Abschlussprüfung Sommer 2024

Fachinformatiker für Anwendungsentwicklung

Dokumentation zur betrieblichen Projektarbeit

Digitalisierung von Erstellung, Verwaltung & Validierung der Reisekostenabrechnungen mithilfe einer Single-Page-Application (SPA).

17.04.2024

Ricardo Hoffmann

Elpke 19a

33605 Bielefeld

Ausbildungsbetrieb

DTS Systeme GmbH

Schrewestraße 2

32051 Herford

Inhaltsverzeichnis

# Einführung

In dieser Projektdokumentation wird der Ablauf des Abschlussprojekts, das durch den

Autor im Rahmen der Abschlussprüfung zum Fachinformatiker für Anwendungsentwicklung durchgeführt wurde, beschrieben.

Das Projekt wurde in der DTS Systeme GmbH durchgeführt. Die DTS Systeme GmbH ist der Ausbildungsbetrieb des Autors und wird im Abschnitt Projektumfeld näher beschrieben.

# Projektumfeld

Die DTS-Gruppe ist eine international tätige IT-Dienstleistungsunternehmensgruppe

mit über 400 Mitarbeitenden an 14 Standorten in Deutschland und Griechenland. Sie

betreut Kunden dabei in den Bereichen Datacenter, Technologies und Security und

bietet selbst entwickelte Software an.

Das Team Software Development entwickelt und betreut Software, wie das DTS

Monitoring, das DTS Cockpit, das DTS Identity Management oder DTS Identity as a

Service und betreibt die Hauseigne Shop Seite DTS Cloud Portal.

# Ausganssituation

Aktuell wird bei der DTS Systeme GmbH ein Komplexe, Fehleranfällige sowie Manipulationsanfällige Microsoft-Excel-Datei zur Erstellung von Reisekostenabrechnungen verwendet.   
Diese soll u.a. wegen den eben genannten Punkten ersetzt werden.

Ursprünglich wurde diese Microsoft-Excel-Datei von einem externen Dienstleister im Auftrag der DTS Systeme GmbH erstellt und seitdem nicht mehr von diesem Externen Anbieter weiterentwickelt oder angepasst, so werden einige ursprüngliche Funktionen und Felder nicht mehr benötigt.

# Projektbeschreibung

Das in Ausgangssituation beschriebene Tool soll nun abgelöst werden.

Dazu soll ein neues Tool entwickelt und langfristig eingeführt werden. Dafür soll zuerst im Rahmen dieses Projekts das Tool als solches implementiert werden. Im Nachgang soll es dann erweitert werden, um vollumfänglich einsetzbar zu werden.

Detaillierte Anforderungen an die Neuentwicklung sind im Lastenheft unter den Funktionalen und Nichtfunktionalen zu finden.

# Projektziel

Das Ziel soll sein, über ein neues, selbst entwickeltes Tool zu verfügen, das auf den

Servern der DTS-Gruppe gehostet werden kann und für Mitarbeitende über einen

Webbrowser erreichbar ist. Dies soll den allgemeinen Unmut über die aktuelle Lösung, sowie die zusätzliche Arbeitsbelastung für die Erstellenden, Prüfenden und Freigebenden zu reduzieren und zu vereinfachen.

# Projektschnittstellen

Wie in dem Lastenheft erwähnt, wird das System als solches in der ersten

Version, der dieses Projekt entspricht, über keine Schnittstellen zu anderen Tools

Verfügen bis auf folgenden vorgesehenen Schnittstellen:

* HTTP-API zur Kommunikation zwischen Front-und Backend
* Datenbankverbindung zur Speicherung von Daten
* Anbindung an den Hauseigenen IDP zum Login, Benutzer und Rechteverwaltung

# Projektabgrenzung

Da die zur Verfügung stehende Zeit begrenzt ist, muss der Projektumfang begrenzt

werden. Wie in Projektbeschreibung beschrieben, sind alle Schnittstellen zu anderen Systemen außerhalb des Scopes für diese Version.

Außerdem sind Stornierungsprozesse und ein Großteil der Managementfunktionalitäten

außerhalb dieses Scopes.

# Projektphasen und Zeitplanung

Wie von der IHK Ostwestfalen zu Bielefeld vorgeschrieben, stehen zur Umsetzung

dieses Projekts 80 Stunden zur Verfügung. Eine Übersicht zu der initialen Einschätzung der Arbeitspaket und der Zeit, die in der real benötigt wurde, folgt:

|  |  |  |
| --- | --- | --- |
| **Bezeichnung** | **Eingeplant** | **Tatsächlich** |
| **Projektdefinitionsphase**   * Durchführung Ist-Analyse * Durchführung Amortisationsrechnung * Erstellung eines Lastenhefts (Unterstützung des Fachbereichs) | **4**  1  1  2 | **3,5**  1  0,5  2 |
| **Projektplanungsphase**   * Erstellung eines Anwendungsfalldiagramm * Erstellung eines Aktivitätsdiagramm * Erstellung eines Pflichtenheftes * Datenbank Modellierung * Erstellung von Sketches * Einteilung von Arbeitspaketen (Kanban Board) | **10**  1  1  2  3  2  1 | **10,5**  0,5  1  2  4  2  1 |
| **Projektdurchführungsphase**   * Container Umgebung erstellen * Grundstruktur der Webanwendung Implementieren * IDP-Integration * Chatfragen für den Regelbasierenden Chatbot formulieren & strukturieren * Implementierung der Backend-Logik   + Berechnungslogik (basierend auf Pauschalen)   + Create, Read, Update, Delete (CRUD) Funktionen für die Reisekostenabrechnung   + Application Programming Interface (API) für das Frontend zur Datenbereitstellung * Implementierung des Frontends   + Mobile-First Oberfläche für CRUD-Operation der Reisekostenabrechnung   + Tabellenansicht zur Überprüfung der Reisekostenabrechnung * Qualitätssicherung und Tests (der Kritischen Berechnungsfunktionen) | **50**  1  5  1  4  7  6  4  8  8  6 | **52**  0,5  3  1  3  4  8  2  10  4  3 |
| **Projektabschlussphase**   * Abnahme * Interne Dokumentation * Projektdokumentation | **16**  2  3  11 | **14**  1  2  10 |

# Ressourcenplanung

Nach dem Aufstellen der Zeitplanung wurde eine Übersicht über die Ressourcen, die

voraussichtlich während der Projektdauer verwendet werden würden, angefertigt. Sie

wurde während des Projekts laufend aktuell gehalten, so dass auch spontan

hinzukommende Ressourcen abgebildet sind.

Die Ressourcenplanung ist im Anhang unter Verwendete Ressourcen zu finden.

Bei der Auswahl der Software wurde darauf geachtet, dass bereits Wissen oder

Erfahrungen zur jeweiligen Software vorhanden ist, um einen potenziellen Zeitverlust

durch Einarbeiten möglichst gering zu halten. Außerdem wurden die jeweiligen Lizenz- und Nutzungsbedingungen der verschiedenen Lösungen beachtet.

# Ist Analyse

Wie bereits unter Ausgangssituation beschrieben, wird aktuell ein veraltete und Unübersichtliche/Fehleranfällige Microsoft-Excel Datei zur erstellung von Reisekostenabrechnungn verwendet. Ein Bild dieser Microsoft Excel Datei finden sie im anhang. Der Prozess einer einreichung einer Reisekostenabrechnung ist im Anhang unter Prozess alt zu finden.

# Herausforderungen

Wie in der Ausgangssituation beschrieben wurde, ist diese Microsoft Excel Datei Komplex was zu den Fehlern führt so wie z.B. das Bedingungen falsch verstanden worden sind und dementsprechend eigentragen werden, obwohl diese nicht eingetragen werden dürften. Grundsätzlich können wegen der Art der Microsoft Excel Datei Manipulationen an berechnugslogik oder Pauschelen stattfinden. Außerdem kommt es oft vor das nötige Belege nicht mit eingereicht werden oder so dass die angegeben kosten nicht validiert werden können. Diese Herausforderung erschweren die Nutzung in den Abteilungen in welchen diese verwendet wird.

# Wirtschaftlichkeit

## Amortisationsrechnung

Von folgenden Stunden kosten der Mitarbeiter gehen wir aus:  
Accounting/Payroll Accounting: 30€  
MA im Außendienst: 25€  
GF: 100€  
Betreuerin/Ausbilderin: 30€  
Author/Programmierer/Azubi: 7€  
Vorgesetzter: 45€

Annahmen zu dem bestehenden Prozess der Erstellung einer Reisekostenabrechnung.

Durchschnittlich werden monatlich 20 Reisekostenabrechnung eingereicht.  
Die Erstellung einer Reisekostenabrechnung dauert je nach Umfang 10-20 min da im Durchschnitt eher umfangreichere Reisekostenabrechnungen erstellt werden gehen wir von einer durchschnittlichen Bearbeitungszeit einer Reisekostenabrechnung von 17 min aus. Durch die hohen Personal Einstellungen der DTS Systeme GmbH wird angenommen das jeden 2. Monat eine Einführung in die Bisherige Microsoft Excel Datei zur Erstellung/Bearbeitung einer Reisekostenabrechnung anfällt. Wir gehen davon aus das diese Einführung 1std dauert und insgesamt 2 MA im Außendienst erfordert. Auch wird angenommen das 25% durch die Unregelmäßigkeit der Erstellung einer Reisekostenabrechnung Ihre Kollegen um Hilfe bitten oder Fragen stellen, was ca. 15min an Zeit für die beiden MA in Anspruch nimmt. Es wird davon ausgegangen, dass die Genehmigung (und Unterschrift) der von dem jeweiligen Vorgesetzten (und ggf. GF falls die kosten >250€ sind was aber i.d.R nur einmal im Monat aufkommt jeweils) 5 Minuten dauern. Außerdem wird angenommen das 75% der Reisekostenabrechnungen über die Hauspost an die jeweiligen Vorgesetzten (ggf. GF) sowie der Accountig Abteilung gebracht werden und pro Reisekostenabrechnung mit 10 Minuten gerechnet wird. Das Ausliefern der Hauspost übernimmt i.d.R ein Auszubildender. Es wird davon ausgegangen, dass die Überprüfung, ob die richtigen Pauschalen genommen wurden und keine Manipulation stattgefunden durch das Payroll accountig 10 Minuten pro Reisekostenabrechnung dauert. Die Anschließenden Prüfungen der Accounting Abteilung dauert je nach Umfang 5-15 Minuten wir gehen hier von 12 Minuten im Durchschnitt aus. Es wird zudem davon ausgegangen das 25% der eingereichten Reisekostenabrechnungen nicht komplett korrekt ausgefüllt wurden. In solch einem Fall wird der Ersteller kurzfristig benachrichtigt mit den konkreten Anmerkungen, um diese kurzfristig umzusetzen was angenommen 20 Minuten dauert. Es wird davon ausgegangen das eine nachträglich Prüfung 7 Minuten dauert. Zusätzlich wird davon ausgegangen das bei 10% der Reisekostenabrechnungen fragen aufkommen und sich direkt an die Accountig Abteilung gewandt wird was angenommen erweise 10 Minuten dauert.

NEU

Angenommen wird das Erstellen einer Reisekostenabrechnung nun 20 Minuten dauert. Angenommen wird das Rückfragen an andere Mitarbeiter im Außendienst jetzt nur noch in 10% der Fälle auftaucht und nur 5 Minuten dauert. Die kosten für Genehmigungen bleiben die gleichen. Hauspost, Einarbeitung und die Prüfung des Payroll Accountings fallen weg. Die Prüfung der Accounting Abteilung reduziert sich nur auf angenommen 7 Minuten und eine Nachprüfung auf 5 Minuten. Angenommen werden nur noch 10% fehlerhafte Reisekostenabrechnungen. Reisekostenabrechnung Korrigieren dauert nun aber angenommen noch 7 Minuten. Rückfragen an die Accounting Abteilung bleiben aber die gleichen. Wartung von 2 std/Monat werden erwartet. Initiale Projekt Erstellungskosten 80 std. Autor kosten, 5std. Betreuerin aufgrund von diversen Hilfestellungen, 6std. Accounting Abteilung (á 2 Personen) zur Erstellung von Lastenheft, Abnahme und diversen rückfragen, 1 std. MA Außendienst zum testen und Feedback. Geschätzt wird ein Aufwand von 40 std. um die nötigen Features zu implementieren, um den Travel Assistent vollumfänglich im Betrieb einzusetzen.

# Anwendungsfälle und Benutzerklassifizierung

Um herauszufinden, wie das Berechtigungskonzept aussehen sollte, wurde analysiert, welche Personengruppen wie mit dem Tool arbeiten müssen.

Grundsätzlich lassen sich folgende drei Personengruppen feststellen:

• Reguläre Benutzer

Jede mitarbeitende Person, die berechtigt ist, Reisekostenabrechnungen einzureichen

• Vorgesetzte/GF

Personen, die berechtigt sind, Reisekostenabrechnungen anderer Personen zu bestätigen

• Prüfer

Personen, die zur Verwaltung von Benutzern und sonstigen Ressourcen berechtigt

sind

Außerdem ist im Anhang ein Anwendungsfalldiagramm zu finden welches die möglichen Aktionen der Jeweiligen Personengruppen zugeordnet wurden.

# Applikationsstart

Wie bereits unter Projektziel genannt, soll die Applikation als Webapplikation

entwickelt werden. Ein Grund dafür ist, dass man die Applikation so in der Zukunft

vergleichsweise einfach aktualisieren kann. Außerdem spricht die Tatsache dafür, dass man so von der Browserumgebung, mit zum Beispiel Passwortmanagern, profitiert.

Die Serverapplikation soll in JavaScript entwickelt werden. Es wurde sich für

JavaScript entschieden, da im Team Software Development grundsätzlich auf

JavaScript gesetzt wird. Die SPA soll mit React und JavaScript entwickelt werden. Die Wahl von React ergibt sich, da zahlreiche Applikationen der DTS-Gruppe bereits mit React umgesetzt worden sind oder aktuell umgesetzt werden. Dadurch kann gegebenenfalls vom Know-How anderer Personen im Team profitiert werden.

# Applikationsarchitektur

Die Applikation wird in zwei Teilapplikationen entwickelt werden: Frontend und

Backend. Diese Unterscheidung wird nötig, da sich bereits für die Implementierung einer SPA entschieden wurde. Eine SPA kann nicht sicher mit Daten interagieren, was eine Backendapplikation nötig macht.

# Frontend

Das Frontend wird dabei von einer SPA repräsentiert. Eine SPA ist eine Applikation,

die vollständig im Browser läuft. Im Gegensatz zu regulären Webanwendungen wird

nicht mit jeder Interaktion mit dem Server HTML zurückgegeben und vom Browser

dargestellt. Stattdessen wird die Interaktion durch browserseitig ausgeführtes

JavaScript durchgeführt.

# Backend

Das Backend ist die Teilapplikation, die auf dem Server läuft und die gesamte

Datenverwaltung sowie Businesslogik enthält. Das Backend speichert die Daten in einer

MongoDB und greift mit Mongoose darauf zu.

Es wird eine HTTP- API bereitgestellt, mit der das Frontend mit dem Backend

kommunizieren kann. Darüber können also Reisekostenabrechnungen eingereicht, bearbeitet,

genehmigt und geprüft werden.

Das Backend wird in der MSC-Architektur entwickelt. Dabei werden die Komponenten

nach Typ, geteilt. Bei der MSC-Architektur gibt es folgende Komponentenarten:

## 1.Model

Das Model beschreibt die zu speichernden Daten im Rahmen einer einzelnen

Entität. Außerdem werden im Model die klassischen CRUD-Operationen

durchgeführt.

## 2. Service

Im Service findet die gesamte Logik statt. Services sind die einzigen Komponenten

der Applikation, die mit Models interagieren. Sie bieten die Möglichkeit, die

CRUD-, und gegebenenfalls noch weitere, Operationen auszuführen.

## 3. Controller

Controller sind die Komponenten, die die tatsächliche Interaktion mit der

Außenwelt, also dem HTTP-Client bestreiten. Sie interagieren mit den Services,

um die Anfragen des Clients auszuführen.

Neben diesen Hauptkomponenten gibt es noch folgende Komponenten:

## 1. Middleware

Neben den mitgelieferten oder nachinstallierten Middlewares können auch eigene

Middlewares erstellt werden. Im konkreten Fall wird eine Middleware genutzt, die

den Zugang zu bestimmten Bereichen der Applikation beschränkt. Dadurch wird

eine Anfrage, bevor sie überhaupt im Controller ankommt, bereits beantwortet,

falls bestimmte Anforderungen nicht erfüllt werden.

## 2. Utilities

Utilities sind Hilfsfunktionen, die in keinen der Bereiche wirklich passen, die aber

auch unabhängig von diesen verwendet werden können.

# Models und Datenstruktur

Durch die Anforderungen ist klar, dass mehrere Models benötigt werden.

So wird ein Model für Länder und ihre Pauschalen benötigt. Die Reisekostenabrechnung an sich welche sich aus dem Chat verlauf ergibt sowie grundlegende Informationen und Dienstreisen welche Informationen aus dem Chat verwertet. Sowie ein Model welches einige grundlegende Berechnung speichert und Benutzer um die jeweiligen Vorgesetzten zu speichern.

# Feststellung der Benutzerberechtigungen

Das in Benutzerklassifizierung beschriebene Benutzerkonzept sieht drei

Benutzerklassen vor. Dabei wird davon ausgegangen, dass jeder Nutzer, der sich an der Applikation bzw. den IDP einloggt, initial ein regulärer Benutzer ist. Im Anschluss wird dann geprüft, ob ein Nutzer über eine Rolle verfügt, die über den IDP dem Nutzer zugeordnet wird.

# Durchführungsphase

## Auswahl von Libraries

Bei der Auswahl von Libraries wurde einigen Aspekten eine besondere Beachtung

geschenkt:

• Lizenz

Selbstverständlich muss die Lizenz der Library die Nutzung im konkreten Umfeld

erlauben.

• Beliebtheit

Für die Beliebtheit wurde sich an den Metriken der GitHub-Stars und

wöchentlichen Downloads bedient. Je beliebter eine Library ist, desto höher ist

die Wahrscheinlichkeit, dass sie langfristig weiter gepflegt wird.

• Dependencies

Neben der Lizenz und der Beliebtheit ist auch die Anzahl und Art der

Dependencies der Library ein wichtiger Punkt. Eine Library, die weniger weitere

Dependencies hat, kann besser eingesetzt werden, da man so besser kontrollieren

kann, welcher Code in das Projekt gelangt.

## Umgang mit Git/Source Code Management und Aufgabenaufteilung/management

### git

Im Unternehmen wird Git zur Versionsverwaltung genutzt.

Git ist ein Versionskontrollsystem, das ursprünglich von Linus Torvalds6 zur

Entwicklung des Linux-Kernels entwickelt worden ist. Git wurde 2005 veröffentlicht

und ist heute de-facto Industriestandard

### Git Flow

Beim Workflow wird sich an Git-Flow7 orientiert.

Git-Flow ist ein Branching-Modell, das 2010 von Vincent Driessen ver¨offentlicht

worden ist.

In Git-Flow gibt es zwei Haupt-Branches: master und develop. Der develop-Branch

enthält zu jeder Zeit eine lauffähige Version der Applikation, der master-Branch

enth¨alt die aktuell produktive Version.

Neben diesen beiden Haupt-Branches gibt es sogenannte Feature-Branches. In ihnen

werden neue Features hinzugefügt und weiterentwickelt. In der Regel werden die

Branch-Bezeichnungen mit feature/-geprefixt. Neue Features werden aus dem

develop-Branch erstellt und auch wieder zurück in diesen gemerget.

Außerdem gibt es die Release-Branches. In ihnen werden letzte Vorbereitungen f¨ur ein

bevorstehendes Release getroffen, wie zum Beispiel die Inkrementierung der

Versionsnummer. In der Regel werden die Branch-Bezeichnungen mit release/

geprefixt. Neue Releases werden aus dem develop-Branch erstellt und sowohl in den

master-Branch als auch in den develop-Branch zurück gemerget.

Sollte es in der Produktivumgebung zu Fehlern kommen, werden diese in Hotfixes behoben. Dazu werden Hotfix-Branches genutzt, die mit hotfix/ geprefixt werden und

aus dem master-Branch erstellt werden. Fehlerbehebungen werden dann, genau wie die

Releasevorbereitungen, in sowohl master als auch in develop gemerget. Diese

Mechanik wird angewandt, um sicherzustellen, dass niemals Features, die noch nicht

released waren, im Rahmen eines Hotfixes released werden.

Aktuellen Empfehlungen8 nach ist der master-Branch aus politischen Gründen nun als

main zu bezeichnen.

### Aufgabenaufteilung/Aufgabenmanagement

Als das Vorgehensmodell hat der Autor sich für das Kanbanboard entschieden, da dieses Hauptsächlich in dem Software-Entwicklungsteam verwendet wird. Die Anforderungen wurden in Aufgaben unterteilt und grob priorisiert, hierzu wurde das Online Web Tool Miro verwendet und die Aufgaben und deren Status zu visualisieren.

Ein Auszug, der des Kanbanboards ist im Anhang zu finden.

## Erstellung des Applikationsgrundgerüsts

Zuerst wurde das Grundgerüst der Applikation vorbereitet. Dazu wurde eine

Git-Repository erstellt. In der Repository wurden zwei Hauptverzeichnisse erstellt:

Backend und frontend. Im backend-Verzeichnis befindet sich das

JavaScript-Projekt der Backendapplikation, im frontend-Verzeichnis das mit vite

erstellte React- JavaScript-Projekt. Außerdem befinden sich im

Hauptverzeichnis die .gitignore-Datei zum Ausschließen von Dateien aus der

Git-Repository hierfür wurde eine Template für gängige Datei(Endungen), eine .env-Datei mit dazugehöriger .env.sample-Datei(im Anhang zu finden) zum Bereitstellen von Umgebungsvariablen, wie z.B. client Id und secret um sich bei dem IDP zu authentifizieren. Und die für Docker nötige Dateien docker-compose.yml und Dockerfile.

# Vorbereitung der Backendapplikation

Nachdem das Grundgerüst der Applikation aufgebaut war, wurde sich der

Implementierung der Backendapplikation gewidmet.

Dabei wurde zuerst die Berechnungslogik der Pauschalen

Dabei wurde zuerst mit einer simplen http API begonnen, die dann

im Verlaufe der Implementierung der SPA immer der konkreten Anforderung

entsprechend angepasst wurde.

Bei der Implementation einer so umfangreichen Applikation werden nahezu immer externe Libraries eingesetzt. Das ist auch in diesem Projekt passiert. Diese ¨Übersicht

enthält die wichtigsten genutzten Libraries:

Projektziel

Die bisher verwendete Microsoft-Excel-Datei zur Erstellung von Reisekostenabrechnungen, die von den Außendienstmitarbeitern schriftlich/digital ausgefüllt wird, soll abgelöst werden. Grund dafür ist, dass diese komplexe Microsoft-Excel-Datei nicht benutzerfreundlich gestaltet ist, was zu Fehlern bei der Bearbeitung führt. Außerdem ist der Prozess durch den notwendigen Ausdruck dieser Microsoft-Excel-Datei und den anschließenden Freigabeprozess sehr zeitaufwendig.

Im Rahmen der Digitalisierung/Automatisierung von manuellen sowie fehlerbehafteten Prozessen der DTS Systeme GmbH, soll eine Mobile-First SPA mit Datenbankanbindung entwickelt werden. Diese soll die Außendienstmitarbeiter bei der Erstellung von Reisekostenabrechnungen sowie bei der Speicherung der Belege unterstützen. Gleichzeitig soll dabei die Prüfung der Reisekostenanträge für die Accounting-Abteilung durch die Digitalisierung erleichtert werden.

Dazu soll die SPA dem Benutzer vordefinierte Fragen stellen (Regelbasierender Chatbot), notwendige Berechnungen durchführen und die Möglichkeit bieten, Belege hochzuladen. Eine eigene Ansicht zur Prüfung von Reisekostenabrechnungen, in welcher die Berechnungen transparent aufgeschlüsselt werden. Login & Benutzer-/Rechteverwaltung soll über die Integration des hauseigenen Identity Provider (IDP) verwaltet werden.

Projektumfeld

DTS Systeme GmbH ist ein internationaler Provider von IT-Lösungen & Services sowie Securitysoftware-Hersteller. Mit über 400 Mitarbeitenden an 14 Standorten bieten wir unser Know-how in den Bereichen Datacenter, Technologies und Security an. DTS hat ein breit gefächertes Portfolio und um dies abdecken zu können gibt es viele Abteilungen welche sich auf Komponenten des Portfolios Spezialisieren. Unter anderem Techniker und Vertriebler, welche Dienstreisen tätigen. Diese Dienstreisen sind oftmals mit kosten gebunden, welche durch Pauschalen mithilfe einer Reisekostenabrechnung im Nachhinein gedeckt werden sollen. Dafür hat die Accounting Abteilung eine Excel-Datei zur Verfügung gestellt, welche sehr komplex aufgebaut, schwer zu verstehen, fehleranfällig sowie manipulationsanfällig ist. Durch diese Umstände muss die Eingereichte Reisekostenabrechnung sorgfältig geprüft werden. Diese Prüfung fängt bereits bei dem Vorgesetzten an, welcher als erstes die Reisekostenabrechnung erhält und diese grundlegend validiert, ob diese Dienstreise und dessen umfang genehmigt war und muss dann im Anschluss unterschrieben werden. Bis die Reisekostenabrechnung bei dem Vorgesetzten vorliegt kann es 0,5-3 Tage dauern da diese Per Post/Hauspost versendet wird. Oftmals gehen solche Dokumente auch unter bzw. es wird vergessen zu validieren. Nachdem die Reisekostenabrechnung validiert wurde, muss diese dann der Payroll Accounting Abteilung zu übergeben, welche die eingetragenen werte prüft, ob diese nicht in irgendeiner Weise manipuliert wurden und die richtigen Pauschelen verwendet worden sind. Das dauert dann je nach Größe 3-5 Minuten und kann an dieser Stelle auch vergessen worden sein bzw. untergegangen sein. Nach dem das geprüft wurde geht es weiter zur Accounting Abteilung, welche die errechneten Werte überprüft und mit den miteingereicht belegen vergleicht. Sofern was unpassend ist oder belege nachgereicht werden müssen wird sich immer wieder mit dem Reisekosten Einreichenden kommuniziert.

Die Softwareentwicklung ist in 4 Sparten/Teams unterteilt, welche sich in den Programmiersprachen aufteilen lassen: Python, JS, C# und das Tochterunternehmen ISL welches eine Ausnahme bildet.

In dem „JS-Team“ in welchem ich arbeite, werden hauptsächlich eigene Webapplikationen programmiert. Das Projekt wird mit dem Technologie-Stack MERN (MongoDB, Express.js, React.js, Node.js) realisiert da wir diesen Technologie Stack in dem JS-Team hauptsächlich nutzen.