Entwicklung einer Web-Anwendung zur Projektantragsverwaltung mit JSON Datenbank

Bericht zur betrieblichen Projektarbeit im Zeitraum:

06.04.2016 – 24.04.2016

von

**Timo Menhorn**

zur Erlangung des Abschlusses als

**Fachinformatiker Anwendungsentwicklung**

Ausbildungsberuf: Fachinformatiker Anwendungsentwicklung

Ausbildungsbetrieb: Donat IT GmbH

Verfasser: Timo Menhorn

Prüfung: Sommer 2020

Prüflingsnummer: 23056

Projektbetreuer: Jan Brinkmann

**Inhaltsverzeichnis**

[1 Einleitung 2](#_Toc35408463)

[1.1 Vorstellung des Ausbildungsbetriebs 3](#_Toc35408464)

[1.2 Projektumfeld 4](#_Toc35408465)

[2 Projektbeschreibung 4](#_Toc35408466)

[2.1 Ausgangssituation 5](#_Toc35408467)

[2.2 Projektziele 5](#_Toc35408468)

[2.2.1 Sachziel 5](#_Toc35408469)

[2.2.2 Zeitliches Ziel 6](#_Toc35408470)

[3 Projektplanung 7](#_Toc35408471)

[3.1 IST-Analyse 7](#_Toc35408472)

[3.2 SOLL-Konzept 7](#_Toc35408473)

[4 Pflichtenheft 8](#_Toc35408474)

[4.1 Zielbestimmung 8](#_Toc35408475)

[4.1.1 Muss-Kriterien 8](#_Toc35408476)

[4.1.2 Soll-Kriterien 8](#_Toc35408477)

[5 Projektdurchführung 9](#_Toc35408478)

[5.1 Angular Vs AngularJS 9](#_Toc35408479)

[5.2 Alternativen 10](#_Toc35408480)

[5.2.1 React 10](#_Toc35408481)

[5.2.2 VueJS 10](#_Toc35408482)

[6 Projektabschluss 11](#_Toc35408483)

[6.1 Projektübergabe 11](#_Toc35408484)

[6.2 Glossar 11](#_Toc35408485)

[6.3 Quellenangabe 13](#_Toc35408486)

[7 Selbsterklärung 14](#_Toc35408487)

[8 Anlagen 15](#_Toc35408488)

# Einleitung

Zu Beginn der Projektarbeit möchte ich meinen Ausbildungsbetrieb vorstellen und kurz auf das Projektumfeld eingehen. Unterstrichene Worte sind im Glossar am Ende der Projekt-arbeit erläutert.

## Vorstellung des Ausbildungsbetriebs

Die Ursprünge des Unternehmens gehen bis in das Jahr 1980 zurück, als Herr Donat sen. zeitgleich mit der Gründung des Media Marktes in Ingolstadt die Donat GmbH als Dienstleistungsunternehmen im Bereich Unterhaltungselektronik gründete. Auch wenn zu diesem Zeitpunkt das Thema IT noch nicht relevant war, zeigt es doch die lange Verbundenheit des Unternehmens mit der Region und der Erbringung von Dienstleistungen.

Mit dem Einstieg in das Unternehmen seines Vaters und der Erweiterung um den Geschäftsbereich, IT legte Hans Donat jun. den Grundstein für die Unternehmen der heutigen Donat group, die als Holding zentrale Aufgaben wahrnimmt, zum Beispiel in den Bereichen Finanzen, Kommunikation und übergeordneten Vertrieb.

Die Donat IT GmbH, als ein Unternehmen der Donat group GmbH, ist auf die unter-schiedlichsten Anforderungen von Privatanwendern sowie Unternehmen im Bereich der elektronischen Datenverarbeitung spezialisiert. Die Leistungen umfassen neben den vollständigen Installationen von Microsoft und Apple Betriebssystemen auch die Konfiguration und Zusammensetzung kompletter Systeme. Aufgrund einer Vertriebspartnerschaft mit Wortmann und HP ist es der Donat IT GmbH möglich, ihre Produkte zu attraktiven Preisen an den Endverbraucher weiterzugeben. Neben dem Hauptfirmensitz in Ingolstadt ist das Unternehmen in Form von Tochtergesellschaften auch in Neckarsulm und im Raum München vertreten. Aktuell beschäftigt die Donat IT GmbH 86 Mitarbeiter (Stand April 2016).

|  |  |
| --- | --- |
| Jahr | Ereignis |
| 1980 | Gründung der Donat GmbH durch H. Donat sen. als Dienstleistungspartner für Produkte der Unterhaltungselektronik |
| 1998 | Eintritt von H. Donat jun. in die Geschäftsführung und Erweiterung des Unternehmens um den Geschäftsbereich IT-Services |
| 2001 | Beginn des Projektgeschäfts im Automotive Bereich (1st & 2nd Level Support) |
| 2005 | * Gründung der Donat IT GmbH * Ausgliederung des Geschäftsfeldes IT Services in die Firma Donat IT GmbH * Gründung der Donat SysTec GmbH |
| 2008 | Ausweitung des Geschäftsbereichs um den Support für produktionsrelevante Hostsysteme |
| 2009 | Ausbau der verschiedenen Geschäftsbereiche und Umstrukturierung der Donat group |
| 2010 | Aufbau des Geschäftsbereichs „Support & Entwicklung für Microsoft SharePoint“ |
| 2012 | Ausbau der Geschäftsbereiche „Project Services“ und „Projektmarketing & Releasekommunikation“ |
| 2013 | Anpassung der Organisationstruktur und Einführung eines eigenen Versorgungswerkes unter dem Namen “Donat group Unterstützungskasse” |
| 2015 | Wechsel der Geschäftsführung, neue Geschäftsführerin: Ediba Hastor |

## Projektumfeld

Die Webapplikation basiert auf einem Angular 9 Frontend. Die Daten, die durch die Usereingaben und Aktionen generiert werden, werden in einer JSON Datenbank abgespeichert, welche zugleich als REST API dient.

Die Komponenten sollen sich größtenteils dynamisch generieren, um eine einfache Wartbarkeit bei sowohl kleinen als auch größeren Änderungen zu garantieren. Die Daten hierfür kommen jeweils aus einer Export Klasse im JSON Format. Die Validierung der Nutzereingaben kommt ebenfalls aus dieser Klasse.

# Projektbeschreibung

Im Zuge der Umstrukturierung des Unternehmens, soll die Handhabung der Projektanträge überarbeitet werden. Hierfür soll eine Webanwendung mit Backend entwickelt werden um die Antragstellung und -verwaltung zu vereinheitlichen. Die Webanwendung soll einen funktionierenden Log In und ein Usermanagement besitzen.

Ein Antragsteller soll die Möglichkeit haben, Anträge anzulegen und den Aktuellen Status seiner gestellten Anträge in einer Übersicht verfolgen zu können.

## Ausgangssituation

Derzeit werden Projektanträge, durch die Bereichsleiter, mithilfe von Excel Dateien erstellt und abgespeichert. Je nach Größe und Umfang des Projektes bedarf es noch weiterer Genehmigungen, die durch Versenden des Excel Dokuments eingeholt werden.

Im Moment wird das firmenweit eingesetzte Tool Excel als Grundlage zur Erstellung der Anträge genutzt. Jedoch sind innerhalb dieser Excelvorlage kaum Einschränkungen durch definierte Felder (nahezu durchgängig erstellte Freitextfelder) gegeben, was zu einer hohen Fehleranfälligkeit führt.

Wie bereits zuvor erwähnt, durchlaufen gewisse Anträge einen Genehmigungsprozess, was jedoch kaum zu überblicken ist. Daher ist eine Statusschaltung gewünscht.

## Projektziele

Seitens des Auftraggebers wurden folgende Projektziele vorgegeben, die zu erfüllen sind:

### Sachziel

Es ist gewünscht, das als Prototyp eine Webapplikation, basierend auf einem Angular 9 Frontend mit Fake JSON Backend, welches zugleich als REST API fungiert, entwickelt wird.

Es soll die Möglichkeit geben, neue Anträge zu stellen und den Status von bestehenden Anträgen einzusehen.

Für die Autorisierung soll es eine Userverwaltung geben, auf der ein Admin bestehende User ändern kann.

User der Rolle „Admin“ sollen den Status von Anträgen weiterschalten können.

Mögliche Status werden basierend auf dem aktuellen Status aus dem Backend geladen.

### Zeitliches Ziel

Im Folgenden wird ein kurzer Überblick über den zeitlichen Projektaufwand gegeben:

|  |  |
| --- | --- |
| Tätigkeit | Zeitaufwand |
| Ermittlung IST-Zustand | 1 Stunden |
| Ermittlung SOLL-Zustand | 1 Stunden |
| Analyse des Genehmigungsprozess | 2 Stunden |
| Ausarbeitung des Genehmigungsprozess | 2 Stunden |
| Erstellen eines Dummy Prototypen | 8 Stunden |
| Login mit Passwort Hashing | 6 Stunden |
| UI zum Anlegen von Anträgen | 10 Stunden |
| UI zum Anzeigen und Filtern von Anträgen | 5 Stunden |
| UI für Usermanagement | 10 Stunden |
| Erstellung der Statusschaltung | 2 Stunden |
| Projektbesprechungen und Wissenstransfer | 1 Stunden |
| Erstellung der Projektdokumentation | 22 Stunden |
| * Gesamt | 70 Stunden |

# Projektplanung

## IST-Analyse

Derzeit werden Projektanträge, durch die Bereichsleiter, mithilfe von Excel Dateien erstellt und abgespeichert. Je nach Größe und Umfang des Projektes bedarf es noch weiterer Genehmigungen, die durch Versenden des Excel Dokuments eingeholt werden.

Im Moment wird das firmenweit eingesetzte Tool Excel als Grundlage zur Erstellung der Anträge genutzt. Jedoch sind innerhalb dieser Excelvorlage kaum Einschränkungen durch definierte Felder (nahezu durchgängig erstellte Freitextfelder) gegeben, was zu einer hohen Fehleranfälligkeit führt.

Wie bereits zuvor erwähnt, durchlaufen gewisse Anträge einen Genehmigungsprozess, was jedoch kaum zu überblicken ist. Daher ist eine Statusschaltung gewünscht. Diese soll anzeigen, ob der Antrag bereits genehmigt oder abgelehnt ist.

## SOLL-Konzept

Die Webapplikation basiert auf einem Angular 9 Frontend. Die Daten, die durch die Usereingaben und Aktionen generiert werden, werden in einer JSON Datenbank abgespeichert, welche zugleich als REST API dient.

Die Komponenten sollen sich größtenteils dynamisch generieren, um eine einfache Wartbarkeit bei sowohl kleinen als auch größeren Änderungen zu garantieren. Die Daten hierfür kommen jeweils aus einer Export Klasse im JSON Format. Die Validierung der Nutzereingaben kommt ebenfalls aus dieser Klasse.

Die Login Komponente soll überprüfen, ob der Nutzer bereits angelegt ist und im erfolgsfall, das Passwort überprüfen. Wenn der Nutzer sich neu registriert, soll das Passwort gehasht übertragen und in der Datenbank abgelegt werden.

Auf der Home Komponente soll der Antragsteller die Auswahl haben, ob er einen Antrag anlegen oder eine Übersicht der gestellten Anträge haben will.

Wenn er einen Antrag anlegen will, bekommt er eine Liste mit möglichen Projektthemen und landet anschließend in der Komponente zum Anlegen.

# Pflichtenheft

Beruhend auf den Ergebnissen der SOLL-Analyse wird nun im Pflichtenheft festgelegt, wie die einzelnen Anforderungen umgesetzt werden.

## Zielbestimmung

### Muss-Kriterien

**Übersichtlichkeit**

Die Anwendung muss übersichtlich und leicht zu verstehen sein, damit jeder Nutzer sofort versteht, wie diese handzuhaben ist.

**Hashing**

Passwörter dürfen nie in Klartext abgespeichert oder hinterlegt werden.

**Validierung der Eingaben**

Um den Nutzer vor falschen Eingaben zu schützen, muss jedes Feld validiert sein und wenn Nutzereingaben beschränkt sind, müssen Dropdowns verwendet werden. Die Statusschaltung muss sicherstellen, dass ein Antrag nur in den dafür vorgesehenen Status weitergeschalten werden kann.

### Soll-Kriterien

**Wartbarkeit**

Da es bei den Anforderungen zu den Projektanträgen öfters zu Änderungen kommt, muss die Applikation leicht zu warten sein.

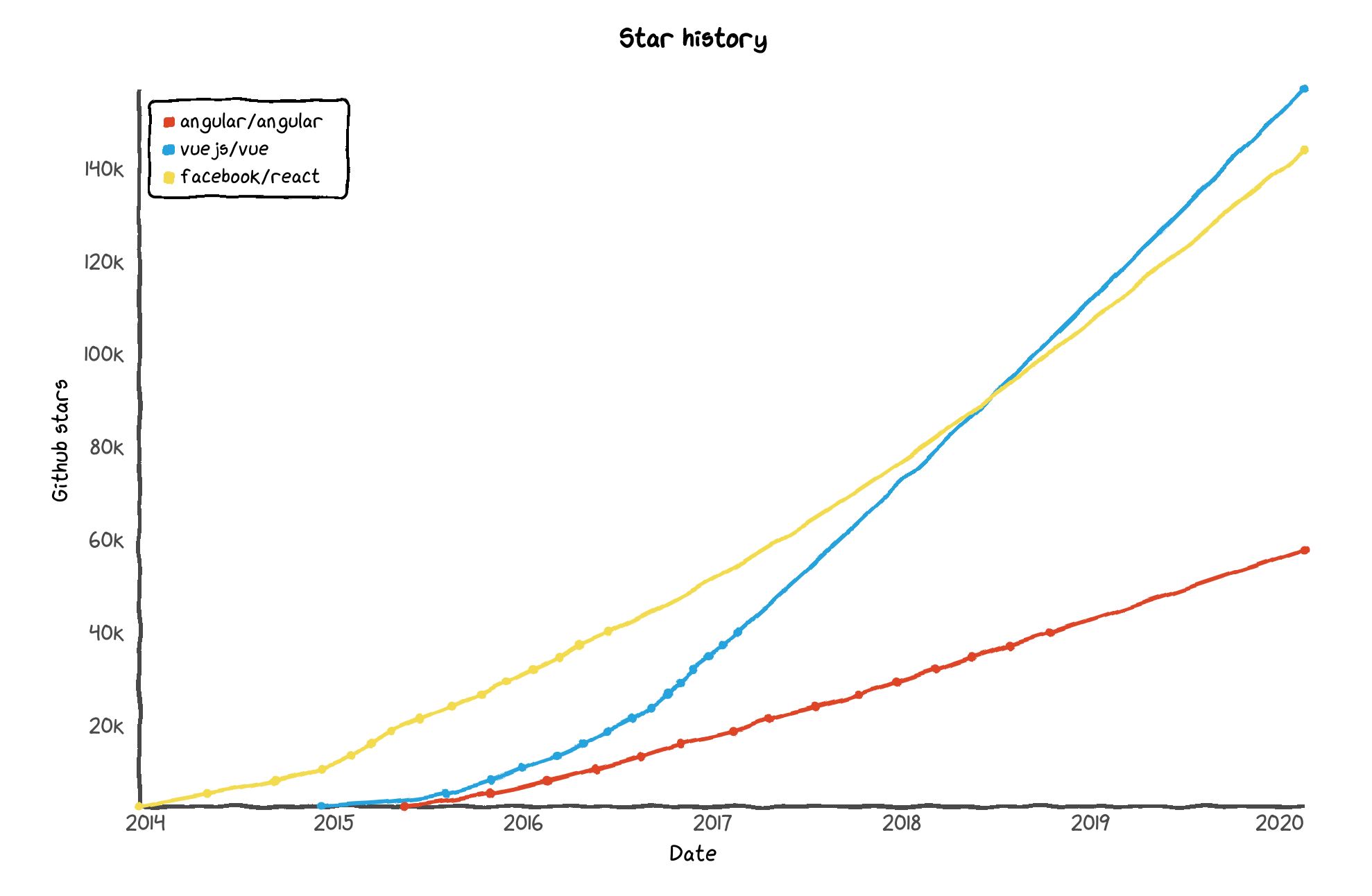
# Projektdurchführung

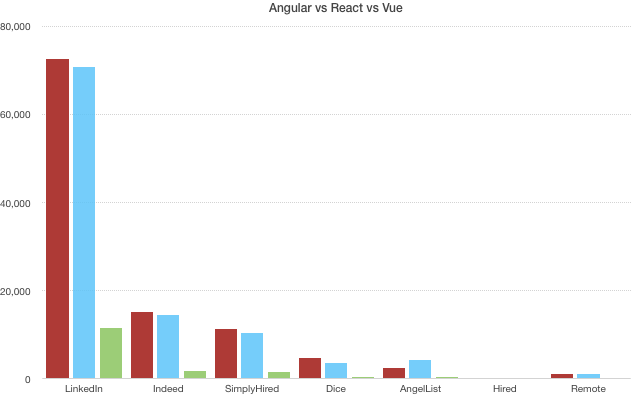
## **Angular Vs AngularJS**



* Angular kennt keine „scopes“ oder Controller, sondern verwendet eine Hierarchie von Komponenten als zentrales Architekturkonzept.
* Angular hat eine einfachere Syntax für Ausdrücke: Mit „[ ]“ werden Bindings für Eigenschaften und mit „( )“ werden Bindings für Events erzeugt.
* Mobile-First-Ansatz: Die Anforderungen von mobilen Plattformen haben besondere Priorität.
* Modularität: Module können Funktionalitäten auslagern, sodass Code leichtgewichtiger und schneller wird.
* Nur moderne Browser werden unterstützt, wodurch weniger Workarounds wegen Browser-Kompatibilitätsproblemen benötigt werden.
* Verbesserte Dependency Injection: Bindings ermöglichen es, dass Abhängigkeiten benannt werden können.
* Direktives dienen dem Zweck, das Aussehen und Verhalten eines Tags dynamisch zu verändern.
* Dynamisches Laden
* Asynchrone Kompilierung von Templates
* Einfacheres Routing
* Kontroller und $scope werden durch Komponenten und Direktiven ersetzt. Eine Komponente ist eine Direktive mit einem Template.

## Alternativen





### React

React ist eine JavaScript-Softwarebibliothek, die ein Grundgerüst für die Ausgabe von User-Interface-Komponenten von Webseiten zur Verfügung stellt (Webframework). Komponenten werden in React hierarchisch aufgebaut und können in dessen Syntax als selbst definierte HTML-Tags repräsentiert werden. Das Modell von React verspricht durch die Konzepte des unidirektionalen Datenflusses und des „Virtual DOM“ den einfachen, aber trotzdem performanten Aufbau auch komplexer Anwendungen. React bildet typischerweise die Basis für Single-Page-Webanwendungen, kann jedoch auch mittels Node.js serverseitig (vor-)gerendert werden.

### VueJS

Vue.js ist ein clientseitiges [JavaScript](https://de.wikipedia.org/wiki/JavaScript)-[Webframework](https://de.wikipedia.org/wiki/Webframework) zum Erstellen von [Single-Page-Webanwendungen](https://de.wikipedia.org/wiki/Single-Page-Webanwendung) nach dem [MVVM](https://de.wikipedia.org/wiki/Model_View_ViewModel)-Muster, es kann allerdings auch in Multipage Webseiten für einzelne Abschnitte verwendet werden. Ab Version 2.0 unterstützt es auch serverseitiges Rendern.

### Warum Angular?

Die Entscheidung für Angular war eher eine Entscheidung gegen React und VueJS.

VueJS ist ein Community getriebenes Projekt, dessen Dokumentation noch nicht vollständig auf Englisch übersetzt ist. Ansonsten bietet VueJS eine gute Alternative.

React ist eine Bibliothek bei der HTML mit JS gemischt wird, zudem unterstützt es nur One-Way Databinding.

Mir gefällt die Struktur von Angular, da es hier eine klare Trennung zwischen den Componenten und dem zugehörigen Styling gibt.

# Projektabschluss

## Projektübergabe

Die angesetzten 35 Stunden für die Projektarbeit sowie der Kostenrahmen konnten eingehalten werden. Das SOLL-Konzept wurde erfolgreich umgesetzt, ist jedoch noch unter Geschäftsbetrieb abschließend zu testen. Das Hotelpersonal wurde mit den neuen Arbeitsplätzen sowie dem Kassensystem vertraut gemacht. Sie haben zudem jederzeit die Möglichkeit, bei Fragen bzw. Problemen uns über unsere Service Hotline zu erreichen.

Die Reaktion des Auftraggebers fiel nach der Vor-Ort-Präsentation sehr positiv aus und somit steht die neue IT-Systemumgebung dem Hotel ab sofort zur Verfügung.

|  |  |  |
| --- | --- | --- |
| **Tätigkeit** | **Geplant** | **Benötigt** |
| Projektanalyse / Ist-Zustand | 2 Stunden | 1 Stunde |
| Projektplanung (inkl. Meeting) | 4,5 Stunden | 5 Stunden |
| Erstellung eines Pflichtenhefts | 2 Stunden | 3 Stunden |
| Vergleich von Herstellern | 2,5 Stunden | 2 Stunden |
| Einholen von Angeboten | 2 Stunden | 2 Stunden |
| Vergleichen der Angebote | 3 Stunden | 3 Stunden |
| Handelskalkulation | 2,5 Stunden | 3 Stunden |
| Beschaffung der Hardware | 1,5 Stunden | 1 Stunde |
| Total Cost of Ownership | 4 Stunden | 5 Stunden |
| Kosten- und Nutzenanalyse | 4 Stunden | 3 Stunden |
| Erstellung der Dokumentation | 5 Stunden | 5 Stunden |
| Übergabe und Projektabschluss | 2 Stunden | 2 Stunden |
| * **Gesamt** | 35 Stunden | 35 Stunden |

## Glossar

**Active Directory**

Active Directory ist der Verzeichnisdienst von Microsoft. Es ermöglicht, ein Netzwerk entsprechend der realen Struktur des Unternehmens oder seiner räumlichen Verteilung zu gliedern. Dazu verwaltet es verschiedene Objekte in einem Netzwerk wie beispielsweise Benutzer, Gruppen, Dienste, Server, Dateifreigaben und andere Geräte wie Drucker und Scanner sowie deren Eigenschaften. Mit Hilfe von Active Directory kann ein Administrator die Informationen der Objekte organisieren, bereitstellen und überwachen.

**Domäne/Domain**

Im Internet oder anderen Computernetzen ist eine Domäne eine Gruppe zusammen-gehörender Computer, die über eine spezielle Adresse erreichbar sind. Die Adressen-struktur einer solchen Gruppe erkennt man beispielsweise anhand der baumstrukturierten Namenshierarchie im Internet, bei denen die Domäne einen zusammenhängenden Teil-bereich des Domain Name System (DNS) darstellt.

**Hot-Spare-Festplatte**

Eine Hot-Spare-Festplatte ist eine Reservefestplatte in einem System. Fällt eine andere Platte aus, wird die Hot-Spare-Platte im laufenden Betrieb automatisch anstelle der defekten eingebunden. Die Festplatte ist im fehlerfreien Betrieb abgeschaltet und wird erst bei Bedarf per Software angeschaltet.

**Microsoft Exchange Server**

Microsoft Exchange Server ist eine E-Mail-Transport-Server-Software des Unternehmens Microsoft. Sie dient der zentralen Ablage und Verwaltung von E-Mails, Terminen, Kontakten, Aufgaben und weiteren Elementen für mehrere Benutzer und ermöglicht so die Zusammenarbeit in einer Arbeitsgruppe oder in einem Unternehmen. Um als Anwender die Funktionen von Exchange Server nutzen zu können, ist eine zusätzliche Client-Software (Microsoft Outlook) nötig.

**RAID (R**edundant **A**rray of **I**ndependent **D**isk)

RAID-Systeme basieren auf Speicherkonzepte, um Daten redundant zu speichern. In der Regel erreicht man dadurch eine Steigerung der Geschwindigkeit der Schreib- und Lesezugriffe (RAID-Level 0) oder die Verbesserung der Datensicherheit (RAID-Level 1). Das eigentliche Ziel beim Einsatz eines RAIDs, ist die Verfügbarkeit der Daten zu verbessern. In der Regel geht es darum, sich vor dem physikalischen Ausfall einer Festplatte zu schützen. Das erreicht man, in dem man die Daten redundant auf mehreren Festplatten speichert, die zu einem logischen Laufwerk zusammengefasst sind.

**Total Cost of Ownership (TCO)**

Total Cost of Ownership bezeichnet ein Verfahren zur Ermittlung von Investitionskosten für eine Anschaffung.

Bei der Kostenberechnung werden nicht nur die reinen Anschaffungskosten zugrunde gelegt, sondern auch alle zukünftigen Folgekosten, wie bspw. Energieverbrauch, Verbrauchs-materialen, Wartung, Reparaturen sowie ggf. die späteren Entsorgungskosten.

## Quellenangabe

**Internet:**

* [www.hp.de](http://www.hp.de)
* [www.dell.de](http://www.dell.de)
* <http://www.elektronik-kompendium.de/>
* [www.wikipedia.de](http://www.wikipedia.de)
* [www.itwissen.info](http://www.itwissen.info)
* [www.heise.de](http://www.heise.de)

**Buch und interne Quelle:**

* Heinrich Hübscher/ Hans J. Petersen/ Carsten Rathgeber/ Klaus Richter/ Dirk Scharf: IT-Handbuch. IT-Systemkaufmann/ -frau, Informatikkaufmann/ -frau. Westermann, 2013
* Interne Informationsplattform basierend auf einem SharePoint

# Selbsterklärung

Ich versichere durch meine Unterschrift, dass ich das zugrunde liegende Projekt und diesen Bericht zur betrieblichen Projektarbeit (Projektbericht) selbstständig und ohne fremde Hilfe angefertigt habe. Alle aufgeführten Preise sind netto und entsprechen nicht der Realität. Sie sind aus wettbewerbstechnischen Gründen fiktiv angegeben. Die Arbeit hat in dieser Form keiner anderen Prüfungsinstitution vorgelegen!



Ingolstadt, 20.04.2016   
Ort, Datum

**Erklärung des Ausbildungsbetriebes**

Das Projekt wurde, wie in der Dokumentation dargestellt, in unserem Unternehmen durchgeführt.

Ingolstadt, 20.04.2016   
Ort, Datum

# Anlagen

Abbildung 1:

Serverraum



Abbildung 2:

Alter Server



Abbildung 3:  
2 Netzwerkschränke

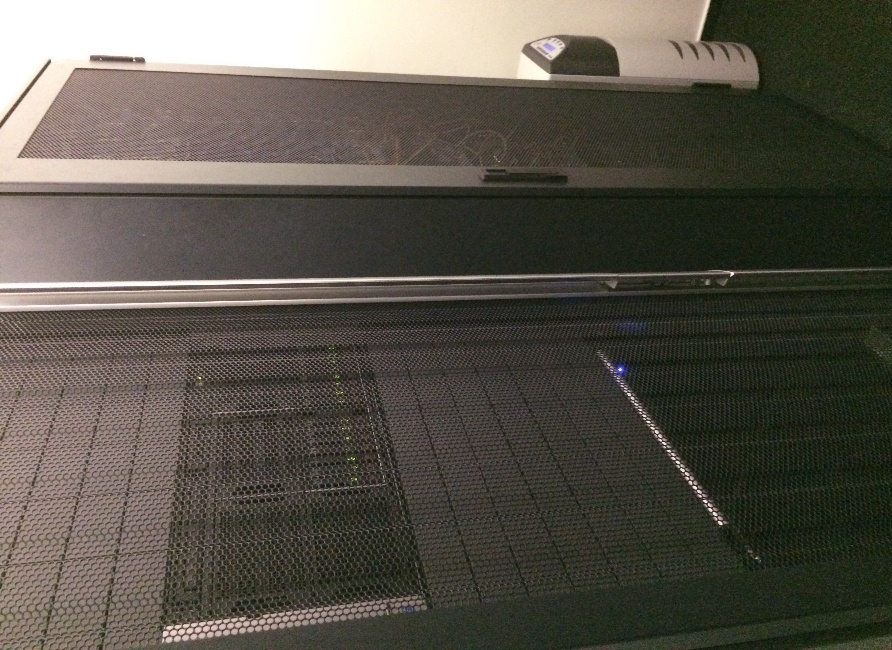


Abbildung 4:  
HP ProLiant ML350P G9 Rack Server



Abbildung 5:  
Verteilerswitche und Telefonanlage

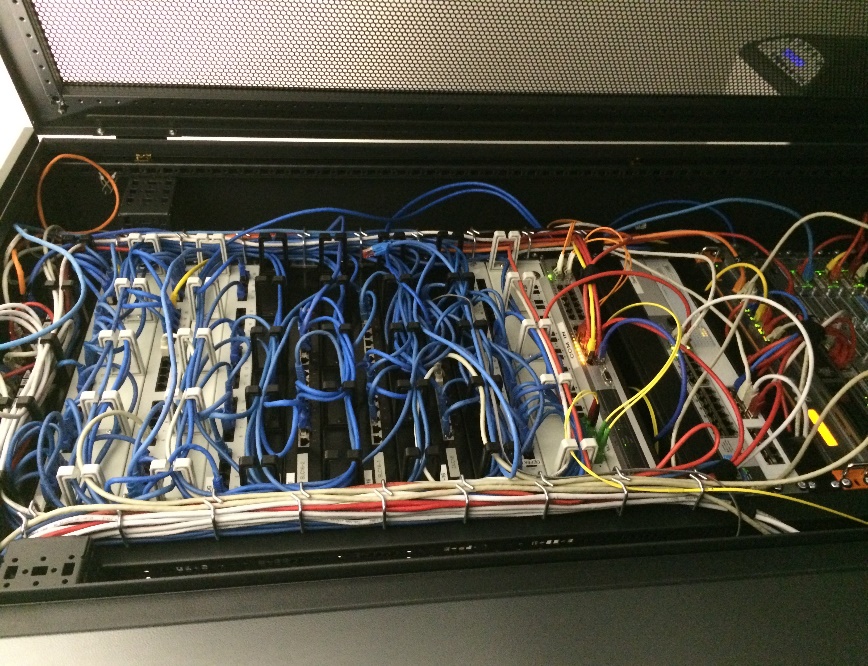


Abbildung 6:  
USV und Backup System

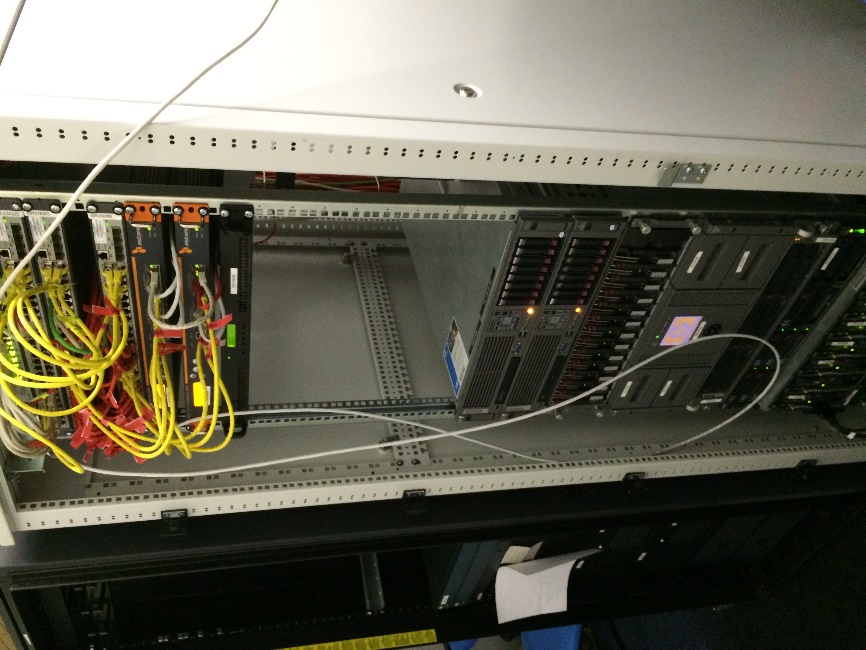


Abbildung 7:  
Etagenverteiler



Abbildung 8:

Netzwerkplan des gesamten Projekts

