Individuelle Praktische Arbeit

**Programmierung für HelpDesk Applikation**

Eine Helpdeskz Projekt fürs verbessern von interner Kommunikation

Ege Bora Ulu

Flying Teachers

Löwenstrasse 20, 8001 Zürich

Inhaltsverzeichnis

[1 Teil 1 – Umfeld und Ablauf 5](#_Toc166256046)

[1.1 Allgemeine Informationen 5](#_Toc166256047)

[1.2 Titel der Arbeit 5](#_Toc166256048)

[1.3 Ausgangslage 5](#_Toc166256049)

[1.4 Detaillierte Aufgabenstellung 6](#_Toc166256050)

[1.5 Mittel und Methoden 7](#_Toc166256051)

[1.6 Vorkenntnisse 7](#_Toc166256052)

[1.7 Vorarbeiten 7](#_Toc166256053)

[1.8 Neue Lerninhalte 7](#_Toc166256054)

[1.9 Arbeiten in den letzten 6 Monaten 7](#_Toc166256055)

[1.10 Firmenstandards 7](#_Toc166256056)

[1.11 Projektaufbauorganisation 8](#_Toc166256057)

[1.11.1 Ausführungszeitraum 9](#_Toc166256058)

[1.11.2 Termine 9](#_Toc166256059)

[1.11.3 Vorkenntnisse 9](#_Toc166256060)

[1.11.4 Vorarbeiten 9](#_Toc166256061)

[1.11.5 Benützte Firmenstandards 9](#_Toc166256062)

[1.11.6 Arbeitspakete 11](#_Toc166256063)

[1.11.7 Meilensteine 12](#_Toc166256064)

[1.12 Dokumentablage 12](#_Toc166256065)

[1.12.1 Versionierung 12](#_Toc166256066)

[1.12.2 Backup 12](#_Toc166256067)

[1.12.3 Quellcode / Skripts 12](#_Toc166256068)

[1.13 Arbeitsplatz 12](#_Toc166256069)

[1.14 Arbeitsjournal 14](#_Toc166256070)

[1.14.1 Tag 1 – 25.04.2024 14](#_Toc166256071)

[1.14.2 Tag 2 – 26.04.2024 15](#_Toc166256072)

[1.14.3 Tag 3 – 29.04.2024 17](#_Toc166256073)

[1.14.4 Tag 4 – 30.04.2024 18](#_Toc166256074)

[1.14.5 Tag 5 – 02.05.2024 19](#_Toc166256075)

[1.14.6 Tag 6 – 03.05.2024 20](#_Toc166256076)

[1.14.7 Tag 7 – 06.05.2024 21](#_Toc166256077)

[1.14.8 Tag 8 – 07.05.2024 22](#_Toc166256078)

[1.14.9 Tag 9 – 08.05.2024 23](#_Toc166256079)

[1.14.10 Tag 10 – 10.05.2024 24](#_Toc166256080)

[1.15 Anmerkung zum Zeitplan 26](#_Toc166256081)

[1.15.1 Begründung für Abweichungen zum Zeitplan 26](#_Toc166256082)

[2 Teil 2 - Projekt 27](#_Toc166256083)

[2.1 Kurzfassung 27](#_Toc166256084)

[2.1.1 Ausganslage 27](#_Toc166256085)

[2.1.2 Umsetzung 27](#_Toc166256086)

[2.1.3 Ergebnis 27](#_Toc166256087)

[2.1.4 Projektumfeld und Systemgrenzen 27](#_Toc166256088)

[2.1.5 Gesamtsystem 27](#_Toc166256089)

[2.2 Einleitung IPERKA 28](#_Toc166256090)

[2.2.1 Informieren 28](#_Toc166256091)

[2.2.2 Planen 28](#_Toc166256092)

[2.2.3 Entscheiden 28](#_Toc166256093)

[2.2.4 Realisieren 28](#_Toc166256094)

[2.2.5 Kontrollieren 28](#_Toc166256095)

[2.2.6 Auswerten 28](#_Toc166256096)

[2.3 Phase «Informieren» 29](#_Toc166256097)

[2.4 Phase «Planen» 30](#_Toc166256098)

[2.5 Phase «Entscheiden» 34](#_Toc166256099)

[2.6 Phase «Realisieren» 35](#_Toc166256100)

[2.6.1 Aufsetzen des Programms 35](#_Toc166256101)

[2.6.2 Installation von Extension-Pakete 47](#_Toc166256102)

[2.6.3 Modul Updates 55](#_Toc166256103)

[2.6.4 Aufgabe Matching von Tickets 60](#_Toc166256104)

[2.6.5 Aufgabe: Antworten zu mehreren Personen 62](#_Toc166256105)

[2.7 Phase «Kontrollieren» 66](#_Toc166256106)

[2.7.1 UI-Tests 66](#_Toc166256107)

[2.7.2 Unit Tests 67](#_Toc166256108)

[2.8 Phase «Auswerten» 68](#_Toc166256109)

[2.8.1 Persönliche Reflexion 68](#_Toc166256110)

[2.8.2 Technisches Review 68](#_Toc166256111)

[2.8.3 Schlusswort 68](#_Toc166256112)

# Teil 1 – Umfeld und Ablauf

|  |  |
| --- | --- |
| **Dokumenttitel** | Programmierung für HelpDesk Applikation |
| **Autoren** | Ege Bora Ulu |
| **Dateiname** | Ege IPA |
| **Ablageort** | C:\Users\ege\Desktop\aw\_anweisung.docx |

## Allgemeine Informationen

## Titel der Arbeit

Programmierung für HelpDesk Applikation

## Ausgangslage

HelpdeskZ ist ein robustes OSS (Open Source Software) Projekt das sehr gut angepasst werden kann. Die Software wird von Flying Teachers seit 2023 eingesetzt für:

- internes uns externes Ticketing

- Verteilung von Aufgaben

- Aufbau einer Wissensdatenbank

- und nicht zuletzt für den Aufbau einer Datenbank von vorgefertigten Antworten.

HelpDeskZ unterstützt folgende Funktionalität die auchg weitgehend von Flying Teachers genutzt wird:

- Tickets können in Gruppen kategorisiert werden und so Agenten einer Gruppe zugewiesen werden

- Häufig genutzte Antworten, können in "Canned Responses" abgelegt und eingefügt werden. statt immer wieder die gleichen Antworten zu schreiben. So ermöglichen die "Canned Responses" für schnelle und einheitliche Antworten auf häufig gestellte Fragen

- Massen Updates: Müssen Sie mehrere Tickets demselben Agenten zuweisen? Mit Massenaktionen können Status, Zuweisung oder Löschen von hunderten Tickets schnell erledigt werden.

- Login-Freigabe: Das Login-Share-Modul ermöglicht die Integration von HelpDeskZ mit Anwendungen von Drittanbietern.

- Wissensdatenbank: Diese ermöglicht eine umfangreiche Wissensdatenbank für Kunden und Angestellte, um die Supportanfragen zu reduzieren.

- Nachrichten: Ein Abschnitt steht für Nachrichten und Meldungen an Kunden und Mitarbeiter zur Verfügung.

- E-Mail-Piping: Über eingehende E-Mails lassen sich Tickets erstellen. So auch von ganz unterschiedlichen Formularen.

- Die Möglichkeit, Tickets per E-Mail zu öffnen oder zu beantworten.

- Verwendung von Twig als Template-Engine, die eine einfache Änderung der Templates ermöglicht.

Für IPA soll eine Spezifikation erstellt werden. Darin soll ersichtlich sein welche Anpassungen notwendig sind und wie diese umgesetzt werden soll.

Im Rahmen dieses Projektes sollen die Anpassungen umgesetzt und getestet werden.

## Detaillierte Aufgabenstellung

Spezifikation für ein IT-Teilprojekt zur Erweiterung von HelpDeskZ

1. Projektziel

Das Ziel dieses Teilprojekts ist die Erweiterung der HelpDeskZ-Software um zwei neue Funktionen:

1. Möglichkeit zum Hinzufügen weiterer Empfänger beim Beantworten von Tickets.

2. Integration eingehender E-Mails in bestehende Tickets anstatt der Erstellung neuer Tickets.

Diese Erweiterungen sollen die Kommunikationseffizienz verbessern und die Bearbeitung von Kundenanfragen optimieren.

2. Funktionale Anforderungen

2.1 Hinzufügen weiterer Empfänger

• Benutzer können beim Beantworten eines Tickets zusätzliche E-Mail-Empfänger angeben.

• Alle Empfänger erhalten identische Antworten auf das Ticket.

2.2 Integration eingehender E-Mails in bestehende Tickets

• Eingehende E-Mails, die über das POP-Protokoll abgerufen werden, sollen anhand der Ticket-ID identifiziert und dem entsprechenden bestehenden Ticket zugeordnet werden.

• Die automatische Erstellung neuer Tickets bei Antworten auf bestehende Tickets soll unterbunden werden.

• Die neue Funktionalität für die Integration eingehender E-Mails betrifft die manuelle Ticketbeantwortung über die Plattform nicht.

3. Technische Anforderungen

3.1 Systemarchitektur und Integration

• Die neuen Funktionen sollen in die bestehende Architektur von HelpDeskZ integriert werden, ohne dass die Hauptfunktionen beeinträchtigt werden.

• Anwenden von Twig als Template-Engine für alle UI-Anpassungen.

3.2 Versionskompatibilität

• Der entwickelte Code muss kompatibel mit zukünftigen Versionen von HelpDeskZ bleiben.

• Nutzung von Git für Versionskontrolle und Code-Management.

• Dokumentation aller Änderungen zur Unterstützung der zukünftigen Integration und Wartung.

4. Entwicklungsprozess

4.1 Verständnis und Planung

• Analyse des bestehenden Source-Codes von HelpDeskZ.

• Erstellen eines UML Diagramms, das sich auf das "Matching" von aufennanderfolgenden Nachrichten bezieht. Folgende Klassen sollen darin vorkommen: Message, Ticket und User.

• Darstellung eines Tickets von der Erstellung bis zum Nachweisdokument im Work-Flow eines Unternehmens.

• Bewertung möglicher Implementierungsstrategien für die genannten neuen Funktionen.

4.2 Code-Entwicklung

• Entwicklung des Codes unter Beachtung der bestehenden Softwarearchitektur und Kodierungsstandards.

• Regelmässiges Committing der Änderungen in ein Git-Repository zur Nachverfolgung der Entwicklungsfortschritte.

• Codierung der Erweiterungen mit starkem Fokus auf die langfristige Wartbarkeit und Kompatibilität mit zukünftigen Software-Versionen.

4.3 Testing

• Entwicklung der Unit-Tests für beide neuen Funktionen, um ihre korrekte Funktionalität sicherzustellen.

• Einrichten von Testplänen zur Bewertung der neuen Funktionen bei zukünftigen Software-Updates.

• Durchführung von Regressionstests zur Überprüfung, dass vorhandene Funktionen nicht beeinträchtigt werden.

5. Dokumentation

• Erstellung einer umfassenden Dokumentation der Codeänderungen und der Logik hinter den neuen Funktionen.

• Dokumentation der Testverfahren und Ergebnisse.

6. Zeitplan

• 10 Tage gemäss Aufwand-Schatzung

• Abschlussbewertung von Ausführung und Resultat der praktischen Arbeit wie auch der Prüfungselemente Dokumentation, Präsentation und Fachgespräch

7. Bewertung und Auswahl

• Beurteilung der Implementierungsoptionen basierend auf den genannten Kriterien

## Mittel und Methoden

IPERKA. Sourcode, HelpDeskZ v2.0.2

## Vorkenntnisse

PHP, Java, UML

## Vorarbeiten

HelpDeskZ ist vorinstalliert und in einer Lauffähigen Version installiert.

SourceCode wurde zur Verfügung gestellt und in Github eingepflegt.

Der Kandidat hat Zugang zu den Anleitungen für Kandidaten.

## Neue Lerninhalte

Einblick in RFC 5321 und RFC 53212, Piping und Redirecting von Meldungen.

## Arbeiten in den letzten 6 Monaten

Powershell Scripting.

Unterhalt von CMS, Web-Conferencing Tools (Zoom BBB, Teams, Webex, Goto)

First Level Support für Lehrpersonen

## Firmenstandards

Keine

## Projektaufbauorganisation

| **Rolle** | **Name** | **Aufgabe und Verantwortung** |
| --- | --- | --- |
| Lehrbetrieb und Durchführungsort | Flying Teachers |  |
| Kandidatin,  Kandidat | Ege Bora Ulu |  |
| cBerufsbildnerin,  Berufsbildner | Christian Walder | Kann verschiedene, unterstützende und  administrative Aufgaben übernehmen, wertet  jedoch die IPA-Arbeit nicht aus. |
| Verantwortliche Fachkraft | Antonio L. Adrover | Entwickelt die Aufgabenstellung. Garantiert, dass  die KAND während der IPA ungestört arbeiten  können und beurteilt die erzielten Resultate der  Arbeit. Ist verantwortlich für einen reibungslosen  Ablauf der IPA |
| Hauptexpertin,  Hauptexperte | Roberto Ranieri | Besucht die Lehrfirma drei Mal und beurteilt die  erbrachte Leistung und Qualität der Arbeit. Er ist  für eine korrekte Umsetzung und Bewertung der  IPA verantwortlich. |
| Nebenexperte,  Nebenexpertin | Sven Frei | Ist für die korrekte Protokollierung der  Präsentation, der Demo und des Fachgesprächs |

### Ausführungszeitraum

Do 25. April 2024 Fr 26. April 2024

Mo 29. April 2024 Di 30. April 2024 Do 02. Mai 2024 Fr 03. Mai 2024

Mo 06. Mai 2024 Di 07. Mai 2024 Mi 08. Mai 2024 Fr 10. Mai 2024

### Termine

1. Expertenbesuch 26.04.2024, 11:15

2. Expertenbesuch -

Präsentation, Demonstration & Fachgespräch

### Vorkenntnisse

PHP, Java, UML

### Vorarbeiten

HelpDeskZ ist vorinstalliert und in einer Lauffähigen Version installiert.

SourceCode wurde zur Verfügung gestellt und in Github eingepflegt.

Der Kandidat hat Zugang zu den Anleitungen für Kandidaten.

### Benützte Firmenstandards

Keine

A graph of a graph with a line

Description automatically generated with medium confidence

### Arbeitspakete

| **#** | | **Name** | **Inhalt / Arbeitsresultate** |
| --- | --- | --- | --- |
|  | | | |
| 00 | Dokumentation | | Dokumentierung von dem ganzen Projekt. |
| 02 | Expertenbesuche | |  |
| 03 |  | |  |
| **Phase Informieren:** | | | |
| 04 | Informationsbeschaffung-E-Mail-Protokolle und Programme | | Da ich mit einem neuen Protokoll arbeiten werde, wird in diesem Paket Informationen gesammelt. |
| 05 | Kriterienkatalog analysieren | | Lesen und verstehen von das Kriterienkatalog ist wichtig, um das Aufgabe richtig durchsetzen zu können. |
| 06 | Analysieren von Source-Code | | Das Wissen und Verständnis über den Source-Code sind kritisch fürs Lösen der Aufgabe. |
| **Phase Planen:** | | | |
| 07 | Zeitplan erstellen | | Ein Zeitplan ist entscheidend, weil er die Organisation und fristgerechte Fertigstellung eines Projekts gewährleistet. Er hilft, den Fortschritt effektiv zu überwachen und sicherzustellen, dass alle Projektphasen termingerecht abgeschlossen werden. |
| 08 | UML-Diagramm für das „Matching“ von Nachrichten | | Ein UML-Diagramm wird erstellt um die Aufgabe besser zu Visualisieren und somit das Kandidat und VF mit der Aufgabe einverstanden sind. |
| 09 | Darstellung Ticket Work-Flow eines Unternehmens | | Darstellung eines Tickets von der Erstellung bis zum Nachweisdokument im Work-Flow eines Unternehmens, somit das Kandidat und VF mit der Aufgabe einverstanden sind. |
| 10 | Konzept erstellen davor, danach | | Ein Konzept wird erstellt, um eine Idee zu haben, wie das Programm am Ende aussehen sollte, dies sollte eine Kontrolle sein, dass VF und Kandidat sich einigen auf das Endresultat. |
| 11 | Isolation von Add-Ons | | Wie beschrieben in der Aufgabestellung wird eine Methode definiert für das |
| 12 | Testkonzept | | Ein Testkonzept wird benutzt, um zu definieren was getestet werden soll. |
| 13 | Entscheidung Lösungsweg Beantwortung mehrere Personen | | Eine Entscheidung zu treffen ist wichtig und um den Projekt einen weg Richtung zu geben |

### Meilensteine

| **Meilenstein** | **Geplanter Zeitpunkt** | **Bemerkungen** |
| --- | --- | --- |
| 13 | Tag 3 | Das Entscheiden, von wie das Projekt am Ende aussehen wird ist eine sehr wichtige Meilenstein. |
| 14 | Tag 3 |

## Dokumentablage

### Source Code Versionierung:

<https://github.com/XSAVAS/helpdeskz-ipa>

### Versionierung

Versionierung dieser Datei wird in einem Dateiserver und auf GitHub geführt.

|  |  |  |
| --- | --- | --- |
| **Tag** | **Datum** | **Branch** |
| 1 | 25.04.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-1 |
| 2 | 26.04.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-2 |
| 3 | 29.04.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-3 |
| 4 | 30.04.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-4 |
| 5 | 02.05.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-5 |
| 6 | 03.05.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-6 |
| 7 | 06.05.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-7 |
| 8 | 07.05.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-8 |
| 9 | 08.05.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-9 |
| 10 | 10.05.2024 | https://github.com/XSAVAS/IPA-Dok/tree/day-10 |

### Backup

Als Backup dieser Datei werde wird es auf meinem PC und in meiner Datenserver gespeichert.

### Quellcode / Skripts

Der Quellcode des Projekts wird auf meinem lokalen Gerät bearbeitet und auf GitHub versioniert. Dies ermöglicht es auch, meine Add-Ons für dieses Projekt umfassend zu dokumentieren. Alle meine Add-Ons werden zudem in diesem Dokument beschrieben, um klarzustellen, welche Änderungen vorgenommen wurden. Dies erleichtert zukünftige Weiterentwicklungen.

## Arbeitsplatz

Mein Arbeitsplatz ist so eingerichtet, dass ich effektiv mehrere Anwendungen gleichzeitig offenhalten kann. Dazu ist mein Laptop mit einem grossen Bildschirm verbunden, und ein weiterer Arbeits-PC steuert die anderen Bildschirme. Mit Hilfe von Microsoft Powertoys wurden die beiden PCs miteinander vernetzt. Dank dieser Einrichtung kann ich alle drei Bildschirme nutzen und über dieselbe Tastatur und Maus steuern, um stets Zugriff auf alle benötigten Anwendungen zu haben."

## Arbeitsjournal

### Tag 1 – 25.04.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 03 | Erstellen und Ausfüllen von bestimmten Abteilungen von Teil 1, damit ich schon mit Planung anfangen kann. Bei Fragen Notieren und am Expertenbesuch fragen. | Erledigt |
| 05 | Analyse von Kriterienkatalog und Aufgabenspezifische Kriterien | Erledigt |
| 07 | Ich habe den Zeitplan am ersten Tag erstellt, was zwar länger dauerte als erwartet, aber kein Problem darstellte, da ich die Flexibilität hatte, Änderungen vorzunehmen. | Erledigt |
| **Ausgeführte Arbeiten:** | | |
| Das Dokumentation Dokument erstellt, alle Priorität gesetzte Aufgaben erledigt, somit ich Problemlos weiterarbeiten und mit der zeit weiterdokumentieren kann, ein GitHub wurde für den Versionierung erstellt, werde Morgen den Experten fragen, ob dies benutzt, werden darf. Eine Erstellung von ein 10 Tage Langes Zeitplan wurde auch durchgeführt | | |
| **Aufgetretene Probleme** | | |
| Das Zeitplan dauerte länger zum Erstellen als erwartet, stellte aber kein Problem dar, da ich noch die Flexibilität hatte Änderungen vorzunehmen. | | |
| **Beanspruchte Hilfestellung** | | |
| Ich benötigte keine zusätzliche Unterstützung. eine Hilfestellung wurde | | |
| **Nacht- und Wochenendarbeiten** | | |
| Keine. | | |
| **Pendenzenliste** | | |
| * Fragen über Versionierung zur experte | | |
| **Reflexion** | | |
| Das Erstellen einer übersichtlichen Excel-Datei sowie das Vorausplanen erwiesen sich als herausfordernd und zeitaufwendig, insbesondere bei der Dateneingabe in die Excel-Liste. Letztendlich konnte ich diese Herausforderungen bewältigen, indem ich Farbcodierungen verwendete und die Spaltenanordnung anpasste. | | |
| **Vergleich mit Zeitplan** | | |
| Im Vergleich zur Erstellung des Zeitplans konnte ich alle Aufgaben, die ich für heute geplant hatte, bereits vor der Zeitplanerstellung erfolgreich abschliessen. | | |

### Tag 2 – 26.04.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 06 | Lesen und verstehen des Source-Codes, für eine effektivere Planungsphase. | Erledigt |
| 04 | Informieren über verschiedene E-Mail-Protokolle wie POP, RFC 5321 und RFC 53212 wie es in der Aufgabestellung beschrieben wurde. | Erledigt |
| 01 | Expertenbesuche, in diesem Besuch wurde ich von der Hauptexperte  Ranieri Roberto sehr gut mit meinen Fragen und potenzielle Fehler, die erstehen können, unterstützt. 11:15 ~ 12:10 | Erledigt |
| 08 | Erstellen von einem UML-Diagramm für «Matching» von Nachrichten. | Erledigt |
| 09 | Erstellen von einer Darstellung von Ticket Work-Flow eines Unternehmens. | Erledigt |
| **Ausgeführte Arbeiten:** | | |
| Zunächst wurde der Quellcode analysiert. Anschliessend informierte ich mich über E-Mail-Protokolle, um ein besseres Verständnis dafür zu entwickeln, wie ich die folgende Aufgabe lösen kann. Um 11:15 Uhr hatte ich meinen ersten Expertenbesuch mit meinem Vorgesetzten. Zudem wurden zwei Diagramme erstellt: ein UML-Diagramm und eine Darstellung des Ticket-Workflows eines Unternehmens. | | |
| **Aufgetretene Probleme** | | |
| Beim Planen von Meetings mit dem Vorgesetzten stellte sich heraus, dass eine gute Organisation entscheidend ist, damit alles reibungslos verläuft. Daher habe ich die Entscheidung getroffen, täglich kurze Meetings mit ihm abzuhalten. | | |
| **Beanspruchte Hilfestellung** | | |
| * ChatGPT für Diagramme * <https://diagrams.helpful.dev/s/s:dSXO4alf> * <https://diagrams.helpful.dev/s/s:fgA0B0FH> * <https://diagrams.helpful.dev/s/s:EFMy6m6c> * <https://diagrams.helpful.dev/s/s:MEsa7IS4> * <https://datatracker.ietf.org/doc/html/rfc5322> * <https://datatracker.ietf.org/doc/html/rfc5321> | | |
| **Nacht- und Wochenendarbeiten** | | |
| Keine | | |
| **Pendenzenliste** | | |
| Planen von Meetings in dem Zeitplan und ein Protokoll für Sie erstellen. | | |
| **Reflexion** | | |
| Heute fand das erste Expertentreffen statt, an dem ich gemeinsam mit meinem Hauptexperten, dem Vorgesetzten, und weiteren Experten teilnahm. Die Ratschläge meines Hauptexperten waren sehr hilfreich. Ein zentraler Punkt des Meetings war die Bedeutung der Kommunikation mit meinem Vorgesetzten, um Missverständnisse zu vermeiden. Mein Hauptexperte empfahl regelmässige Treffen – eine Empfehlung, die ich noch am selben Tag umsetzte. Dies erwies sich bereits als nützlich, da wir einen kleinen Fehler entdeckten und beheben konnten. | | |
| **Vergleich mit Zeitplan** | | |
| Heute lief alles wie geplant in dem Zeitplan. | | |

### Tag 3 – 29.04.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 10 | Erstellen von Konzepten, wie das Helpdeskz aussehen soll und wie es jetzt gerade ist. | Erledigt |
| 11 | Isolieren von Add-Ons, wie wird das Neu addierte Code isoliert und dokumentiert von dem Source-Code | Erledigt |
| 12 | Testkonzept | Erledigt |
| 13 | Lösungsweg Antworten zur Mehrer Personen | Erledigt |
| 14 | Lösungsweg festlegen Eingehender Mails in bestehende Tickets | Erledigt |
| **Ausgeführte Arbeiten:** | | |
| Ein Konzept von meinem Vorgesetzten wurde dokumentiert, und ein weiteres wurde speziell für die Aufgabe erstellt. Zudem führten wir ein kurzes Gespräch, um sicherzustellen, dass alles richtig verstanden wurde. Ein kleines Testkonzept wurde entwickelt, und alle notwendigen Tests wurden aufgezeichnet. Beide Lösungsansätze wurden festgelegt. Da ich den Lösungsweg bereits am Tag 2 mit meinem Vorgesetzten besprochen hatte, war vieles bereits klar. Diese Zeit nutzte ich für Notizen und die Dokumentation von Tag 3. | | |
| **Aufgetretene Probleme** | | |
| Manche Änderungen sind schwierig zu dokumentieren, insbesondere weil Modifikationen an der Benutzeroberfläche sichtbar sind, während Änderungen im Code oder in der Datenbank möglicherweise nicht direkt erkennbar sind. Deshalb werden kleinere Anpassungen, wie beispielsweise die Erstellung eines Tickets, in den Log-Dateien und der Datenbank nicht dokumentiert. Der Prozess der Ticket-Erstellung selbst wurde jedoch festgehalten. Dies liegt daran, dass ich keine Änderungen an der Art und Weise vornehme, wie Log-Einträge und Datenbankeinträge generiert werden. | | |
| **Beanspruchte Hilfestellung** | | |
|  | | |
| **Nacht- und Wochenendarbeiten** | | |
| Keine | | |
| **Pendenzenliste** | | |
| Keine | | |
| **Reflexion** | | |
| Heute wollte ich mit so viel wie möglich fertig sein damit ich morgen mit das Programmieren anfangen kann, ohne dass ich etwas nachholen muss. Das Konzept war von uns beide einverstanden, und es war auch klar mit welchen Methoden ich meine Aufgabe durchsetze, da wurde für das E-Mail lesen das Pop Protokoll Gewalt. | | |
| **Vergleich mit Zeitplan** | | |
| Alles lief nach der Zeitplan | | |

### Tag 4 – 30.04.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 15 | Mail Server | Erledigt |
| 16 | Programmieren erste Aufgabe | Aktiv |
| **Ausgeführte Arbeiten:** | | |
| Heute musste ich keinen Mail-Server aufsetzen, da ich bereits vor einigen Arbeitstagen Hilfe von meinem Vorgesetzten erhalten hatte und drei E-Mail-Adressen bekommen habe, die noch eingerichtet werden mussten. Diese wurden erfolgreich aufgesetzt. Das gesamte HelpdeskZ-System wurde installiert, die E-Mails sind angebunden, und kleine Tests wurden durchgeführt, um das Senden und Empfangen von Mails zu überprüfen. | | |
| **Aufgetretene Probleme** | | |
| Ich hatte verschiedene Probleme während das Aufsetzen der Source-Code und das Verbinden zu der Mail-Server danach Funktionierte alles | | |
| **Beanspruchte Hilfestellung** | | |
| * <https://docs.helpdeskz.com/en/latest/> * Chatgpt – chatverlauf in der Doku Teil * <https://www.infomaniak.com/en/support/faq/468/ports-protocols-for-messaging> * <https://www.google.com/search?q=xampp+change+php+version&rlz=1C1ONGR_deCH1035CH1035&oq=xammp++cha&gs_lcrp=EgZjaHJvbWUqCQgBEAAYChiABDIGCAAQRRg5MgkIARAAGAoYgAQyCQgCEAAYChiABDIJCAMQABgKGIAEMgkIBBAAGAoYgAQyCQgFEAAYChiABDIJCAYQABgKGIAEMgkIBxAAGAoYgAQyCQgIEAAYChiABDIJCAkQABgKGIAE0gEIMzg3NWowajeoAgCwAgA&sourceid=chrome&ie=UTF-8#fpstate=ive&vld=cid:5b45b1ae,vid:CnIXzKJqCqs,st:85> | | |
| **Nacht- und Wochenendarbeiten** | | |
| Keine | | |
| **Pendenzenliste** | | |
| Das Fertigstellen des Matchings und das Lesen von E-Mails sollten problemlos funktionieren – dafür ist Zeit morgen eingeplant, es besteht also keine dringende Pendenzen. | | |
| **Reflexion** | | |
| Heute hatte ich einige Schwierigkeiten bei der Vorbereitung des Quellcodes, um eine funktionierende Webseite zu erstellen. Was ich heute hätte verbessern können, wäre das gründlichere Studium des Quellcodes und eventuell das Notieren der Standorte der wichtigsten benötigten Dateien. Ich hatte das Gefühl, dass ich zu viel Zeit damit verbracht habe. | | |
| **Vergleich mit Zeitplan** | | |
| Heute war alles noch wie geplant, abhängend davon ob ich mein Problem mit Matching lösen kann, sollte alles in der Zeitplan fertig sein. Heute verlief alles noch nach Plan. Abhängig davon, ob ich mein Problem mit dem Matching lösen kann, sollte alles gemäss dem Zeitplan abgeschlossen sein. | | |

### Tag 5 – 02.05.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 16 | Einlesen von eingehendem E-Mail | Erledigt |
| **Ausgeführte Arbeiten:** | | |
| Heute habe ich das Matching der Aufgaben abgeschlossen. Zudem bin ich auf PHP-Version 8.2 zurückgewechselt, um sicherzustellen, dass das System weiterhin Sicherheitsupdates erhält und um Sicherheitsprobleme zu minimieren. | | |
| **Aufgetretene Probleme** | | |
| Ich hatte weiterhin Probleme mit dem Lesen von E-Mails, diese konnten jedoch durch Updates der Composer-Dateien gelöst werden. Das Herunterladen und die Nutzung von Composer sind im Dokument festgehalten. | | |
| **Beanspruchte Hilfestellung** | | |
| * ChatGPT – für das Einrichten und Ausführen von Befehlen mit Composer. | | |
| **Nacht- und Wochenendarbeiten** | | |
| Keine | | |
| **Pendenzenliste** | | |
| Das Durcheinander mit der Versionierung wurde behoben – das verwendete Dokument stammt von Tag 3. Ich habe einzelne Dokumente erstellt und diese auf GitHub versioniert. Zudem wurden alle Screenshots eingefügt und im Teil zur Realisierung beschrieben. | | |
| **Reflexion** | | |
| Ich muss besser auf die Versionierung meiner Dateien achten. ChatGPT kann sehr hilfreich sein beim Verstehen und Lösen von Problemen. Wenn ich nicht weiterkomme, ist es sinnvoll, eine Pause zu machen, vielleicht etwas zu essen und den Kopf für 10–20 Minuten freizubekommen, bevor ich weitermache. Das kann sehr hilfreich sein. Heute habe ich angemessene Pausen eingelegt und in der Zwischenzeit dokumentiert. | | |
| **Vergleich mit Zeitplan** | | |
| Im Vergleich zum Zeitplan bin ich momentan nicht mehr im Zeitrahmen. Dies kann jedoch durch schnelleres Arbeiten, insbesondere bei den meisten Front-End-Aufgaben, aufgeholt werden. | | |

### Tag 6 – 03.05.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 17 | Einfügen von mehreren Personen in der CC-Feld | Erledigt |
| 18 | Einlesen von Eingehende E-Mails | Erledigt |
| **Ausgeführte Arbeiten:** | | |
| Die Funktion zum Antworten auf E-Mails wurde hinzugefügt, und es wurde bereits eine kleine Kontrolle zur Überprüfung ihrer Funktionalität durchgeführt. Ich habe meinem Vorgesetzten gezeigt, wie eingehende Mails aussehen. Es sind keine Änderungen notwendig. | | |
| **Aufgetretene Probleme** | | |
| Es trat das Problem auf, dass man nicht mehreren Personen gleichzeitig E-Mails senden konnte, wenn die Adressen durch Kommas getrennt waren. Dies wurde jedoch ziemlich schnell behoben. | | |
| **Beanspruchte Hilfestellung** | | |
| - | | |
| **Nacht- und Wochenendarbeiten** | | |
| - | | |
| **Pendenzenliste** | | |
| - | | |
| **Reflexion** | | |
| Heute ist es mir gelungen, wieder in den Zeitplan zu kommen, indem ich die Implementierung der Antwortfunktion schneller abschliessen konnte. Zudem habe ich die Zustimmung meines Vorgesetzten erhalten, dass das Antwortsystem so funktioniert, wie es soll. | | |
| **Vergleich mit Zeitplan** | | |
| Heute bin ich wieder im Zeitplan. | | |

### Tag 7 – 06.05.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 19 | Das Aussehen von Anpassen, wenn nötig nach Gespräch mit VF | Erledigt |
| 20 | Dokumentieren von allen Tests, aufschreiben von was getestet wird. | Erledigt |
| 21 | Testfälle testen | X |
| **Ausgeführte Arbeiten:** | | |
| Heute habe ich das Aussehen dokumentiert sowie das Gespräch mit meinem Vorgesetzten über die vorgenommenen Änderungen. UI-Testfälle wurden dokumentiert, und ein Statuskonzept wurde definiert, das zuvor noch nicht festgelegt war. Der Beginn der Testung von Testfällen hat stattgefunden, allerdings gibt es noch einige, die nicht getestet wurden. Dafür sind morgen vier Stunden eingeplant. | | |
| **Aufgetretene Probleme** | | |
| Das Error Meldung beim einlesen von einer Mail mit POP. | | |
| **Beanspruchte Hilfestellung** | | |
| Noch keine über das POP-Fehlermeldung. | | |
| **Nacht- und Wochenendarbeiten** | | |
| Keine | | |
| **Pendenzenliste** | | |
| Das Abschliessen der Tests ist für morgen im Zeitplan vorgesehen. | | |
| **Reflexion** | | |
| Heute habe ich mit dem Testen begonnen und das Erstellen von Testfällen angegangen, auch wenn dies noch eher dem Planungsteil des Projekts zuzuordnen ist. Für mich war das Erstellen von Testplänen hier wichtig, da ich nach der Programmierung eine viel bessere Vorstellung davon habe, was getestet werden sollte und was nicht. Bisher lief das Testen gut. Ich werde die Tests morgen, am 07.05.2024, fortsetzen. Der umfangreich geplante Zeitrahmen für das Testing ergibt sich daraus, dass ich von meinen Klassenkameraden erfahren habe, sie hätten weniger Zeit dafür eingeplant und es war am Ende nicht ausreichend. Aus diesen Gründen habe ich aus den Fehlern meiner Klassenkameraden gelernt und liege somit noch im Zeitplan. | | |
| **Vergleich mit Zeitplan** | | |
| Im Vergleich zum Zeitplan bin ich heute gut unterwegs. Ich habe morgen vier Stunden Zeit, um die letzten Fehlermeldungen und nicht funktionierenden Fälle zu korrigieren. | | |

### Tag 8 – 07.05.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 21 | Testfälle erstellen | Erledigt |
| 22 | Mehrere Personen Antworten Auswertung | Erledigt |
| **Ausgeführte Arbeiten:** | | |
| Heute wurden Unit-Tests durchgeführt, und es erfolgte die Auswertung der Antworten von mehreren Personen. | | |
| **Aufgetretene Probleme** | | |
| Ich hatte beim Einrichten der PHP Unit Tests viele Probleme. Dafür habe ich mehrere Hilfestellungen in Anspruch genommen. Am Ende, nach einer langen Zeit, hatte ich funktionierende Unit Tests, nachdem ich meine Methode geändert hatte, wie ich die Unit Tests durchführe. | | |
| **Beanspruchte Hilfestellung** | | |
| * <https://www.w3resource.com/php/PHPUnit/a-gentle-introduction-to-unit-test-and-testing.php> | | |
| **Nacht- und Wochenendarbeiten** | | |
| Keine | | |
| **Pendenzenliste** | | |
| 23: Einlesen und Auswertung von E-Mails  Dokumentation von Unit Tests | | |
| **Reflexion** | | |
| Verglichen mit dem Zeitplan liege ich jetzt gerade noch hinten, aber ich hoffe, dass ich es mit einer schnelleren Dokumentation der Auswertung wieder schaffen kann. | | |
| **Vergleich mit Zeitplan** | | |
| Verglichen mit dem Zeitplan liege ich jetzt gerade noch hinten, aber ich hoffe, dass ich es mit einer schnelleren Dokumentation der Auswertung wieder in den Zeitplan schaffen kann. | | |

### Tag 9 – 08.05.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 23 | Auswertung Einlesen von E-Mails. | Erledigt |
| 24 | Auswertung Unit Tests | Erledigt |
| 25 | Reflexion | Erledigt |
| 02 | Rechtschreibung und Formatierung | Noch nicht Erledigt |
| **Ausgeführte Arbeiten:** | | |
| Die Auswertung des E-Mail-Einlesens wurde durchgeführt, da sie gestern nicht abgeschlossen werden konnte. Die Auswertung der Unit Tests wurde erledigt. Das Schreiben einer detaillierten Reflexion wurde ebenfalls abgeschlossen. Die Rechtschreibkontrolle des gesamten Dokuments steht noch aus. | | |
| **Aufgetretene Probleme** | | |
| Die Rechtschreibkontrolle dauert viel länger als erwartet. | | |
| **Beanspruchte Hilfestellung** | | |
| Sprachen Modelle z.B. Chatgpt | | |
| **Nacht- und Wochenendarbeiten** | | |
| Keine | | |
| **Pendenzenliste** | | |
| Das Erledigen von Rechtschreibung und Formatierung. | | |
| **Reflexion** | | |
| Ich hätte viel länger Zeit für das Rechtschreibung Planen sollen, jetzt im moment dauert es viel zu lange. | | |
| **Vergleich mit Zeitplan** | | |
| Verglichen mit dem Zeitplan habe ich gestern die zu spät erledigten Aufgaben nachgeholt, aber jetzt bin ich wieder zurückgeblieben, weil ich viel zu lange für die Rechtschreibung benötige.s | | |

### Tag 10 – 10.05.2024

| **Nr.** | **Tagesziele** | **Status** |
| --- | --- | --- |
| 26 | Fazit | Erledigt |
| 27 | Letzte Dokuemntkontrolle & Upload | Erledigt |
| **Ausgeführte Arbeiten:** | | |
| Die Kontrolle dauerte viel zu lange, und unter Stress muss ich noch das Fazit und die letzte Dokumentationskontrolle durchführen. | | |
| **Aufgetretene Probleme** | | |
| Das sehr lange dauern der Textkorrektur führte zu sehr viel Stress. | | |
| **Beanspruchte Hilfestellung** | | |
| - | | |
| **Nacht- und Wochenendarbeiten** | | |
| - | | |
| **Pendenzenliste** | | |
| - | | |
| **Reflexion** | | |
| Ich bin erleichtert, dass es endlich fertig ist, aber gleichzeitig auch traurig, dass mein Dokument nicht so gut geworden ist, wie es sein könnte, da ich zu lange für die Korrektur der Texte gebraucht habe. | | |
| **Vergleich mit Zeitplan** | | |
| - | | |

## Anmerkung zum Zeitplan

Der Zeitplan wurde zu Beginn der IPA erstellt und basiert auf Schätzungen. Im Verlauf des Projekts kann es zu Abweichungen der Zeiten, wie auch der Reihenfolge kommen. Zudem habe ich noch

Arbeitsschritte hinzugefügt. Die Änderungen, werden im folgenden Abschnitt genauer beschrieben.

### Begründung für Abweichungen zum Zeitplan

|  |
| --- |
| **Do 25.04.2024** |
|  |

|  |
| --- |
| **Fr 26.04.2024** |
| Es wurde eine kurze Abweichung vom Zeitplan vorgenommen, um mein erstes kurzes Meeting mit meinem Vorgesetzten Antonio zu führen, wie es vom Hauptexperten empfohlen wurde. Ab heute sind täglich 10-minütige Kommunikationsmeetings geplant, die so lange durchgeführt werden, wie Antonio Zeit dafür hat. Aus diesen Gründen wird dies zukünftig nicht mehr als eine Abweichung im Zeitplan vermerkt. |

|  |
| --- |
| **Mi 29.04.2024** |
|  |

|  |
| --- |
| **Do 30.04.2024** |
|  |

|  |
| --- |
| **Mo 02.05.2024** |
| Die Abweichung erfolgte aufgrund von Problemen beim Einlesen von E-Mails. |

|  |
| --- |
| **Di 03.05.2024** |
| Die Abweichung erfolgte aufgrund der vorherigen Aufgabe. |

|  |
| --- |
| **Mi 06.05.2024** |
|  |

|  |
| --- |
| **Do 07.05.2024** |
| Die Abweichung erfolgte aufgrund der langen Zeit, die benötigt wurde, um die Fehler zu beheben. |

|  |
| --- |
| **Mo 08.05.2024** |
|  |

|  |
| --- |
| **Di 10.05.2024** |
| Die Abweichung erfolgte aufgrund der sehr langen Dauer der Textkontrolle. |

# Teil 2 - Projekt

## Kurzfassung

### Ausganslage

HelpDeskZ ist eine robuste Open-Source-Software, die seit 2023 von Flying Teachers eingesetzt wird, um die interne und externe Kommunikation zu verbessern. Die Hauptfunktionalitäten umfassen internes und externes Ticketing, Aufgabenverteilung, Wissensdatenbankaufbau und die Erstellung vorgefertigter Antworten. Das Ziel dieses Teilprojekts ist die Erweiterung der HelpDeskZ-Software um zwei neue Funktionen: die Möglichkeit, weitere Empfänger beim Beantworten von Tickets hinzuzufügen, und die Integration eingehender E-Mails in bestehende Tickets. Diese Erweiterungen sollen die Kommunikationseffizienz verbessern und die Bearbeitung von Kundenanfragen optimieren.

### Umsetzung

Die Implementierung der neuen Funktionen erfolgte in mehreren Schritten. Zuerst wurde der bestehende Source-Code von HelpDeskZ analysiert, um die Struktur und verwendeten Technologien zu verstehen. Danach wurden die notwendigen Änderungen in die Software integriert, wobei Twig als Template-Engine für die UI-Anpassungen genutzt wurde. Der entwickelte Code wurde regelmässig in ein Git-Repository committet, um den Entwicklungsfortschritt nachzuverfolgen. Parallel dazu wurden umfassende Unit-Tests entwickelt, um die korrekte Funktionalität der neuen Features sicherzustellen.

### Ergebnis

Das Ergebnis der Implementierung ist eine verbesserte Version von HelpDeskZ, die nun die Möglichkeit bietet, beim Beantworten eines Tickets zusätzliche E-Mail-Empfänger anzugeben und eingehende E-Mails anhand der Ticket-ID zu identifizieren und bestehenden Tickets zuzuordnen. Diese neuen Funktionen wurden erfolgreich getestet und in das bestehende System integriert, ohne die Hauptfunktionen zu beeinträchtigen. Die Implementierung wurde als erfolgreich bewertet, da sie die gestellten Anforderungen erfüllt und die Effizienz der Kundenanfragenbearbeitung signifikant verbessert hat.

### Projektumfeld und Systemgrenzen

Das Projektumfeld umfasst die bestehende IT-Infrastruktur von Flying Teachers, in die HelpDeskZ integriert ist. Die Systemgrenzen wurden klar definiert, um sicherzustellen, dass die neuen Funktionen nahtlos in die vorhandene Softwarearchitektur integriert werden konnten. Dies beinhaltete die Nutzung der Twig-Template-Engine für UI-Anpassungen und die Sicherstellung der Kompatibilität mit zukünftigen Versionen von HelpDeskZ. Das Projektumfeld wurde sorgfältig dokumentiert, um potenzielle Schnittstellenprobleme zu vermeiden und eine reibungslose Integration zu gewährleisten.

### Gesamtsystem

Das Gesamtsystem besteht aus der erweiterten HelpDeskZ-Software, die nun durch die neuen Funktionen ergänzt wird. Die Systemarchitektur bleibt intakt, und die neuen Features wurden so integriert, dass sie die bestehende Funktionalität nicht beeinträchtigen. Das System wurde ausführlich getestet, um sicherzustellen, dass alle neuen und alten Funktionen reibungslos zusammenarbeiten. Die Dokumentation der Änderungen und der Implementierungsprozess stellen sicher, dass zukünftige Updates und Wartungen problemlos durchgeführt werden können.

## Einleitung IPERKA

### Informieren

In der Informationsphase geht es darum, sich ein umfassendes Bild von der Aufgabe zu machen. Wir sammeln alle notwendigen Informationen, um das Projektziel und die Anforderungen vollständig zu verstehen. Das bedeutet, dass wir die bestehende Dokumentation durchgehen, uns mit den relevanten Technologien und Methoden vertraut machen und offene Fragen durch Rücksprache mit Experten oder Stakeholdern klären. Ziel ist es, eine solide Grundlage zu schaffen, auf der wir unser Projekt aufbauen können. Nur wenn wir genau wissen, was von uns erwartet wird und welche Rahmenbedingungen es gibt, können wir erfolgreich planen und umsetzen.

### Planen

In der Planungsphase erstellen wir einen detaillierten Plan, der uns durch das Projekt führt. Wir definieren die einzelnen Schritte und Meilensteine, legen einen Zeitplan fest und identifizieren die benötigten Ressourcen. Dabei ist es wichtig, realistisch zu bleiben und mögliche Risiken und Hindernisse zu berücksichtigen. Ein guter Plan hilft uns, den Überblick zu behalten und sicherzustellen, dass wir alle Aufgaben fristgerecht und effizient erledigen. Zudem ermöglicht er uns, den Fortschritt regelmässig zu überprüfen und bei Bedarf Anpassungen vorzunehmen.

### Entscheiden

Nachdem wir einen Plan erstellt haben, müssen wir Entscheidungen treffen. Das bedeutet, dass wir aus den verschiedenen Lösungsansätzen diejenige auswählen, die am besten zu unseren Anforderungen und Ressourcen passt. Wir wägen die Vor- und Nachteile der verschiedenen Optionen ab und entscheiden uns für diejenige, die am effizientesten und effektivsten ist. Diese Phase ist entscheidend, da sie den weiteren Verlauf des Projekts massgeblich beeinflusst. Eine gut durchdachte Entscheidung legt den Grundstein für eine erfolgreiche Umsetzung.

### Realisieren

In der Realisierungsphase setzen wir unseren Plan in die Tat um. Wir beginnen mit der eigentlichen Arbeit und führen die geplanten Schritte aus. Dabei ist es wichtig, sich an den Plan zu halten und die Aufgaben strukturiert und methodisch abzuarbeiten. Während dieser Phase kann es auch notwendig sein, auf unvorhergesehene Probleme zu reagieren und Lösungen zu finden. Flexibilität und Anpassungsfähigkeit sind hier entscheidend, um das Projekt trotz eventueller Herausforderungen erfolgreich abzuschliessen.

### Kontrollieren

In der Kontrollphase überprüfen wir regelmässig den Fortschritt unseres Projekts. Wir vergleichen die aktuellen Ergebnisse mit den geplanten Zielen und Meilensteinen und stellen sicher, dass wir auf dem richtigen Weg sind. Sollten Abweichungen auftreten, analysieren wir die Ursachen und ergreifen Massnahmen, um wieder auf Kurs zu kommen. Diese Phase ist wichtig, um die Qualität unserer Arbeit zu gewährleisten und sicherzustellen, dass wir die Projektziele erreichen. Durch regelmässige Kontrollen können wir frühzeitig Probleme erkennen und korrigieren.

### Auswerten

Die Auswertungsphase bildet den Abschluss des Projekts. Hier reflektieren wir über den gesamten Projektverlauf und die erzielten Ergebnisse. Wir bewerten, ob die Ziele erreicht wurden und welche Methoden und Ansätze erfolgreich waren. Diese Reflexion hilft uns, aus unseren Erfahrungen zu lernen und unser Vorgehen bei zukünftigen Projekten zu verbessern. Die Auswertung ist ein wichtiger Schritt, um kontinuierlich besser zu werden und aus Erfolgen und Misserfolgen die richtigen Schlüsse zu ziehen.

## Phase «Informieren»

In der Informationsphase habe ich mich intensiv mit dem Source Code und dem Projekt HelpDeskZ auseinandergesetzt. Mein Hauptaugenmerk lag auf der Beschaffung von Informationen zu den E-Mail-Protokollen und den relevanten Programmen. Diese Phase wurde so ausgedehnt wie nötig, um sicherzustellen, dass ich alle wichtigen Informationen gründlich recherchiere und verstehe. Dazu gehörte die detaillierte Analyse der E-Mail-Protokolle und Programme, um ein umfassendes Verständnis für die Anforderungen und Funktionalitäten zu entwickeln. Ausserdem habe ich den bestehenden Source-Code von HelpDeskZ eingehend untersucht, um die Struktur und die verwendeten Technologien zu erfassen und mögliche Ansatzpunkte für die Implementierung neuer Funktionen zu identifizieren.

## Phase «Planen»

Aufgabenstellung übernommene Beschreibung VF:

A screenshot of a web page

Description automatically generated

A screenshot of a computer

Description automatically generated

Ist Lösungsweg-Modell:

A diagram of a user

Description automatically generated

Soll Lösungsweg-Modell:

Eine UML für das ungefähre Soll aussehen am Ende des Projekts.


Darstellung Ticket Workflow:

A diagram of a document

Description automatically generated

Isolation von Add-Ons

Alle Add-Ons und Änderungen in der Source-code die ich machen werde, wird systematisch ein Screenshot in der Teil Realisieren gemacht und erklärt, was das addierte code macht. Alle Änderungen werden auch auf GitHub Dokumentiert.

Testkonzept:

Für den Teskonzept habe ich ein Simples designt gewählt, jede Test bekommt eine Nummer und bekommt nach Testen ein Ergebnis über.

|  |  |  |
| --- | --- | --- |
| **ID** | **Testname** | **Ergebnis** |
|  |  |  |

## Phase «Entscheiden»

(Meilenstein) Die Gestaltung und Funktionalität des Endprodukts wurden in enger Absprache mit meinem Verantwortlichen Fachkraft Antonio definiert. Bereits am zweiten Projekttag entschied ich mich für die Nutzung der integrierten Protokollbehandlung von HelpDeskZ, speziell des POP-Protokolls. Diese Entscheidung traf ich, nachdem ich detaillierte Informationen zu verschiedenen Protokollen recherchiert hatte, wobei Antonio drei bereits konfigurierte E-Mail-Adressen zur Verfügung stellte. Diese Vorbereitung ermöglichte es mir, mich während der geplanten Projektphase ausschliesslich auf das Setup dieser Adressen zu konzentrieren.

Ein wesentlicher Aspekt der Kommunikationsoptimierung war die Einrichtung eines CC-Feldes für das Antworten an mehrere Personen. Dies ermöglicht es Benutzern, mehrere E-Mail-Empfänger durch Kommas getrennt hinzuzufügen, was die Effizienz der Ticketbearbeitung steigert.

Zusätzlich wurde die Methode des E-Mail-Matchings detailliert geplant. Basierend auf einem vorgesehenen Konzeptbild, wurden Zeitressourcen speziell für Anpassungen reserviert, um sicherzustellen, dass das Interface den Anforderungen entspricht und intuitiv von den Endnutzern bedient werden kann.

Für die lokale Entwicklung entschied ich mich gegen die Nutzung von Docker zugunsten von XAMPP, hauptsächlich aufgrund meiner Vorkenntnisse und der bereits bestehenden Installation auf meinem System. Diese Entscheidung beruhte auf der Benutzerfreundlichkeit und meiner Vertrautheit mit XAMPP, was mir eine effizientere und fehlerfreiere Entwicklungsumgebung bot.A screenshot of a computer

Description automatically generated

## Phase «Realisieren»

### Aufsetzen des Programms

Bevor ich mit dem Aufsetzen des Programms begann, besprach ich mit meiner Verantwortlichen Fachkraft (VF) die Notwendigkeit eines Mail-Servers, den ich zum Testen verschiedener Testfälle und für das Einrichten von Helpdeskz benötige. Er unterstützte mich, indem er folgende drei E-Mail-Adressen für mich erstellte:

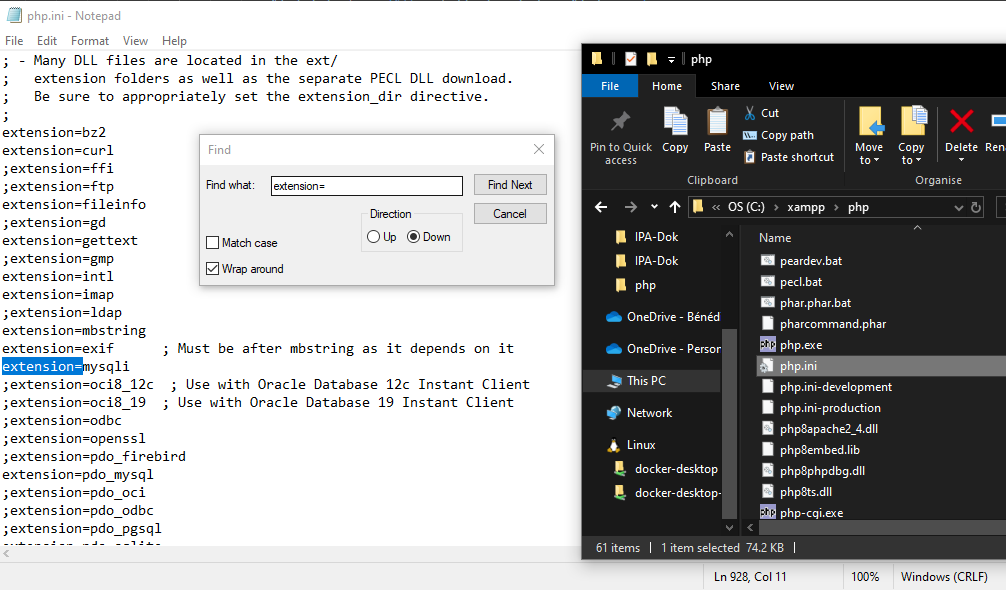
* [ticket\_agent@cras.ch](mailto:ticket_agent@cras.ch)
* [ticket\_client@cras.ch](mailto:ticket_client@cras.ch)
* [ticket\_user@cras.ch](mailto:ticket_user@cras.ch)

Dadurch konnte ich Zeit sparen und mit der Realisierung des „Matching“-Systems fortfahren. Wie bereits entschieden, begann ich mit dieser Aufgabe, weil ich anschliessend bereits mit dem E-Mail-System gearbeitet haben werde, was das Ausführen der Antwortfunktion vereinfacht.

Für die Konfiguration von Helpdeskz und das User-Interface musste ich zunächst den Anweisungen von Helpdeskz folgen. Da ich XAMPP bereits installiert hatte, musste ich lediglich die erforderlichen Erweiterungen für das Helpdeskz-Projekt aktivieren.

A screenshot of a computer

Description automatically generated

Im Zuge der Vorbereitung für die Implementierung von Helpdeskz war die Aktivierung spezifischer PHP-Extensions erforderlich. Dazu öffnete ich die Konfigurationsdatei **php.ini**, welche die Einstellungen für den PHP-Interpreter auf meinem lokalen XAMPP-Server enthält. Mittels der Suchfunktion des Texteditors identifizierte ich die Zeilen, die die benötigten Extensions betrafen. Durch das Entfernen des Semikolons am Anfang jeder relevanten Zeile deaktivierte ich die Kommentarfunktion, wodurch diese Extensions aktiviert wurden. Diese Änderungen wurden effektiv, nachdem ich den Apache-Server innerhalb der XAMPP-Kontrollpanel neu gestartet hatte. Dieser Schritt war entscheidend, um sicherzustellen, dass Helpdeskz mit allen erforderlichen PHP-Funktionalitäten arbeiten kann.

Nach der erfolgreichen Aktivierung der erforderlichen PHP-Extensions für das Helpdeskz-Projekt führte ich einen Neustart der Apache- und MySQL-Server durch, um die Änderungen zu übernehmen. Dies geschah über das XAMPP-Kontrollpanel, das einen einfachen Zugriff auf die Steuerung der Serverdienste bietet.A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

Nach dem Neustart der Server wurde eine Datei namens Test.php erstellt, die die phpinfo() Funktion enthält. Diese Funktion generiert eine ausführliche Übersicht aller aktuellen PHP-Konfigurationen und -Extensions. Die Test.php-Datei kann über die URL http://localhost/test.php im Webbrowser aufgerufen werden.

A screen shot of a computer

Description automatically generated

Durch die Verwendung der Suchfunktion (Ctrl-F) im Browser lässt sich innerhalb dieser Informationsseite gezielt nach bestimmten Begriffen suchen. Durch Eingabe von „enabled“ oder „disabled“ können spezifisch die aktivierten oder deaktivierten Extensions identifiziert werden. Dieser Schritt ist entscheidend, um zu verifizieren, dass alle für das Helpdeskz-Projekt erforderlichen PHP-Extensions korrekt geladen und einsatzbereit sind.A screenshot of a computer

Description automatically generated

Im nächsten Schritt folge ich der detaillierten Installationsanleitung für HelpDeskZ, um das System auf meinem Server einzurichten. Die Anleitung umfasst mehrere wichtige Schritte:

1. **Verbindung via FTP:** Ich verbinde mich über FTP mit dem öffentlichen Verzeichnis meines Servers, wo der Rest der Website gespeichert ist.
2. **Erstellung eines neuen Ordners:** Hier lege ich einen neuen Ordner an, in dem HelpDeskZ installiert wird, beispielsweise unter **/public\_html/support**.
3. **Hochladen der Dateien:** Alle Dateien von HelpDeskZ werden in diesen neuen Ordner hochgeladen.
4. **Konfiguration der Einstellungsdatei:** Die Datei **hdz/app/Config/Helpdesk.new.php** wird bearbeitet, um die notwendigen Informationen wie die URL der Seite und Daten zur Datenbank einzutragen. Anschliessend wird diese Datei in **Helpdesk.php** umbenannt.
5. **Ausführung des Installations-Skripts:** Durch Aufrufen von **/install** in meinem Browser starte ich das Installationsskript, das mich durch die Lizenzvereinbarung und die Einrichtung führt.
6. **Abschluss der Installation:** Nach Abschluss der Installation lösche ich den Ordner **/hdz/install** vom Server, um Sicherheitsrisiken zu minimieren.
7. **Einrichtung des Help Desks:** Ich öffne das Personalpanel unter **/support/staff** und logge mich mit den zuvor festgelegten Zugangsdaten ein.
8. **Zugriff auf die Einstellungen:** Über **Setup -> General** gelange ich zu den Einstellungsoptionen, um das System nach meinen Bedürfnissen zu konfigurieren.
9. **Vertrautmachen mit dem System:** Ich nehme mir Zeit, um alle verfügbaren Einstellungen kennenzulernen. Die meisten Optionen sind selbsterklärend.

Durch diese Schritte wird HelpDeskZ erfolgreich auf meinem Server installiert und konfiguriert, bereit für den Einsatz zur Unterstützung und Verwaltung von Benutzeranfragen.

Im Rahmen der Installation von HelpDeskZ war die Bearbeitung der Datei Helpdesk.new.php ein entscheidender Schritt. In dieser Konfigurationsdatei wurden wesentliche Einstellungen vorgenommen, die für die Funktionalität des Systems notwendig sind. Dazu gehören die Konfiguration der Datenbankverbindung, die Festlegung der URL der Webseite sowie die Einstellung der Standardsprache des Systems.

Für die Datenbankverbindung mussten nur minimale Änderungen vorgenommen werden, da ich als Benutzernamen root und kein Passwort verwende, was die Standardkonfiguration in XAMPP beim Start ist. Diese Voreinstellungen erleichterten den Konfigurationsprozess erheblich.

Untenstehend finden Sie einen Screenshot der Helpdesk.new.php-Datei, der die durchgeführten Einstellungen zeigt. Diese visuelle Darstellung hilft dabei, den erfolgreichen Abschluss dieser Konfigurationsphase zu dokumentieren und bietet einen klaren Überblick über die vorgenommenen Anpassungen.

A screen shot of a computer program

Description automatically generated

Nachdem alle Vorbereitungen abgeschlossen und die Konfigurationsdatei helpdesk.new.php zu helpdesk.php umbenannt wurde, kann der nächste Schritt der Installation von HelpDeskZ erfolgen. Durch die Eingabe des Links:

**http://localhost/helpdeskz-ipa/helpdeskz-ipa/install**

in den Browser gelangt man zur Installationsseite von HelpDeskZ. Wenn die vorherigen Schritte korrekt ausgeführt wurden, wird die Seite zur Installation oder zum Upgrade des Systems angezeigt. Diese Seite ist der Ausgangspunkt, um entweder eine Erstinstallation durchzuführen oder ein vorhandenes System auf die neueste Version zu aktualisieren.

Die korrekte Umbenennung der Konfigurationsdatei und das erfolgreiche Erreichen dieser Seite bestätigen, dass die bisherigen Konfigurationsschritte richtig durchgeführt wurden und das System bereit für die Installation ist.

A screenshot of a computer

Description automatically generated

Im nächsten Schritt der HelpDeskZ-Installation habe ich die notwendigen Informationen für das Administrationskonto eingegeben. Da dieses Konto in meiner Konfiguration sowohl für das Senden als auch für das Empfangen aller E-Mails zuständig ist, habe ich die E-Mail-Adresse **ticket\_agent@cras.ch** verwendet. Diese E-Mail-Adresse wird für alle administrativen Aufgaben und als primärer Kontakt für Benachrichtigungen innerhalb des HelpDeskZ-Systems genutzt.

Der Screenshot zeigt den Abschnitt des Installationsassistenten, in dem diese Informationen eingetragen werden. Hier füllte ich die Felder für den vollen Namen, die E-Mail-Adresse, den Benutzernamen und das Passwort aus, welches zur Sicherheit bestätigt wird. Nach der Eingabe aller erforderlichen Daten wird die Installation durch Klicken auf „Install HelpDeskZ“ abgeschlossen, was den Beginn der Konfiguration des Helpdesk-Systems markiert.

A screenshot of a login screen

Description automatically generated

Die Installation von HelpDeskZ ist erfolgreich abgeschlossen, wie der Bildschirm bestätigt. Es wird hervorgehoben, dass die Software nun vollständig installiert ist. Zusätzlich wird darauf hingewiesen, dass der Installationsassistent momentan gesperrt ist und die Datei **install.config** aus dem Cache-Verzeichnis entfernt werden muss, um fortzufahren.

Nach der Installation bietet die Webseite zwei Optionen: Einerseits kann man über den Button „Go to staff panel“ direkt zum Administrationsbereich wechseln. Dies ermöglicht sofortigen Zugriff auf die Verwaltungseinstellungen des Systems. Andererseits ermöglicht der Button „Go to HelpDesk home page“, zur Startseite von HelpDeskZ zu navigieren.

A screenshot of a computer error

Description automatically generated

Nach der Weiterleitung zur Login-Seite wird das zuvor erstellte Administrator-Konto für die erste Anmeldung verwendet. Diese initiale Anmeldung ist entscheidend, um die korrekte Einrichtung und Funktionsbereitschaft des HelpDeskZ-Systems zu bestätigen. Der erste Zugang prüft zudem die Benutzerfreundlichkeit und Funktionalität des Systems, um sicherzustellen, dass alle Konfigurationen wie vorgesehen funktionieren.

A cup of coffee with a design on top of it

Description automatically generated

A screenshot of a phone

Description automatically generatedIm Rahmen der Einrichtung von HelpDeskZ war eine der Schlüsselkomponenten die korrekte Konfiguration der E-Mail-Kommunikationsprotokolle, um eine effiziente Bearbeitung und Verwaltung der Support-Tickets zu ermöglichen. Für das Senden und Empfangen von E-Mails innerhalb des Systems habe ich mich entschieden, SMTP und POP zu verwenden, die beiden Hauptprotokolle für ausgehende und eingehende E-Mails.

A screenshot of a computer

Description automatically generated

A white rectangular object with a black border

Description automatically generated

Die Einrichtung der E-Mail-Funktionalitäten in HelpDeskZ, einschliesslich der Protokolle SMTP für das Senden und POP für das Empfangen von E-Mails, war ein wesentlicher Bestandteil der Systemkonfiguration. Entsprechend der Aufgabenstellung wurde der Mailserver **mail.infomaniak.com** verwendet. SMTP wurde auf Port 465 mit SSL-Verschlüsselung und POP auf Port 995 ebenfalls mit SSL-Verschlüsselung konfiguriert. Diese Ports bieten eine erhöhte Sicherheit und sind für die zuverlässige E-Mail-Kommunikation innerhalb des HelpDeskZ-Systems entscheidend.

Die Einhaltung der Sicherheitsprotokolle und die Wahl der Ports basieren auf den Richtlinien, die in der Dokumentation unseres Mailservers, verfügbar auf [Infomaniak's Messaging Ports and Protocols FAQ](https://www.infomaniak.com/en/support/faq/468/ports-protocols-for-messaging), detailliert beschrieben sind. Diese Dokumentation empfiehlt spezifische Einstellungen für optimale Sicherheit und Leistung.

Diese sorgfältige Konfiguration gewährleistet, dass HelpDeskZ effektiv für den operativen Betrieb vorbereitet ist und die E-Mail-Kommunikation im Support-System sicher und effizient durchgeführt werden kann.

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

Um sicherzustellen, dass alle E-Mail-Zeitstempel korrekt sind, habe ich die Zeitzone im HelpDeskZ-System angepasst. Diese Änderung war notwendig, um die Konsistenz der Zeitangaben in den E-Mails und im gesamten Ticket-System zu gewährleisten. Die korrekte Zeitzone stellt sicher, dass alle Kommunikations- und Aktionsprotokolle die tatsächliche lokale Zeit widerspiegeln, was für die Verwaltung der Supportanfragen und die Planung der Antwortzeiten entscheidend ist.

A screenshot of a computer

Description automatically generated

Während der Realisierungsphase habe ich Tests durchgeführt, um die Funktionalität des E-Mail-Versands zu überprüfen. Trotz korrekter Konfiguration wurden zunächst keine E-Mails versendet, was mich dazu veranlasste, nach möglichen Ursachen für dieses Problem zu suchen.

Ein signifikanter Hinweis kam aus einem Forum, in dem diskutiert wurde, dass das Projekt noch nicht auf PHP Version 8 oder höher getestet wurde. Daraufhin führte ich ein Downgrade auf PHP 7.2 durch und wiederholte die Tests, die jedoch weiterhin ohne Erfolg blieben. Nach einer eingehenden Überprüfung des Quellcodes stellte ich fest, dass ein zusätzliches Leerzeichen im Passwortfeld die Ursache für das Problem war. Nach der Korrektur dieses Fehlers konnte ich dennoch keine Tests durchführen, da zusätzlich technische Schwierigkeiten beim Mailserver von Infomaniak vorlagen.

Nachdem ich begonnen hatte, die bis dahin durchgeführten Arbeiten zu dokumentieren, wurde das Problem mit dem Mailserver innerhalb einer halben Stunde behoben, und ich konnte endlich erfolgreich bestätigen, dass alle E-Mails nun problemlos gesendet wurden.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

### Installation von Extension-Pakete

Im Zuge der kontinuierlichen Verbesserung und Sicherstellung der Sicherheit und Stabilität meines Projekts habe ich Composer installiert, um die Verwaltung von PHP-Abhängigkeiten effizient zu handhaben. Dieser Schritt ermöglichte es mir, alle notwendigen Bibliotheken und Pakete einfach zu aktualisieren und zu verwalten.

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

Automatisch generierte Beschreibung

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

Automatisch generierte Beschreibung

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

Automatisch generierte Beschreibung

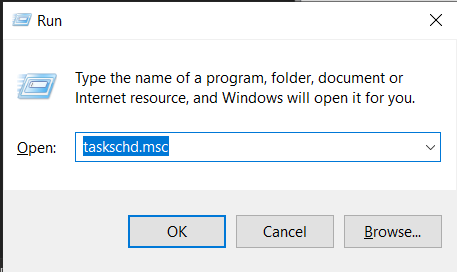
Während der Entwicklung meines Projekts war es notwendig, die PHP-Version anzupassen, um Kompatibilitäts- und Sicherheitsanforderungen zu erfüllen. Anfänglich wechselte ich die PHP-Version zu 7.2, um spezifische Probleme und Anforderungen zu adressieren. Dieser Wechsel erforderte das Überschreiben der vorhandenen PHP-Verzeichnisse im XAMPP-Ordner mit den Dateien der Version 7.2. Nachdem ich festgestellt hatte, dass die gewünschten Funktionen auch unter einer aktuelleren Version lauffähig sind, stellte ich erneut auf PHP 8.2.12 um, um von den neuesten Sicherheitsupdates und Leistungsverbesserungen zu profitieren.

Für beide Versionen musste ich die Konfiguration in der **php.ini** Datei entsprechend anpassen. Dies beinhaltete die Aktualisierung von Einstellungen wie **memory\_limit**, **upload\_max\_filesize** und die Anpassung von Modulen, die durch das Ein- oder Auskommentieren von Erweiterungen aktiviert wurden. Diese Schritte waren entscheidend, um sicherzustellen, dass die Umgebung optimal konfiguriert war und die Anwendung korrekt und sicher auf dem neuesten technischen Stand funktioniert.

Zum Automatisieren des E-Mail-Leseprozesses durch das IMAP-Protokoll wurde eine Windows Task im Scheduler eingerichtet. Diese Aufgabe wurde so konfiguriert, dass sie jede Minute ausgeführt wird, um sicherzustellen, dass alle eingehenden E-Mails zeitnah verarbeitet werden. Dabei wird das Skript index.php im Pfad C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\ mit dem Argument imap\_fetcher aufgerufen. Dies ermöglicht das regelmässige Abrufen von E-Mails über das IMAP-Protokoll, wobei speziell darauf geachtet wurde, die Windows-Umgebung für die Ausführung der Cron-Jobs anzupassen, die traditionell auf Unix-basierten Systemen eingesetzt werden.

Die Konfiguration im Windows Task Scheduler erforderte die genaue Definition des Ausführungspfads und der Skriptparameter, um die Funktionalität des HelpDeskZ-Systems mit der E-Mail-Funktion nahtlos zu integrieren. Für die Entscheidung, wie diese Aufgabe optimal zu konfigurieren war, habe ich Unterstützung von ChatGPT gesucht. Nach einer detaillierten Beschreibung meines Systems konnte ChatGPT relevante Anweisungen und technische Details bereitstellen, die mir erheblich geholfen haben, die Aufgabe korrekt einzurichten.

Das starten von Windows Task Manager:



Das Erstellen eines neues Tasks:

A screenshot of a computer task schedule

Description automatically generated

**Generelle-Einstellungen:** Der Name und wann die Aufgabe laufen darf wird hier eingegeben.

A screenshot of a computer

Description automatically generated

**Trigger:** Das Aufgabe wird ab heute jede Minute aufgeführt, um sicherstellen, dass E-Mails egelmassig abgerufen werden.

A screenshot of a computer

Description automatically generated

**Aktion:** Hier wird das Aktion **-f "C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\index.php" imap\_fetcher** ausgeführt, mit der Nutzung von **php.exe**.

A screenshot of a computer program

Description automatically generated

### Modul Updates

**ERROR** - 2024-05-02 09:08:49 --> IMAP connection failed: *InvalidArgumentException*: Argument 1 passed to PhpImap\Imap::search must be a valid resource! in C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Imap.php:1077

Stack trace:

#0 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Imap.php(1094): PhpImap\Imap::EnsureResource(Object(IMAP\Connection), 'PhpImap\\Imap::s...', 1)

#1 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Imap.php(821): PhpImap\Imap::EnsureConnection(Object(IMAP\Connection), 'PhpImap\\Imap::s...', 1)

#2 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Mailbox.php(621): PhpImap\Imap::search(Object(IMAP\Connection), 'ALL', 1, 'UTF-8')

#3 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\app\Libraries\MailFetcher.php(38): PhpImap\Mailbox->searchMailbox('ALL')

#4 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\app\Controllers\MailFetcher.php(16): App\Libraries\MailFetcher->parse\_imap()

#5 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\framework\CodeIgniter.php(936): App\Controllers\MailFetcher->imap()

#6 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\framework\CodeIgniter.php(432): CodeIgniter\CodeIgniter->runController(Object(App\Controllers\MailFetcher))

#7 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\framework\CodeIgniter.php(333): CodeIgniter\CodeIgniter->handleRequest(NULL, Object(Config\Cache), false)

#8 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\index.php(44): CodeIgniter\CodeIgniter->run()

#9 {main}

Next PhpImap\Exceptions\*ConnectionException*: Argument 1 passed to PhpImap\Imap::search must be valid resource! in C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Imap.php:1096

Stack trace:

#0 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Imap.php(821): PhpImap\Imap::EnsureConnection(Object(IMAP\Connection), 'PhpImap\\Imap::s...', 1)

#1 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Mailbox.php(621): PhpImap\Imap::search(Object(IMAP\Connection), 'ALL', 1, 'UTF-8')

#2 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\app\Libraries\MailFetcher.php(38): PhpImap\Mailbox->searchMailbox('ALL')

#3 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\app\Controllers\MailFetcher.php(16): App\Libraries\MailFetcher->parse\_imap()

#4 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\framework\CodeIgniter.php(936): App\Controllers\MailFetcher->imap()

#5 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\framework\CodeIgniter.php(432): CodeIgniter\CodeIgniter->runController(Object(App\Controllers\MailFetcher))

#6 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\framework\CodeIgniter.php(333): CodeIgniter\CodeIgniter->handleRequest(NULL, Object(Config\Cache), false)

#7 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\index.php(44): CodeIgniter\CodeIgniter->run()

#8 {main}

Da ich meine VSC nach der Installation der Composer noch nicht neu gestartet hatte bekam ich als erstes diesen Fehler:

A black screen with yellow and green text

Description automatically generated

Hier wurde der Name der **composer.json** geändert, somit es mit der richtigen Namenkonzept ein Update durchgeführt werden kann.

A computer screen with text

Description automatically generated

Änderung der **name:** , dieses Feld braucht einen Namekonzept wie unten sichtbar:

A screen shot of a computer program

Description automatically generated

In diesem Fall habe ich ChatGPT gefragt, was man hier machen könnte und welche Optionen vorhanden sind. Es gab mir zwei Möglichkeiten: das Update von **composer.json** und die Installation neuer Versionen der Pakete oder das Zurücksetzen der PHP-Version. Da jedoch die alte PHP-Version 7.2 nicht mehr unterstützt wird und keine Sicherheitsupdates mehr erhält, habe ich mich für die erste Option entschieden.

A screen shot of a computer program

Description automatically generated

Version Änderung erste Modul:

A screen shot of a computer

Description automatically generated

Das 2. Problem ist jetzt gelöst:

A screen shot of a computer

Description automatically generated

Version Änderung zweite Modul:

A screen shot of a computer

Description automatically generated

Dies führte zu einem Update von allen Modulen:

A screenshot of a computer program

Description automatically generated

Dieser Vorgang mit dem Updates mit Composer wurde noch ein paar Mal durchgeführt, bis alle Pakte aktuell waren.

Somit hat das Composer Programm alle Pakete sichtbar auf dem Bild aktualisiert da wollte ich schon ein Test machen ob das Pop Protokoll funktioniert.

Obwohl ich das unterstehende Fehlermeldung bekam, wurde mein Mail Problemlos gelesen und eiene Neue Ticket wurde erstellt:

**CRITICAL** - 2024-05-02 10:12:37 --> IMAP\Connection is already closed

#0 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Mailbox.php(479): imap\_ping(Object(IMAP\Connection))

#1 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Mailbox.php(544): PhpImap\Mailbox->hasImapStream()

#2 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\vendor\php-imap\php-imap\src\PhpImap\Mailbox.php(187): PhpImap\Mailbox->disconnect()

#3 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\app\Controllers\MailFetcher.php(16): PhpImap\Mailbox->\_\_destruct()

#4 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\framework\CodeIgniter.php(936): App\Controllers\MailFetcher->imap()

#5 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\framework\CodeIgniter.php(432): CodeIgniter\CodeIgniter->runController(Object(App\Controllers\MailFetcher))

#6 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\framework\CodeIgniter.php(333): CodeIgniter\CodeIgniter->handleRequest(NULL, Object(Config\Cache), false)

#7 C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\index.php(44): CodeIgniter\CodeIgniter->run()

#8 {main}

Von diesem Kode verstehe ich das der IMAP Verbindung doppelt abgeschlossen wird, da aber es schon Mails Problemlos lesen kann, habe ich mich dafür entschieden den Programmierung weiterzuführen, somit ich noch in meine Zeitplan bleiben kann.

### Aufgabe Matching von Tickets

In meinem jüngsten Update der **parseToTicket**-Methode des HelpDeskZ-Systems habe ich eine entscheidende Logik eingeführt, die prüft, ob eingehende E-Mails als neue Tickets angelegt oder bestehenden Tickets zugeordnet werden sollen. Diese Anpassung verbessert die Flexibilität und Effizienz der Ticketverwaltung, indem sie sicherstellt, dass alle relevanten Informationen korrekt erfasst und verarbeitet werden.

**Detaillierte Beschreibung der Anpassung:**

1. **Einrichtung der Funktion:**
   * **Client- und Ticketidentifizierung:** Die Funktion beginnt mit dem Abrufen des Kunden-IDs mittels **getClientID**, gefolgt von einer Überprüfung, ob ein entsprechendes Ticket existiert (**getTicketFromEmail**). Die Parameter für die Client- und Betreffzeile werden standardisiert, falls sie leer sind (**NoName** für den ClientNamen und **n/a** für den Betreff).
2. **Logik für neue und bestehende Tickets:**
   * **Erstellung neuer Tickets:** Falls kein passendes Ticket gefunden wird, erstellt die Funktion ein neues Ticket mit der Methode **createTicket**. Anschliessend wird eine Nachricht zum Ticket hinzugefügt (**addMessage**) und das Ticket abgerufen (**getTicket**).
   * **Aktualisierung bestehender Tickets:** Falls bereits ein Ticket existiert, wird das bestehende Ticket-Objekt verwendet, um eine neue Nachricht hinzuzufügen und das Ticket entsprechend zu aktualisieren (**updateTicketReply**). Zusätzlich wird eine automatische Antwortbenachrichtigung ausgelöst (**autoResponseNotification**), um den Kunden sofort über die Bearbeitung seines Anliegens zu informieren.
3. **Kommentierung und Dokumentation:**
   * Ich habe den Code ausführlich kommentiert, um die Logik hinter jedem Schritt zu erklären und die Nachvollziehbarkeit der Anpassungen zu erleichtern. Dies erleichtert zukünftige Wartungsarbeiten und stellt sicher, dass andere Entwickler die Funktion und ihre Mechanismen leicht verstehen können.
4. **Benachrichtigungen und Rückgaben:**
   * Unabhängig vom Pfad der Logik wird eine Benachrichtigung an das Personal gesendet (**staffNotification**), um sicherzustellen, dass das Team über neue Entwicklungen oder Updates im Ticketstatus informiert ist. Die Funktion gibt schliesslich die IDs des Tickets und der Nachricht zurück, was weitere Interaktionen ermöglicht.

A screen shot of a computer program

Description automatically generated

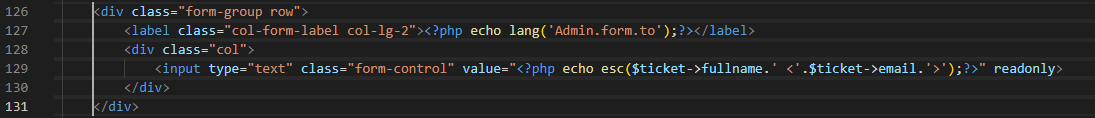
**Zweck und Bedeutung:**

Durch diese sorgfältig konzipierte Anpassung werden Effizienz und Reaktionsfähigkeit des HelpDeskZ-Ticketsystems erheblich verbessert. Die detaillierte Kommentierung unterstützt dabei die Wartbarkeit und fördert das Verständnis der Prozesse innerhalb des Teams.

### Aufgabe: Antworten zu mehreren Personen

Im Rahmen der Verbesserung der Kommunikationsmöglichkeiten innerhalb unseres Ticketing-Systems habe ich die Aufgabe bekommen, die Möglichkeit zu implementieren, Antworten an mehrere Personen (CC) zu senden. Dies war notwendig, um die Funktionalität unseres Systems an gängige E-Mail-Standards anzupassen und die Kommunikationseffizienz zu verbessern.

Ich habe das bestehende Feld für die Anzeige des normalen Absenders kopiert und angepasst, um ein zusätzliches Feld für CC-Adressen zu integrieren.



Für die Gestaltung verwendete ich Bootstrap-Klassen (**col-form-label col-lg-2** für das Label und **form-control** für das Eingabefeld), um die Konsistenz mit dem bestehenden Design zu wahren.

A computer screen with text

Description automatically generated

So sah das nach die vorherigen änderungen aus:

A screenshot of a computer

Description automatically generated

A black screen with white text

Description automatically generatedDie Anpassung der admin.php unter den Spracheinstellungen ermöglichte es, das Kürzel cc als "CC" einzuführen. Diese Modifikation erleichtert zukünftige Übersetzungen und Anpassungen ohne die Notwendigkeit, den gesamten Code zu überarbeiten.

A screen shot of a computer program

Description automatically generated

Die Platzierung des CC-Feldes unter dem **To**-Feld wurde durch meine Verantwortliche Fachkraft (VF) vorgeschlagen und als sinnvoller erachtet, da es der üblichen Darstellung in E-Mails entspricht. Ich habe daraufhin das CC-Feld entsprechend umpositioniert, um die Benutzerfreundlichkeit zu verbessern und das Design intuitiver zu gestalten.A screenshot of a computer

Description automatically generated

In der **hdz/app/controllers/staff/tickets.php**-Datei habe ich die Logik implementiert, um die im CC-Feld eingegebenen Daten zu erfassen und zu verarbeiten. Dies beinhaltet das Abrufen der Daten mit **$this->request->getPost('cc')**.

A screen shot of a computer program

Description automatically generated

Die **replyTicketNotification**-Methode wurde angepasst, um die CC-Informationen in den Benachrichtigungen zu berücksichtigen. Ich führte eine neue Variable **$cc** ein, die in den Email-Template-Funktionen verwendet wird, um sicherzustellen, dass CC-Adressen korrekt in die ausgehenden E-Mails eingefügt werden.

A black background with white text

Description automatically generated

Die **sendFromTemplate**-Methode im HelpDeskZ-System wurde erweitert, um die Unterstützung von CC-Adressen in den E-Mail-Versand einzubeziehen. Diese Funktion ermöglicht es, zusätzliche Empfänger automatisch in die Kommunikation einzubinden, was besonders in Szenarien nützlich ist, in denen eine Information breit gestreut werden muss oder mehrere Abteilungen über den Stand eines Tickets informiert bleiben sollen.A screen shot of a computer code

Description automatically generated

A screen shot of a computer code

Description automatically generated

Nach gründlicher Überprüfung der «replyTicketNotification»-Methode unter der «File» habe ich eine neue Variable ($cc) eingeführt.

A computer screen shot of text

Description automatically generated

Addieren von $ cc in public fuction sendFromTemplate

A screen shot of a computer program

Description automatically generated

Danach gab es ein Problem was beim Kontrollieren(link zur Title kontorllieren) gelöset wurde

A screen shot of a computer code

Description automatically generated

**E-Mail Beispiele CC:**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

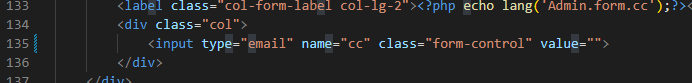
## Phase «Kontrollieren»

### UI-Tests

Testfälle werden in diesem Testplan nummeriert und Beschrieben, die Ergebnisse sind schon hier Sichtbar, das Feld Ergebnis hat das folgende Format Anzahl Probleme/Fehler | Gelöst Ja/Nein z.B. 3 | Ja

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Testname** | **Anz. Fehler | Gelöst Ja/Nein** | |
| 00 | Test-E-Mail-Sender zur Mehrere Personen | 1 | Ja |
| 01 | Antworten über der Webseite | 0 | Ja |
| 02 | Antworten zu mehreren Personen | 0 | Ja |
| 03 | Test POP: werden Mails gelesen? | 1 | Tag8: Ja |
| 04 | Test POP: werden Mails gelöscht? | 1 | Tag8: Ja |
| 05 | Das Matching von Mail | 0 | Ja |
| 06 | Matching von Personen | 0 | Ja |
| 07 | Matching von Departments | 0 | Tag8: Ja |

Fehler bei Feld, hatte keine Value:



### Unit Tests

**Klarheit und Wartbarkeit: Trennung von Testcode und Produktionscode**

Um eine klare Struktur in unserem Projekt zu gewährleisten und die Wartbarkeit zu verbessern, habe ich den Unittest-Code strikt vom Produktionscode getrennt. Diese Praxis erhöht nicht nur die Übersichtlichkeit des Codes, sondern erleichtert auch dessen Verwaltung und Verständnis. Durch die separate Aufbewahrung des Testcodes vermeiden wir Seiteneffekte auf die Produktionsumgebung und stellen sicher, dass der Produktionscode schlank und effizient bleibt.

**Standardpraxis: Einklang mit den Standards der PHP-Community**

Die Trennung von Test- und Produktionscode entspricht den gängigen Praktiken innerhalb der PHP-Community. Diese Konvention fördert die Kollaboration, da sie anderen Entwicklern, die möglicherweise zum Projekt beitragen, das Verständnis und die Navigation in der Projektstruktur erleichtert.

**Integration mit Werkzeugen: Vereinfachung der Integration mit CI/CD-Tools**

Die klare Trennung und Standardisierung der Testumgebung vereinfacht auch die Integration mit Continuous Integration (CI) und Continuous Deployment (CD) Tools. Diese Werkzeuge erwarten häufig eine standardisierte Struktur, um automatisch Tests durchführen und Ergebnisse melden zu können. Die Einrichtung in einer bekannten und akzeptierten Struktur erleichtert die Automatisierung der Testprozesse erheblich und unterstützt eine robuste Entwicklungspraxis.

Durch die Umsetzung dieser Massnahmen habe ich eine klare und funktionale Testumgebung eingerichtet, die mit dem Projekt skalieren kann. Diese Einrichtung gewährleistet, dass die Tests leicht zu verwalten und sowohl lokal als auch in zukünftigen CI/CD-Pipelines problemlos ausführbar sind.

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Testname** | **Anz. Fehler | Gelöst Ja/Nein** | |
| 00 | Testen von CC | 0 | Ja |
| 01 | Malching von E-Mails | 1 | Nein |

Path zu Unit-Test dateien: C:\xampp\htdocs\helpdeskz-ipa\helpdeskz-ipa\hdz\tests\unit\

(Screenshots und Erklärungen)

A screen shot of a computer

Description automatically generated

## Phase «Auswerten»

### Persönliche Reflexion

Ich denke, ich könnte vieles verbessern, um meine Zeitpläne besser einzuhalten und am Ende nicht so viel Stress zu haben. Aber ich habe aus meinen Fehlern gelernt, und ich weiss, dass ich mich bei meiner nächsten Dokumentation deutlich verbessern werde.

### Technisches Review

Ich denke, am Ende des Tages ist alles gut gelaufen. Die Art und Weise, wie ich die Versionierung durchgeführt habe, hat zu einem ziemlich unübersichtlichen GitHub-Repository geführt, und ich könnte mich in diesem Bereich sehr verbessern.

**Auswertung zur Mehrere Personen:**

Das Erstellen von Antworten für mehrere Personen verlief fast reibungslos. Ich hatte insgesamt 2 Probleme, die recht einfach zu lösen waren. Das Verschieben des Feldes war sehr einfach, und das Problem mit mehreren E-Mails im CC-Feld wurde gelöst, indem ich das Feld geändert habe. Wenn ich jetzt darüber nachdenke, denke ich, dass es eine bessere Lösung geben könnte. Mindestens eine kleine Anleitung für die Benutzer, wie sie E-Mails an mehrere Personen senden können, indem sie Kommas verwenden, wäre eine gute Idee gewesen. Aber daraus habe ich gelernt und werde in Zukunft viel mehr darauf achten.

**Auswertung Einlesen und Matchen von einkommenden E-Mails:**

Das Matchen von E-Mails war schwieriger, besonders beim Matchen und beim Einrichten des gesamten Mailfetchers. Danach wurden die Probleme ziemlich schnell gelöst.

### Schlusswort

Ich bin glücklich, ein Teil der Verbesserung unseres täglichen Ticketsystems sein zu können. Auch wenn ich nicht alles so gut fertigstellen konnte, wie ich wollte, hatte ich am Ende Spass während des Projekts.