# Einleitung

Einen wunderschönen guten Tag zusammen!   
  
Herzlich willkommen zu dieser Präsentation. Mein Name ist Abdulsalam und ich werde im Folgenden mein Abschluss Projekt vorstellen.

Beginnen wir zunächst mit dem Inhalt der Präsentation. Ich möchte Ihnen im Folgenden zunächst das Unternehmen einen Ausbildungsbetrieb vorstellen, dann über die Analyse bzw. den ISI-/ SOLL-Zustand zu Beginn des Projekts sprechen.

Dazu noch die Projekt Planung, Projekt Durchführung bzw. Entwurf und Implementierung des Projektes behandeln und zuletzt noch einen Kürzen Review und Ausblick vorstellen.

Also beginnen wir zunächst mit dem Unternehmen. Ich arbeite bei der Meier AG in der Entwicklungsabteilung, die sich in Wesel Am Rhein befindet und im Jahr 2018 gegründet wurde und die drei Bereiche: Business Solutions, Onlineprodukte & Lösungen und Immobilien & Investitionen fassen die Geschäftsfelder der Meier AG.

Das Unternehmen beschäftigt rund die 13 Mitarbeiter bzw. Mitarbeiterin.

Die Mitarbeiter sind 1 Geschäftsführer, 1 Mediengestalterin, 1 Webentwickler, 10 Auszubildenden und 1 Mitarbeiter des Jahres (Kaffeemaschine). Ohne ihre Leitung kenn keiner Arbeiten 😉

# Analyse

Damit kommen wir auch zur Analyse, die mit dem Ist-Zustand anfängt.

# IST-Zustand

Im Betrieb der Meier Ag befinden sich zurzeit 10 Auszubildenden, die in verschiedenen Bereichen ausgebildet werden. Ja, natürlich führt Jeder Auszubildende in der Ausbildung ein Berichtsheft. Dies geschieht, indem jeder Auszubildende einen wöchentlichen Ausbildungsnachweis schreibt. Das Schreiben des Ausbildungsnachweis wird durch die Bearbeitung einer PDF-Datei Vorlage erfolgen, wodurch mehrere Probleme auf verschiedenen Ebenen auftreten könnten.

Die am häufigsten auftretenden Probleme sind:

Ein falscher Zeitraum des Ausbildungsnachweis oder eine falsche Nummerierung der Ausbildungsnachweise oder Ein falsches Ausbildungsjahr einzutragen, da diese Daten händig eingetragen und mit den letzten Ausbildungsnachweis verglichen werden.

Nach der Erstellung des Ausbildungsnachweis, wird er vom Auszubildenden ausgedruckt und vom Ausbilder überprüft. Im Fall der Ablehnung des Ausbildungsnachweis, muss er vom Auszubildende nochmal bearbeitet werden und nochmal vom Ausbilder überprüft werden, wodurch eine Verschwendung von Zeit und Papier entsteht.

Außerdem könnten die überprüften und freigegebenen Ausbildungsnachweise verloren gehen, wenn der Auszubildende sie nicht richtig und ordentlich abheftet.

Aufgrund dieser Probleme wurde die Entscheidung getroffen, dass wir effizienter werden wollen, dass unsere Auszubildenden ihre Berichtshefte effizienter schreiben sollen. Also, dazu ist das Auszubildenden Verwaltungssystem gedacht.

# Soll-Zustand

SOLL also, was erwarten wir aus diesem System?

1- Die Erreichbarkeit des Systems ist hoch angesagt, deshalb sollte das System jederzeit an jedem Ort erreichbar sein.

2- Das System sollte die Möglichkeit haben, das ganze Prozess der Erstellung, Überprüfung, Abheftung und Generierung der Ausbildungsnachweis zu automatisieren, also,

3-Der Auszubildende kann in seinem Konto sein Berichtsheft durchführen, indem ein Formular für einen wöchentlichen Ausbildungsnachweis von ihm ausgefüllt wird. Das Formular generiert eine PDF-Datei für den Ausbildungsnachweis, Von dem System wird das Datum, das Ausbildungsjahr und die Nummerierung des Ausbildungsnachweis automatisch ergänzt.

4-Der Ausbilder kann in seinem Konto neuen Auszubildende erstellen. Dazu verwaltet er die Konten seiner jetzigen Auszubildenden und überprüft ihre Ausbildungsnachweise und kann sie entweder freigeben oder mit einem Grund in Form von einem Hinweis ablehnen. Der Hinweis erscheint beim Auszubildenden in der Maske des entsprechenden Ausbildungsnachweis. Dieser muss dann vom Auszubildenden nochmal bearbeitet werden.

5-Außerdem das System sollte auf ein Admin Konto der alle Konten und Berichtshefte im System verwalten kann, verfügen.

# Projektplanung

Ich habe mich in diesem Fall für das agiler Entwicklungsprozess Scrum entschieden Dabei geht es darum, möglichst schnell auf die innerhalb der Entwicklung auftretenden Änderungen der Anforderungen reagieren zu können und einen einwandfreien Entwicklungsprozess zu haben, ohne dabei viele Beschränkungen zu definieren.

Ein Überblick über die kalkulierten Zeiten für das Projekt sieht man hier unten der Tabelle. Eine detaillierte und erweiterte Zeitplanung kann man auch nochmal in der Dokumentation entnehmen.

|  |  |
| --- | --- |
| Projektphase | Geplante Zeit |
| Planung und Analyse | 6 Stunden |
| Realisierung | 43 Stunden |
| Dokumentation | 10 Stunden |
| Abschlusstest mit Bugfixing | 9 Stunden |
| Puffer | 2 Stunden |
| Gesamt | **70 Stunden** |

#### [Tabelle 1: Grobe Zeitplanung](#_Tabelle_1:_Grobe_1)

Noch ein Wort zur Kosten Planung. Das Projekt umfasst ein Investment von 2150 Euro. in diesen 2150 Euro sind alle entstehenden Kosten, die in der Zeit der Entwicklung des Systems, unmittelbar anfallen. Also dabei werden sowohl die Personalkosten als auch die Hard- und Softwarekosten berücksichtigt.  
Dazu sind die Ressourcenkosten in Betracht genommen, also die Ressourcen sind in dem Fall mein PC und einen Büroarbeitsplatz, die Nutzung von Hard- und Software sowie Fixkosten z. B. für Strom.  
in dieser Tabelle sind die Kosten nach den Projektaktivitäten und Projektmaßnahmen dargestellt.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Vorgang | Mitarbeiter | Zeit | Personal | Ressourcen | Gesamt |
| Entwicklungskosten | 1 x Auszubildender | 70 Std. | 700,00 € | 1.050,00 € | 1.750,00 € |
| Fachgespräch | 1 x Mitarbeiter | 6 Std. | 150,00 € | 90,00 € | 240,00 € |
| Code-Review | 1 x Mitarbeiter | 3 Std. | 75,00 € | 45,00 € | 120,00 € |
| Abnahme | 1 x Mitarbeiter | 1 Std. | 25,00 € | 15,00 € | 40,00 € |
|  |  |  |  |  | **2.150,00 €** |

#### Tabelle 2: Kostenaufstellung

Projekt Durchführung

Ja, nehmen wir also an dieser Stelle über zur Projekt Durchführung. Was gab es da in diesem Zusammenhang zu tun bei der Entwicklung des Auszubildenden Verwaltungssystem?

Entwurfsphase

Das System wurde mithilfe des freien, quelloffenen PHP-Webframework Laravel entwickelt, da es unterstützt viele Services, die für das System eine notwendig spielen. ja, Dazu Laravel unterstützt eine ganz einfache Integration für andere Open-Source Bibliotheken, die mit dem PDF-Vorlage umgehen sollen.

Dazu verwendet das Laravel-Framework verwendet das Model View Controller (MVC)-Architektur. Die drei Teile bestehen aus:

* Also das Modell ist dafür zuständig die Kommunikation mit dem Datenbank zu halten also die Daten abzurufen, aktuallisieren, speichern oder löschen
* Die View ist einfach zuständig für Darstellung und Präsentation von Ergebnisse oder für die Daten, die wir von dem Controller bekommen. Dazu die Benutzeroberfläche die wird in dem View implementiert
* Der Controller ,also Bezug auf seinen Namen halt, kontrolliert und steuert das System. Dazu enthält er die Logik von das ganze System und bearbeitet alle Ergebnisse die er von einem Model bekommt

Diese Abstraktionen vereinfachen spätere Änderungen und das Ausführen von Erweiterungen auf dem System.

Bei der Auswahl der Programmiersprache waren die Möglichkeiten eingeschränkt, weil das ausgewählte Laravel-Webframework lediglich nur PHP unterstützt. Auf die Erstellung einer Nutzwertanalyse zur Auswahl der Programmiersprache wurde deshalb verzichtet.

Für die Benutzeroberfläche (User Interface - UI) wurde entschieden, eine CSS-Framework „Tailwind CSS“ auszuwählen, da dieses Framework responsive Utilities bietet, über die die Benutzeroberfläche ganz einfach für unterschiedliche Bildschirmgrößen angepasst werden kann.

Für die Erstellung der Benutzeroberfläche ist das Tailwind CSS-Framework zu verwenden. Das Tailwind CSS-Framework wird mit einer Reihe von Utility-Klassen geliefert, die zur Erstellung des von der Design-Abteilung bereitgestellten Designs verwendet werden können. Tailwind CSS ist nicht meinungsbildend, so dass das Design der Elemente und Komponenten völlig frei gewählt werden kann

Die View Komponente wird mit [HTML](#_HTML) implementiert und mit den Utility-Klassen, die vom Tailwind-CSS-Framework bereitgestellt werden, gestylt.

Diese Art von Implementierung der Benutzeroberfläche bietet den Vorteil, dass bei Änderungen in einer Benutzeroberfläche nur die View Komponente bearbeitet werden muss, ohne etwas anderes zu ändern

Um einen Gesamtüberblick über alle Daten zu erhalten, die für das System relevant sind, wurde zunächst eine Analyse des Systems durchgeführt, um zu wissen, welche Daten gespeichert werden müssen und wie sie zueinander in Beziehung stehen.

Zu diesem Zweck wurde ein Entity-Relationship-Model (ERM) erstellt, um die Typen und ihre Beziehungen grafisch darzustellen. Dieses finden Sie in [Anhang](#_A._Anhang_1) [A.5 Entity-Relationship-Model](#_A.5_Entity-Relationship-Model) auf S. IX.

Die folgenden Entitätstypen wurden identifiziert: User – Department – Permision – Role –   
Weekly Report – Apprentice

Das fertig entwickelte System wird bei einem externen professionellen Webhoster gehostet.

## Implementierungsphase

Kommen wir also in diesem System zu den Apps, die hier im Rahmen dieses Systems verwendete Software mal in drei Kategorien eingeteilt

Zuerst wurde das Open-Source-Programm namens PHPMyAdmin verwendet, um einen neue Datenbankinstanz mit dem Namen "auszubildendes\_verwaltungssystem" zu erstellen

Dazu bietet Laravel eine komponent an durch diese comprend war es möglich die datenbanktabellen für das System zu generieren und in dem erstellten datenbankinstanz zu integrieren

Die Meier AG stellt jedem Entwickler eine kostenpflichtige Lizenz für die Jetbrains-Programme zur Verfügung. Aus diesen Programmen wurde PHPStrom als [IDE](#_IDE) für die Programmierung des gewünschten Systems ausgewählt, weil es die Programmiersprache PHP hervorragend unterstützt und den Programmierprozess mit Hilfe von Shortcuts, Autovervollständigung und Syntaxhervorhebung vereinfacht. Dadurch wird der Code besser lesbar, ohne dass dem Entwickler Kernaufgaben der Programmierung abgenommen werden.

Zusätzlich zur IDE wird ein virtueller Server benötigt, da PHP in einer Server-Umgebung arbeitet und im Browser nicht interpretiert werden könnte. Daher wird eine kostenlose Version von [WAMP](#_WAMP) (eine völlig kostenlose lokale Webserver-Software) verwendet.

WAMP erstellt eine virtuelle Server-Umgebung mit Hilfe von Apache-Webserver, welches inklusive bei der Installation von WAMP mitkommt. Apache-Webserver ist dafür verantwortlich die Anfragen zu verarbeiten und Web-Assets und Inhalte über HTTP bereitzustellen. Ein lokaler Webserver ist notwendig, wenn man PHP-Skripte auf einem Computer schreiben und ausführen möchten, ohne sie irgendwo auf einen Server hochladen zu müssen.

Im nächsten Schritt wird „[Composer](#_Composer)“ auf dem Computer installiert, ein Tool zur Verwaltung von Abhängigkeiten in PHP. Es erlaubt, die Bibliotheken zu deklarieren, von denen ein Projekt abhängt, und es wird diese verwalten (installieren/aktualisieren).

Außerdem muss eine neue Instanz vom Laravel in das entsprechende Verzeichnis installiert werden. Dann kann man mit der Integration der Open-Source Bibliotheken, die mit den Berechtigungen der Konten im System und die mit der PDF-Vorlage umgehen können.

Nachdem die Entwicklungsumgebung eingerichtet wurde, kann die eigentliche Programmierung beginnen. Im ersten Iterationsschritt hat der Autor die Datenbanktabellen mit Hilfe der eingebauten Komponenten von [Laravel](#_Laravel) implementiert. Zu jeder Tabelle wird ein Model erstellt, das den singulären Namen der Tabelle hat und mit ihr kommuniziert. Ein Ausschnitt der Tabelle „weekly\_reports“, worin die notwendige daten für einen ABN gespeichert werden, befindet sich im [Anhang](#_A._Anhang_1) [A.8 Datenbank Tabelle: Weekly Reports](#_A.8_Datenbank_Tabelle:_1) auf der Seite XII befindet.

Innerhalb dieser Models werden die Beziehungen zwischen den Tabellen vordefiniert. Dazu hat jedes Model bestimmte Attribute die man innerhalb der Programmierung verwenden kann. In diesem Model werden drei Konstanten „PENDING – APPROVED – REJECTED“ festgelegt, die den Status jedes ABN beschreibt. Ein Ausschnitt des Models „WeeklyReport“, welches den Ausbildungsnachweis darstellt, befindet sich im [Anhang](file:///C:\Users\abdul\Desktop\projecktdokomentation\doco.docx#_A._Anhang_1) [A.9 Model WeeklyReport](#_A.9_Model_WeeklyReport_1) auf der Seite XIII.

Im nächsten Iterationsschritt werden zwei Open-Source-Bibliotheken mit Hilfe von "[Composer](#_Composer)" installiert. Die erste heißt "spatie/laravel-permission", die für die Definition der Rollen und Berechtigungen der einzelnen Konten im System verantwortlich ist.

Die Bibliothek tut dies, indem sie die eingebauten Komponenten von Laravel verwendet, um Tabellen in der Datenbank zu erstellen. In diesen Tabellen werden die Rollen und Berechtigungen gespeichert. Denn diese werden verwendet, um im nächsten Schritt die Berechtigungen eines Kontos in einem Controller zu überprüfen. Ein Ausschnitt des Controllers „WeeklyReportsController“, welches für die Verwaltung eines [ABN](#_ABN) zuständig ist, befindet sich im [Anhang](file:///C:\Users\abdul\Desktop\projecktdokomentation\doco.docx#_A._Anhang_1) [A.10 WeeklyReportsController](#_A.10_WeeklyReportsController_1) auf der Seite XIV befindet.

Die zweite heißt "barryvdh/laravel-dompdf". Sie stellt eine Komponente zur Verfügung, von der eine neue Instanz erzeugt und mit einer View aus [Laravel](#_Laravel) und den benötigten Daten für eine [ABN](#_ABN) aus der Datenbank gefüttert wird. Diese Komponente erzeugt dann eine PDF-Datei mit den bereitgestellten Daten und der View. Ein Screenshot der generierten PDF für einen [ABN](#_ABN) befindet sich im [Anhang](file:///C:\Users\abdul\Desktop\projecktdokomentation\doco.docx#_A._Anhang_1) [A.11 Vom System generiertes Ausbildungsnachweis](#_A.11_Vom_System) auf der Seite XVI befindet.

Laravel bietet eine Komponente namens „Middleware“, die als Brücke zwischen einer Anfrage und einer Antwort dient. Sie ist als eine Art Filtermechanismus zu beschreiben. Laravel enthält zum Beispiel eine Middleware, die prüft, ob der Benutzer der Anwendung authentifiziert ist oder nicht. Diese Bibliothek verwendet diese Komponente, um die Berechtigungen und Rollen der einzelnen Konten im System zu überprüfen

Die Controller werden im nächsten Iterationsschritt erstellt und so benannt, dass ihre Aufgaben einigermaßen beschrieben werden. Jeder Controller hat eine oder mehrere Methoden und jede Methode hat die Fähigkeit, Daten von einem Model zu empfangen oder zu geben.

In jedem Controller wird geprüft, ob das authentifizierte Konto bestimmte Berechtigungen besitzt. Dies wird mit Hilfe der Komponente aus der Bibliothek "spatie/laravel-permission" durchgeführt,

Dazu sorgt Jeder Controller dafür, dass die Datenschicht (das Model) mit der Präsentationsschicht (das View) verheiratet wird. Die Logik einer bestimmten Aufgabe ist in ihm implementiert. Zum Beispiel empfängt der Controller die Daten für einen [ABN](#_ABN) vom entsprechenden Model, prüft dann die Integrität der Daten und erstellt eine neue Instanz der Komponente, die für die Generierung der [ABN](#_ABN) verantwortlich ist, und übergibt ihr die Daten mit der entsprechenden View.

Der nächste Iterationsschritt ist der Aufbau der View-Komponenten unter Verwendung des Tailwind-CSS-Frameworks und der bereits von der Design-Abteilung erstellten Markups.

# Testing

Das System wurde von meinem Ausbilder und ein paar Auszubildenden getestet. Sie haben alle möglichen Funktionen des Systems getestet. Dazu haben neuen Kontos für neuen Ausbilder und neue Auszubildenden erstellt Dazu haben die Auszubildenden neue Ausbildungsnachweise geschrieben und dafür wurde einen entsprechenden PDF-Datei generiert.

# Review und Ausblick

Bei dem Rückblick auf das Projekt kann festgestellt werden, dass alle vorher definierten Anforderungen gemäß Pflichtenheft erfüllt wurden. Es ist aber zu erkennen, dass es einen gewissen Zeitunterschied bei den verschiedenen Phasen gab, was die Soll/Ist Vergleichstabelle zeigt. Da die Phasen Planung und Analyse und die Realisierung Zeit in Anspruch nahm, konnten die benötigten Stunden von der Phase Abschlusstest mit Bugfixing abgenommen werden. Somit konnte das Projekt in den vorgegebenen 70 Stunden umgesetzt werden.

|  |  |  |  |
| --- | --- | --- | --- |
| Projektphase | Soll | Ist | Differenz |
| Planung und Analyse | 6 Stunden | 8 Stunden | +2 Stunden |
| Realisierung | 43 Stunden | 48 Stunden | +5 Stunden |
| Dokumentation | 10 Stunden | 10 Stunden | 0 Stunden |
| Abschlusstest mit Bugfixing | 9 Stunden | 4 Stunden | -5 Stunden |
| Puffer | 2 Stunden | 0 Stunden | -2 Stunden |
| Gesamt | **70 Stunden** | **70 Stunden** | **0 Stunden** |

Das System wird zurzeit in unserem Unternehmen aktive von allen Mitarbeitern bzw. Mitarbeiterinnen und Auszubildenden verwendet. Über 350 Ausbildungsnachweise wurden bis heute geschriebnen un überprüft  
  
Dazu wurde 2 neue Funktionen durchgeführt, und zwar das jeder Ausbilder und Auszubildende sein Unterschrift in dem System einfügen kann. Dieser Unterschriften werden verwendet, wenn ein Auszubildende einen Ausbildungsnachweis schreibet und wenn der Ausbildungsnachweis vom Ausbilder überprüft und freigegeben wird

Quellen

Scrum image <https://bit.ly/34MXh4W>

mysql <https://bit.ly/2Sg1kE1>

phpstorm <https://bit.ly/2TO2YgR>

composer <https://bit.ly/3gfGYCK>

php <https://bit.ly/3ikB9qv>

tailwind css <https://bit.ly/3fUvgi1>

wamp <https://bit.ly/3z50WJf>

phpMyAdmin <https://bit.ly/3ikUOqo>