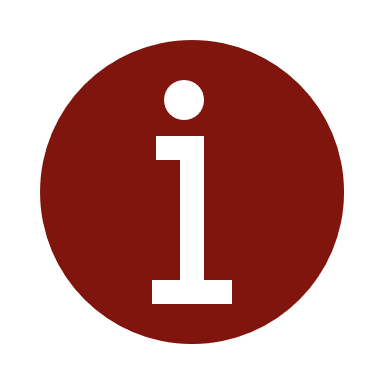
IPA 2024  
 Abschlussarbeit

OVWEB – Workflow Unterbruch

20. März 2024



|  |  |
| --- | --- |
| Kanditat / Autor:  Firma:  Berufsfachschule:  Abteilung:  Hauptexperte:  Nebenexperte:  Berufsbildner:  Verantwortliche Fachkraft:  Fachrichtung:  Projektvorgehensmodell: | Samuel Hajnik  Generali AG  TBZ  IT Apprentices  Marc Siegmund  Janis Marty  Roger Sorg  Roman Babenko  Applikationsentwicklung  IPERKA |

Inhalt

[Vorwort 4](#_Toc160531718)

[Teil 1 - Administration 5](#_Toc160531719)

[1. Projektorganisation 5](#_Toc160531720)

[2. Organisation der Arbeitsergebnisse 6](#_Toc160531721)

[3. Datensicherung 6](#_Toc160531722)

[3.1 Code 6](#_Toc160531723)

[3.2 Dokumentation 6](#_Toc160531724)

[4. Deklaration der benützten Firmenstandards 7](#_Toc160531725)

[5. Zeitplan 10](#_Toc160531726)

[6. Arbeitsjournal 11](#_Toc160531727)

[6.1 Dienstag, 05.03.2024 11](#_Toc160531728)

[6.2 Mittwoch, 06.03.2024 12](#_Toc160531729)

[6.3 Donnerstag, 07.03.2024 13](#_Toc160531730)

[6.4 Freitag, 08.03.2024 14](#_Toc160531731)

[6.5 Dienstag, 12.03.2024 15](#_Toc160531732)

[6.6 Mittwoch, 13.03.2024 16](#_Toc160531733)

[6.7 Donnerstag, 14.03.2024 17](#_Toc160531734)

[6.8 Freitag, 15.03.2024 18](#_Toc160531735)

[6.9 Dienstag, 19.03.2024 19](#_Toc160531736)

[6.10 Mittwoch, 20.03.2024 20](#_Toc160531737)

[Teil 2 – Projekt 21](#_Toc160531738)

[7. Kurzfassung 21](#_Toc160531739)

[7.1 Ausgangslage 21](#_Toc160531740)

[7.2 Umsetzung 21](#_Toc160531741)

[7.3 Ergebnis 21](#_Toc160531742)

[8. Projektbeschreibung 22](#_Toc160531743)

[8.1 IPERKA 22](#_Toc160531744)

[8.1.1 Informieren 22](#_Toc160531745)

[8.1.2 Planen 22](#_Toc160531746)

[8.1.3 Entscheiden 22](#_Toc160531747)

[8.1.4 Realisieren 23](#_Toc160531748)

[8.1.5 Kontrollieren 23](#_Toc160531749)

[8.1.6 Auswerten 23](#_Toc160531750)

[9. Einleitung 24](#_Toc160531751)

[10. Informieren 24](#_Toc160531752)

[10.1 Ziele der Aufgabenstellung 24](#_Toc160531753)

[10.2 Vorgaben 24](#_Toc160531754)

[10.3 Ist-Analyse 24](#_Toc160531755)

[10.3.1 Frontend 25](#_Toc160531756)

[10.3.2 Backend 26](#_Toc160531757)

[10.3.3 Überblick 27](#_Toc160531758)

[10.3.4 Klassendiagramme 28](#_Toc160531759)

[10.3.5 Aktivitätsdiagramme 28](#_Toc160531760)

[10.4 User Stories 29](#_Toc160531761)

[10.4.1 Rollen 29](#_Toc160531762)

[10.4.2 Funktionale Anforderungen 29](#_Toc160531763)

[10.4.3 Nicht funktionale Anforderungen: 32](#_Toc160531764)

[11. Planen 35](#_Toc160531765)

[11.1 Realisierungskonzept 35](#_Toc160531766)

[11.2 Testkonzept 35](#_Toc160531767)

[11.3 Datenmodell 35](#_Toc160531768)

[11.4 Design 35](#_Toc160531769)

[12. Entscheiden 36](#_Toc160531770)

[12.1 Entscheid 36](#_Toc160531771)

[13. Realisieren 37](#_Toc160531772)

[13.1 Projektumgebung 37](#_Toc160531773)

[13.2 Datenbank 37](#_Toc160531774)

[13.3 Backend Unit Tests 37](#_Toc160531775)

[13.4 Frontend Unit Tests 37](#_Toc160531776)

[14. Kontrollieren 38](#_Toc160531777)

[14.1 Testkonzept 38](#_Toc160531778)

[14.2 Testprotokoll Samuel Hajnik 38](#_Toc160531779)

[15. Auswerten 39](#_Toc160531780)

[15.1 Vergleich Ist / Soll 39](#_Toc160531781)

[15.2 Fazit zur IPA 39](#_Toc160531782)

[15.3 Reflexion 39](#_Toc160531783)

[Danksagung 40](#_Toc160531784)

[Glossar 41](#_Toc160531785)

[Abbildungsverzeichnis 42](#_Toc160531786)

[Quellenverzeichnis 43](#_Toc160531787)

Vorwort

Teil 1 - Administration

1. Projektorganisation

**Lehrbetrieb und Durchführungsort: Nebenexperte:**Generali AG Marty Janis  
Soodmattenstrasse 4 janis.marty@ubs.com  
8134 Adliswil 079 227 53 58  
[058 472 75 00](https://www.google.com/search?client=firefox-b-d&q=generali+adliswil+&lqi=ChFnZW5lcmFsaSBhZGxpc3dpbCIDiAEBSKmIiN3lgICACFofEAAYABgBIhFnZW5lcmFsaSBhZGxpc3dpbCoECAIQAJIBEGluc3VyYW5jZV9hZ2VuY3mqAVcKDS9nLzExYjY2anJoc2QQASoMIghnZW5lcmFsaSgFMh8QASIb7rjLivujVg1VYAHWgXdZ_l1vC9h1_he9K8OnMhUQAiIRZ2VuZXJhbGkgYWRsaXN3aWw)  
life.ch@generali.com

**Kandidat:**Hajnik Samuel  
Im Heumoos 2  
8906 Bonstetten  
077 968 38 56  
samuel.hajnik@generali.com

**Berufsbildner / Lehrfirma:**Sorg Roger  
Generali AG  
Soodmattenstrasse 4  
8134 Adliswil  
058 472 40 42  
roger.sorg@generali.com

**Verantwortliche Fachkraft:**Babenko Roman  
Generali AG  
Soodmattenstrasse 4  
8134 Adliswil  
076 410 87 29  
roman.babenko@loewenzahn.dev

**Hauptexperte:**Siegmund Marc  
[marc.siegmund+pkorg@gmail.com](mailto:marc.siegmund+pkorg@gmail.com)  
078 656 35 16

1. Organisation der Arbeitsergebnisse

Dieses Kapitel beschreibt, wie während der IPA sicherstellt wird, dass alle relevanten Daten gesichert sind und nötigenfalls wiederhergestellt werden können.

1. Datensicherung

Die Datensicherung ist essenziell für den Fall, dass Probleme auftreten sollen, somit ist sie ein wichtiger Bestandteil der IPA. Für die tägliche Sicherung der Arbeit wurde auf verschiedene Speicherungssysteme gesetzt.

* 1. Code

Für den Code wurde ein neuer Github-Branch in der Team-Repository erstellt.

**Branch**: [Link]

Hier ist der Verlauf der Commits auf den Branch:

* 1. Dokumentation

Durch OneDrive wird der neuste Stand laufend gespeichert. Für zusätzliche Sicherheit wird eine separate Github Repository am Schluss von jedem Tag ein Backup erstellt.  
**Repository:** [Link]

Hier ist der Verlauf der Commits auf die Repository:

1. Deklaration der benützten Firmenstandards

Im Entwickler-Team werden folgende Technologien verwendet.

|  |  |
| --- | --- |
| Frontend | |
| **Name** | **Einsatz** |
| Next.js | Frontend-Technologie |
| shadcn | Komponenten Design Bibliothek |
| Typescript | Frontendsprache |
| Prettier | Code Formatierung |
| Jest | Tests |
| Storybook | Frontend Dokumentation |

|  |  |
| --- | --- |
| Backend | |
| **Name** | **Einsatz** |
| Springboot | Backend-Technologie |
| Java | Backendsprache |
| Google Java Styleguide | Code Formatierung |
| Lombok | Annotations für Automatisierung |
| Slf4J | Logging |
| JUnit | Tests |
| Mockito | Testdaten Mocking |

|  |  |
| --- | --- |
| Daten Layer | |
| **Name** | **Einsatz** |
| PostgreSQL | Datenbank |
| Flyway | Datenbankversionierungstool |

|  |  |
| --- | --- |
| Identity Provider | |
| **Name** | **Einsatz** |
| OpenID Connect | Identitätsprotokoll |
| Keycloak | Identitätsmanagement |

|  |  |
| --- | --- |
| Lokale Entwicklung | |
| **Name** | **Einsatz** |
| Jetbrains IntelliJ | Entwicklungsumgebung |
| Docker | Containerisierungstool |

|  |  |
| --- | --- |
| UI/UX Design | |
| **Name** | **Einsatz** |
| Figma | Diagramme/Skizzen |
| Generali Styleguide | Styleguide |

|  |  |
| --- | --- |
| Dokumentation und Admin | |
| **Name** | **Einsatz** |
| Microsoft Office | Dokumentation |
| Google Chat | Kommunikation |
| Github CoPilot | Textgenerierung, Informationen |
| Draw.io / Plantuml | UML-Diagramme |

1. Zeitplan
2. Arbeitsjournal
   1. ****Dienstag, 05.03.2024****

|  |  |
| --- | --- |
| Tätigkeiten | * Zeitplan erstellen * Dokumentation starten * User Stories * Expertengespräch |
| **Erreichte Ziele** | * Zeitplan * User Stories |

|  |
| --- |
| **Probleme** |
| - |
| Hilfestellung |
| Roman hat mir nach dem Expertengespräch noch Tipps gegeben zu meinem Zeitplan. |
| Ausserplanmässige Arbeiten |
| - |
| Reflexion |
| ***Was lief Heute positiv:***  **Ich konnte Heute positiv und motiviert in meine IPA starten. Ich habe mir vorgenommen vieles möglichst schnell zu erledigen, damit ich der begrenzten Zeit vorauskomme. Ich konnte den Zeitplan in einer relative schnellen Zeit erledigen, was ich mir nicht dachte, da es doch eine anstrengende Aufgabe ist. Zugleich hat mir das Dokumentieren auch einfach gefallen Heute, ich konnte schon vieles runterschreiben, vor allem in der Phase Informieren, was mich nach vorne drängen wird in den nächsten Tagen. Beim Expertengespräch konnte ich sehr gute Einblicke holen, wie es laufen wird, was wichtig ist und auf was ich achten muss.**  ***Was lief nicht so gut:***  **Vor dem Expertengespräch habe ich mich kurzfristig entschieden meinen Zeitplan noch zu umstrukturieren, da es für mich ungenau war. Nach dem Gespräch holte ich mir noch Hilfe bei Roman, damit ich eine Idee bekomme, wie ich es verbessern kann. Er hat mir vorgeschlagen Formeln zu benutzen. Somit konnte ich meinen Zeitplan verbessern.**  ***Fazit:*** Heute war ein erfolgreicher Tag. Ich konnte super starten und fortlaufend arbeiten. Das Expertengespräch hat mich nur sicherer gemacht und ich bin mir sicher, dass es in den nächsten Tagen gut fortlaufen wird. |

* 1. Mittwoch, 06.03.2024

|  |  |
| --- | --- |
| Tätigkeiten |  |
| **Erreichte Ziele** |  |

|  |
| --- |
| **Probleme** |
|  |
| Hilfestellung |
|  |
| Ausserplanmässige Arbeiten |
|  |
| Reflexion |
| ***Was lief Heute positiv:***  ***Was lief nicht so gut:***  ***Fazit:*** |

* 1. Donnerstag, 07.03.2024

|  |  |
| --- | --- |
| Tätigkeiten |  |
| **Erreichte Ziele** |  |

|  |
| --- |
| **Probleme** |
|  |
| Hilfestellung |
|  |
| Ausserplanmässige Arbeiten |
|  |
| Reflexion |
| ***Was lief Heute positiv:***  ***Was lief nicht so gut:***  ***Fazit:*** |

* 1. Freitag, 08.03.2024

|  |  |
| --- | --- |
| Tätigkeiten |  |
| **Erreichte Ziele** |  |

|  |
| --- |
| **Probleme** |
|  |
| Hilfestellung |
|  |
| Ausserplanmässige Arbeiten |
|  |
| Reflexion |
| ***Was lief Heute positiv:***  ***Was lief nicht so gut:***  ***Fazit:*** |

* 1. Dienstag, 12.03.2024

|  |  |
| --- | --- |
| Tätigkeiten |  |
| **Erreichte Ziele** |  |

|  |
| --- |
| **Probleme** |
|  |
| Hilfestellung |
|  |
| Ausserplanmässige Arbeiten |
|  |
| Reflexion |
| ***Was lief Heute positiv:***  ***Was lief nicht so gut:***  ***Fazit:*** |

* 1. Mittwoch, 13.03.2024

|  |  |
| --- | --- |
| Tätigkeiten |  |
| **Erreichte Ziele** |  |

|  |
| --- |
| **Probleme** |
|  |
| Hilfestellung |
|  |
| Ausserplanmässige Arbeiten |
|  |
| Reflexion |
| ***Was lief Heute positiv:***  ***Was lief nicht so gut:***  ***Fazit:*** |

* 1. Donnerstag, 14.03.2024

|  |  |
| --- | --- |
| Tätigkeiten |  |
| **Erreichte Ziele** |  |

|  |
| --- |
| **Probleme** |
|  |
| Hilfestellung |
|  |
| Ausserplanmässige Arbeiten |
|  |
| Reflexion |
| ***Was lief Heute positiv:***  ***Was lief nicht so gut:***  ***Fazit:*** |

* 1. Freitag, 15.03.2024

|  |  |
| --- | --- |
| Tätigkeiten |  |
| **Erreichte Ziele** |  |

|  |
| --- |
| **Probleme** |
|  |
| Hilfestellung |
|  |
| Ausserplanmässige Arbeiten |
|  |
| Reflexion |
| ***Was lief Heute positiv:***  ***Was lief nicht so gut:***  ***Fazit:*** |

* 1. Dienstag, 19.03.2024

|  |  |
| --- | --- |
| Tätigkeiten |  |
| **Erreichte Ziele** |  |

|  |
| --- |
| **Probleme** |
|  |
| Hilfestellung |
|  |
| Ausserplanmässige Arbeiten |
|  |
| Reflexion |
| ***Was lief Heute positiv:***  ***Was lief nicht so gut:***  ***Fazit:*** |

* 1. Mittwoch, 20.03.2024

|  |  |
| --- | --- |
| Tätigkeiten |  |
| **Erreichte Ziele** |  |

|  |
| --- |
| **Probleme** |
|  |
| Hilfestellung |
|  |
| Ausserplanmässige Arbeiten |
|  |
| Reflexion |
| ***Was lief Heute positiv:***  ***Was lief nicht so gut:***  ***Fazit:*** |

Teil 2 – Projekt

1. Kurzfassung
   1. Ausgangslage
   2. Umsetzung
   3. Ergebnis
2. Projektbeschreibung
   1. IPERKA

Diese Arbeit wird mit der IPERKA-Methode implementiert. Sie wird in sechs Schritte gegliedert: Informieren, Planen, Entscheiden, Kontrollieren und Auswerten. Die Bedeutung von Planung wird in dieser Methodik besonders hervorgehoben, um eine genaue Vorgehensweise zu fördern. Es ist die bevorzugte Projektmanagementmethode an der Berufsschule.

Das Projekt ist in die sechs Phasen unterteilt, um eine geordnete Struktur beizubehalten.

* + 1. Informieren

Die Ziele und Herausforderungen des Prozesses werden identifiziert. Es wird untersucht, welche Prozesse gelöst werden müssen, welche Ressourcen dafür gebraucht werden und Fragen werden geklärt. Dies ist wichtig für den Verstand der Aufgabe, sowie für die Erstellung des Zeitplans.

* + 1. Planen

Es wird ein detaillierter Plan erstellt, um einen strukturierten Projektablauf zu garantieren. Dies umfasst Festlegung von Aufgaben, Verantwortlichkeiten, sowie auch ein Realisierungskonzept und ein Testkonzept.

* + 1. Entscheiden

Der bestmögliche Lösungsweg wird ausgewählt in dem man die verschiedenen Ansätze zusammen vergleicht. Es ist wichtig sich zu überlegen, ob man die Herangehensweise auch implementieren kann und ob die Ideen überhaupt sinnvoll sind.

* + 1. Realisieren

Sobald alles klar definiert ist und alle Entscheidungen getroffen sind, wird mit der Realisierung begonnen. Die Arbeitsabläufe werden stetig protokolliert und die IST-Werte werden im Zeitplan eingetragen.

* + 1. Kontrollieren

Erst wenn die Realisierung vollständig implementiert wurde, werden die Resultate so getestet, wie es im Tesprotokoll steht. Fehlgeschlagene Tests werden entweder direkt korrigiert und behoben oder als Fehler vermerkt und dokumentiert.

* + 1. Auswerten

Als Abschluss gibt es eine Reflexion über die Arbeit und die Erfahrungen, die man gesammelt hat. Man geht durch all seine Schritte, welche man innerhalb des Projektes gemacht hat, und es wird untersucht was gut gelaufen ist und wo Verbesserungen möglich sind. Das ist auch praktisch für die Umsetzung zukünftiger Projekte.

1. Einleitung

In diesem Teil zeige ich die Umsetzung des Projektes mit der IPERKA-Methode auf. Die Kapitel sind in die Phasen unterteilt, um sich einfacher durch die Struktur des Dokumentes orientieren zu können.

1. Informieren
   1. Ziele der Aufgabenstellung

Das Ziel ist, dem Benutzer zu ermöglichen, die Kundenerstellung zu stoppen, um später fortzufahren. Wenn die Aufgabe die Frist von N-Tagen übersteigt, wird an den Benutzer eine E-Mail geschickt mit dem Link zu dieser Aufgabe, um zu fortfahren.

* 1. Vorgaben

Die Kundenerstellung soll bei einem beliebigen Punkt unterbrochen werden können. Die ausgefüllten Daten sollten in der Datenbank gespeichert werden ohne Validierung. Die Erstellung kann später weitergeführt werden. Sobald die Frist von N-Tagen bei einer Aufgabe abläuft, wird der Benutzer per E-Mail informiert.

* 1. Ist-Analyse

Die Offerten Verwaltung Web, kurz OVWEB, wird dazu dienen die alte Applikation in eine Cloud-basierte Webanwendung zu verwandeln. Das Programm ist gedacht für Kundenverwaltung und Erstellung von Offerten. Im jetzigen Stand der Applikation ist es möglich sich einzuloggen über den Keycloak-Server und Kunden erstellen via ein Formular. In der bestehenden Legacy Applikation mussten die Anwender die Erstellung von Kunden manchmal unterbrechen. Zum Beispiel mussten zusätzliche Abklärungen beim Kunden gemacht werden. Dieser war aber in diesem Moment nicht verfügbar. Bei einem teilerfassten Kunden gingen so die bisherigen Eingaben verloren, da die Validierung im halbfertigen Zustand der Erfassung eine Speicherung nicht zulässt. Diese Anforderung setze ich mir an, um den Unterbruch so einfach wie möglich zu gestalten.

* + 1. Frontend

Das Frontend wurde mit Next.js 14 umgesetzt und ist nach dem Generali-Styleguide gestaltet. Die Session wird mit NextAuth geregelt. Für Components wird die Shadcn Library benutzt. Für die Backend Kommunikation wird ein TypeScript-Client verwendet.

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

Automatisch generierte Beschreibung

Abbildung 1: Home-Page

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

Automatisch generierte Beschreibung

Abbildung 2: Kundenformular Ist Zustand

* + 1. Backend

Das Backend wurde mit dem Java-Framework Springboot realisiert. Es besteht ein Model, das User, Customer, Task und Adresse beinhaltet. Als Query-Sprache wird PostgreSQL verwendet. Die Datenbank Anbindung läuft via einen lokalen Docker-Container und wird durch Flyway versioniert.

* + 1. Überblick

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte BeschreibungIn der folgenden Abbildung wird die Architektur der OVWEB-Applikation dargestellt.

Abbildung 3: OVWeb-Struktur

* + 1. Klassendiagramme

[Screenshot Model]

[Screenshot Services]

* 1. User Stories

Um ein tieferes Verständnis der Anforderungen zu gewinnen, werden User Stories erfasst. Die Anforderungen der Benutzer und der Implementierung der neuen Workflow-Funktionalität liegen dabei im Vordergrund. Ich habe jedoch auch nicht-funktionale Anforderungen als User Stories zu hinunterschreiben, um einen besseren Überblick zu behalten über die IPA relevanten Kriterien. Somit kann ich auch beim Testen gleich überprüfen, ob die Entwicklerthemen vollständig umgesetzt wurden, sowie am Ende den Ist- und Soll-Zustand vergleichen.

* + 1. Rollen

Es gibt Benutzer (Berater, Versicherungsvertreter), Entwickler und Administratoren.

* + 1. Funktionale Anforderungen

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Frontend Button | 4 | 1 |
| Beschreibung |  |  |

Als ein **Benutzer** kann ich auf einen **Button klicken**, damit ich **den Fortschritt speichern kann**.

|  |
| --- |
| Acceptance Criteria |

**AC 1.1:** Auf der Customer Creation Page soll ein klar ersichtlicher «Später Fortfahren» Button ersichtlich sein.

**AC 1.2:** Die Daten werden in der Datenbank abgespeichert.

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Anzeige in Task Liste | 2 | 2 |
| Beschreibung |  |  |

Als ein **Benutzer** kann ich **nicht fertige Customer Creation Tasks sehen**, damit ich **einen Überblick habe über meine Tasks**.

|  |
| --- |
| Acceptance Criteria |

**AC 2.1:** Es sollte relevante Details wie Datum/Uhrzeit, Erfassungsobjekt und/oder den Namen des Kunden anzeigen.

**AC 2.2:** Jede unterbrochene Aufgabe in der Übersichtsliste sollte angeklickt werden können, um die Fortsetzung zu ermöglichen.

**AC 2.3:** Nachdem die konfigurierbare Frist von N-Tagen abgelaufen ist, wird die Aufgabe visuell vorgehoben.

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Formular Vorbefüllung | 4 | 3 |
| Beschreibung |  |  |

Als ein **Benutzer** kann ich auf den **«Fortsetzen» Button klicken**, damit ich **mit** **der Aufgabe weiterfahren kann**.

|  |
| --- |
| Acceptance Criteria |

**AC 3.1:** Auf der Customer Creation Page sollen die abgerufenen Daten ersichtlich und korrekt formatiert sein.

**AC 3.2:** Die Backend-Logik sollte unterbrochene Aufgaben aus der Datenbank abrufen

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Task Liste Sortierung | 2 | 4 |
| Beschreibung |  |  |

Als ein **Benutzer** kann ich auf den **auf das «Beschreibung» Feld klicken**, damit ich **die Tasks alphabetisch sortieren kann**.

|  |
| --- |
| Acceptance Criteria |

**AC 4.1:** Die Tasks werden von A-Z absteigend oder aufsteigend, wenn man doppelt klickt, angezeigt

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Task Liste Filter | 2 | 5 |
| Beschreibung |  |  |

Als ein **Benutzer** kann ich auf den **ein Begriff in das «Suchen» Feld eingeben**, damit ich **die Tasks filtrieren kann**.

|  |
| --- |
| Acceptance Criteria |

**AC 5.1:** Das Suchfeld erscheint auf der Task-Liste

**AC 5.2:** Es werden nur Tasks angezeigt, die die Sucheingabe beinhalten.

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| E-Mail versenden | 5 | 6 |
| Beschreibung |  |  |

Als ein **Benutzer** kann ich **per E-Mail über überfällige Aufgaben benachrichtigt werden**.

|  |
| --- |
| Acceptance Criteria |

**AC 6.1:** Nach dem die konfigurierbare Frist von N-Tagen abgelaufen ist, wird eine E-Mail an den Benutzer geschickt mit dem folgenden Inhalt: Ein Link zur Weiterführung des abgebrochenen Workflows.

**AC 6.2:** Der Link leitet auf die richtige Aufgabe weiter.

* + 1. Nicht funktionale Anforderungen:

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Usability | 1 | 7 |
| Beschreibung |  |  |

Als ein **Benutzer** möchte ich, dass **der Unterbruch benutzerfreundlich ist**, damit ich **ohne Hilfe die Aufgabe beenden kann**.

|  |
| --- |
| Acceptance Criteria |

**AC 7.1:** Die Anzeige ist in einer klar verständlichen Sprache, die Knöpfe sind klar beschriftet.

**AC 7.2:** Soll ein Fehler vorkommen, dann wird der Benutzer durch Hinweise darauf aufmerksam gemacht.

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Logging | 2 | 8 |
| Beschreibung |  |  |

Als ein **Administrator** möchte ich, dass **die neuen Funktionalitäten strukturiertes Logging implementieren**, damit **die Fehler einfacher zu beheben sind.**

|  |
| --- |
| Acceptance Criteria |

**AC 8.1:** Strukturiertes Logging wurde aktiv umgesetzt.

**AC 8.2:** Alle notwendigen Informationen wurden in den Logs enthalten, um die Aktionen nachvollziehen zu können.

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Tracing | 4 | 9 |
| Beschreibung |  |  |

Als ein **Entwickler** möchte ich, dass **jede Funktion mit einer Trace-Id versehen werden kann**, damit ich **die Fehlerbehebung vereinfacht werden kann**.

|  |
| --- |
| Acceptance Criteria |

**AC 9.1:** Bei jeder Frontend-Anfrage wird eine Trace-Id generiert, die dann an den Backend geschickt wird.

**AC 9.2:** Die Trace-Id wird konsquent durch alle Dienste hindurch weitergeleitet und in den Logs festgehalten.

**AC 9.3:** Es kann ein Diagramm der Service-Aufrufe erstellt werden, indem man die Trace-IDs als Grundlage verwendet.

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Unit Testing | 4 | 10 |
| Beschreibung |  |  |

Als ein **Entwickler** möchte ich, dass **die neuen Funktionalitäten durch automatisiertes Testing abgedeckt sind**, damit **ich die Wirksamkeit überprüfen kann**.

|  |
| --- |
| Acceptance Criteria |

**AC 10.1:** Für die neuen Funktionen wurden folgende Unit-Tests implementiert: Positiv- Negativszenarien, Null/Non-Null Tests für Datensätze, Empty/Not-Empty für Collections.

**AC 10.2:** Die Unit-Tests wurden im Backend, sowie im Frontend implementiert.

|  |  |  |
| --- | --- | --- |
| Titel der User Story | Schätzung in Stunden | User Story Nummer |
| Wiederverwendbarkeit | 2 | 11 |
| Beschreibung |  |  |

Als ein **Entwickler** möchte ich, dass **UI-Komponenten wiederverwendbar sind**, damit **ich sie in der Zukunft gebrauchen kann**.

|  |
| --- |
| Acceptance Criteria |

**AC 11.1:** Die Komponenten sind so aufgebaut, dass sie aus kleineren und einfacheren Komponenten bestehen.

1. Planen
   1. Realisierungskonzept
   2. Testkonzept
   3. Aktivitätsdiagramme
   4. Sequenzdiagramme
   5. Klassendiagramme
   6. Datenmodell
   7. Design
2. Entscheiden
   1. Entscheid
3. Realisieren
   1. Projektumgebung
   2. Datenbank
   3. Backend Unit Tests
   4. Frontend Unit Tests
4. Kontrollieren
   1. Testkonzept
   2. Testprotokoll Samuel Hajnik
5. Auswerten
   1. Vergleich Ist / Soll

Checkliste User Stories

* 1. Fazit zur IPA
  2. Reflexion

Danksagung

Glossar

|  |  |
| --- | --- |
| Begriff | Erklärung |
|  |  |

Abbildungsverzeichnis

Quellenverzeichnis