalten die Logik des P0rogramms. Für einen Teil der Realisierung dieser Aufgabe musste der «TaskService» angepasst werden, sowie der «TaskController».

Um die Rollenerweiterung umzusetzen, damit die Berechtigung des Stellvertreters ebenfalls ohne Probleme funktioniert, mussten ebenfalls der «CustomerController», der «LifeInsuranceRequestController» und der «UserController» verändert werden. Diese Anpassungen waren an diesen Stellen notwendig, wenn der Controller den eingeloggten User aufrufen wollte. Wenn ein Stellvertreter nun also seine Aufgabe als Stellvertretung ausführt und diese Abfrage durchgeführt worden wäre, wäre ein Fehler aufgetreten. Daher musste an diesen Stellen ein Check eingebaut werden.

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung Um den Rahmen der Dokumentation nicht zu sprengen, inkludiere ich hier nur ein Repräsentatives Beispiel, alle anderen dieser Änderungen sind natürlich ebenfalls im Anhang verfügbar.

TaskService

Im «TaskService» wurde die E-Mail-Benachrichtigungsfunktion aufgebaut und implementiert, ebenfalls wurden die updateTask und updateTaskDetails Methoden eingeführt.   
Je nach dem welche Antwort der angefragte Stellvertreter gibt, muss eine andere E-Mail zurückgesendet werden und der Status des Tasks beziehungsweise der Anfrage muss überarbeitet werden.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungEs gibt drei unterschiedliche Arten von E-Mails, die versendet werden können. Eine E-Mail wird versendet an den angefragten Stellvertreter, um ihm mitzuteilen, dass er eine neue Anfrage bekommen hat. Die anderen beiden E-Mails werden versendet, nachdem der angefragte Stellvertreter entweder angenommen oder abgelehnt hat.

Ein Bild, das Text, Screenshot, Dokument, Schrift enthält.

Automatisch generierte BeschreibungEin Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

TaskController

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungEin Bild, das Text, Screenshot, Schrift, Dokument enthält.

Automatisch generierte BeschreibungIm Hinblick auf die «update Task»-Funktionalität wurde einiges von Samuel Hajnik übernommen und angepasst, da die update Task Methode keine funktionale oder nicht funktionale Anforderung dieser Aufgabe ist. Daher sind die zwei «@ExampleObject» von Samuel übernommen.

### 11.3 Frontend

Im Frontend gab es drei Teile der Applikation, die bearbeitet oder erstellt werden mussten. Die Pendenzenliste musste um einen «Accept»- und einen «Decline»-Button erweitert werden. Zusätzlich musste der Manager für die Stellvertretung und das Pop-up für die Anfrage eines Stellvertreters, im Frontend aufgebaut werden.

#### 11.3.1 Components

dask-table.tsx

Ein Bild, das Text, Screenshot, Schrift, Dokument enthält.

Automatisch generierte BeschreibungIn diesem File wurde die Pendenzenliste angepasst. Dafür wird der Typ und der Status des Task überprüft. Falls es ein «DelegationRequestTask» ist und der Status in der Datenbank nicht bereits «ACCEPTED» oder «DECLINED» ist, werden dem Stellvertreter zwei Buttons in der Pendenzenliste angezeigt. Ebenfalls wurde eingebaut, dass nur der Stellvertreter und nicht derjenige, welcher die Anfrage gestellt hat, diese Buttons sehen kann. Die Anfrage ist dennoch inklusive Status auch bei dem User, welcher die Anfrage gestellt hat, sichtbar.

delegation-manager.tsx, delegation-manager.spec.tsx & delegation-manager.stories.tsx

In diesem Component wurde der Manger für die Stellvertretung aufgebaut. Es gibt einige Abhängigkeiten, welche beim Generieren in der Datenbank überprüft werden müssen. Zu jedem Zeitpunkt ist in diesem Component sichtbar: Der Name des Users und der Button, um einen Stellvertreter anzufragen.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungEin Bild, das Text, Schrift, Screenshot, Rechteck enthält.

Automatisch generierte BeschreibungRot abgebildet wird das aktive Profil, dadurch wird visuell symbolisiert, als welche Person man aktuell im Programm arbeiten kann. Hier ein Beispiel wie dies aussehen kann:

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte BeschreibungSollte man zusätzlich selbst einen Stellvertreter haben, gibt es unterhalb der ersten eine zusätzliche Tabelle. Dann sieht die Oberfläche wie folgt aus:

drawer.tsx, dialog.tsx, command.tsx & create-task.ts

Diese vier Klassen wurden von anderen Lernenden übernommen, da die Funktion der Klassen generell verwendbar ist und wichtig für den Aufbau des UI beziehungsweise des Managements der unterschiedlichen Tasks,

request-delegation-form.tsx, request-delegation-form.spec.tsx & request-delegation-form.stories.tsx

In diesem «Form» wird das Pop-up für die Stellvertretungs-Anfrage definiert. Wird auf den Button mit dem Schriftzug «Request Delegate» geklickt, öffnet sich dieses Pop-up und der Benutzer kann den Stellvertreter wählen, sowie das Start- und Enddatum bestimmen.

Ein Bild, das Text, Screenshot, Software, Display enthält.

Automatisch generierte BeschreibungBeim Aufrufen des Feldes «+ Stellvertreter wählen» wird eine Liste aus den Usern, welche in der Datenbank vorhanden sind, aufgerufen und aufgelistet, es gibt ebenfalls eine Suchoption. Falls es keine User in der Datenbank gibt, wird angezeigt: «Keinen Stellvertreter gefunden.»

Ein Bild, das Screenshot, Text, Software, Multimedia-Software enthält.

Automatisch generierte Beschreibung

Die Anfrage kann erst versendet werden, wenn ein Start- und ein Enddatum festgelegt wurden, ansonsten ist der «Anfrage senden»-Button deaktiviert.

Im Falle eines Erfolgs oder eines Fehlschlages beim Versenden der Anfrage erhält der User nach dem Betätigen des Knopfes eine Mitteilung.

#### 11.3.2 Seite in der Applikation

Im Umfang der laufenden Applikation sieht die Seite der Tasks wie folgt aus:

[SCREENSHOT]

### 11.4 Automatisiertes Testing

## Kontrollieren

### 13.1 Manuelles Testing

Mittels den zuvor erstellten Testkonzept aus dem Kapitel 10.5 werden die Tests durchgeführt.

#### 13.1.1 Testprotokoll

Die Testumgebung für das folgende Testprotokoll kann im Kapitel 10.5.2 nachgeschlagen werden.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testfall | Resultat | Datum | Tester | Bemerkung |
| 1 |  |  | Mia Rascher |  |
| 2 |  |  | Mia Rascher |  |
| 3 |  |  | Mia Rascher |  |
| 4 |  |  | Mia Rascher |  |
| 5 |  |  | Mia Rascher |  |
| 6 |  |  | Mia Rascher |  |
| 7 |  |  | Mia Rascher |  |
| 8 |  |  | Mia Rascher |  |
| 9 |  |  | Mia Rascher |  |
| 10 |  |  | Mia Rascher |  |
| 11 |  |  | Mia Rascher |  |

#### 13.1.2 Testbericht Massnahmen schreiben

Testfall 1

Testfall 2

Testfall 3

Testfall 4

Testfall 5

Testfall 6

Testfall 7

Testfall 8

Testfall 9

Testfall 10

Testfall 11

## Auswerten

### 14.1 Vergleich Ist / Soll

### 14.2 Fazit

### 14.3 Schlussreflexion

## Danksagung

## Glossar

|  |  |
| --- | --- |
| Begriff | Definition |
| [Begriff 01] | [Definition] |
| [Begriff 02] | [Definition] |
| [Begriff 03] | [Definition] |
| [Begriff 04] | [Definition] |

## Literaturverzeichnis

### 17.1 Abbildungsverzeichnis

## Anhänge