**Individuelle Praktische Arbeit**

**Trophäen-Verwaltungs-App**

Dieses Dokument enthält den Bericht zur individuellen praktischen Arbeit von Julien Rädler.

Inhalt

[Aufgabenstellung 7](#_Toc114555035)

[Trophäen-Verwaltungs-App 7](#_Toc114555036)

[Ausgangslage 7](#_Toc114555037)

[Detaillierte Aufgabenstellung 7](#_Toc114555038)

[Mitteln und Methoden 9](#_Toc114555039)

[Neue Lerninhalte 9](#_Toc114555040)

[Projektaufbauorganisation 9](#_Toc114555041)

[Projektorganisation 9](#_Toc114555042)

[Ausführungszeitraum 9](#_Toc114555043)

[Termine 9](#_Toc114555044)

[Involvierte Personen 10](#_Toc114555045)

[Vorkenntnisse 10](#_Toc114555046)

[Vorarbeiten 10](#_Toc114555047)

[Benützte Firmenstandards 10](#_Toc114555048)

[Verwendete Projektmanagemantmethode 11](#_Toc114555049)

[Versionisierung und Datensicherheit 11](#_Toc114555050)

[Arbeitsprotokoll 13](#_Toc114555051)

[Dienstag, 30. August 2022 13](#_Toc114555052)

[Mittwoch, 31. August 2022 14](#_Toc114555053)

[Montag, 05. September 2022 15](#_Toc114555054)

[Dienstag, 06. September 2022 16](#_Toc114555055)

[Mittwoch, 07. September 2022 17](#_Toc114555056)

[Dienstag, 13. September 2022 18](#_Toc114555057)

[Mittwoch, 14. September 2022 19](#_Toc114555058)

[Montag, 19. September 2022 20](#_Toc114555059)

[Montag, xx. Monat yyyy 21](#_Toc114555060)

[Montag, xx. Monat yyyy 22](#_Toc114555061)

[Montag, xx. Monat yyyy 23](#_Toc114555062)

[Montag, xx. Monat yyyy 24](#_Toc114555063)

[Kurzfassung 25](#_Toc114555064)

[Ausgangssituation 25](#_Toc114555065)

[Umsetzung 25](#_Toc114555066)

[Ergebnis 25](#_Toc114555067)

[Informieren 26](#_Toc114555068)

[Ausgangslage 26](#_Toc114555069)

[Abklärungen 26](#_Toc114555070)

[Verstandene Aufgabenstellung und Ziel der Arbeit 26](#_Toc114555071)

[Verfeinerung des Auftrages 26](#_Toc114555072)

[Projektumfeld und Systemgrenzen 26](#_Toc114555073)

[Planen 27](#_Toc114555074)

[Verwendete Projektmanagementmethode 27](#_Toc114555075)

[Versionierung und Datensicherheit 27](#_Toc114555076)

[Priorisierung der Tätigkeiten 27](#_Toc114555077)

[Kernfeature ABC 27](#_Toc114555078)

[Use-Case für Abc 27](#_Toc114555079)

[Klassendiagramm 27](#_Toc114555080)

[GUI-Design / Mockup 28](#_Toc114555081)

[Konzeptioneller Aufbau 28](#_Toc114555082)

[Kernfeature ZYZ 28](#_Toc114555083)

[Use-Case für zyz 28](#_Toc114555084)

[Klassendiagramm 28](#_Toc114555085)

[GUI-Design / Mockup 28](#_Toc114555086)

[Konzeptioneller Aufbau 29](#_Toc114555087)

[Geplantes Vorgehen für die Qualitätssicherung 29](#_Toc114555088)

[Anmerkungen zum Zeitplan 29](#_Toc114555089)

[Begründung für Abweichungen zum Zeitplan 29](#_Toc114555090)

[Entscheiden 30](#_Toc114555091)

[Datenbankfeld für Farbe 30](#_Toc114555092)

[Beschreibung Variante mit frei wählbarem Text 30](#_Toc114555093)

[Beschreibung Variante mit vorgegebener Auswahl an Farben 30](#_Toc114555094)

[Kurzbeschreibung der Entscheidungskriterien 30](#_Toc114555095)

[Datenbank Redundanzen 31](#_Toc114555096)

[Beschreibung Variante mit zusätzlichen Tabellen 31](#_Toc114555097)

[Beschreibung Variante ohne zusätlzichen Tabellen 31](#_Toc114555098)

[Kurzbeschreibung der Entscheidungskriterien 31](#_Toc114555099)

[Realisieren 32](#_Toc114555100)

[Abbildung des Gesamtsystems 32](#_Toc114555101)

[Interaktionen zwischen den Teilsystemen 32](#_Toc114555102)

[Vorgehensweise 32](#_Toc114555103)

[Grundaufbau / Architetkur des Projektes 33](#_Toc114555104)

[Anbindung an die Datenbank 33](#_Toc114555105)

[Implementierung des Kernfeatures JSON Funktionalität 33](#_Toc114555106)

[Funktionsweise des Kernfeatures JSON Funktionalität 34](#_Toc114555107)

[Ziel des Kernfeatures JSON Funktionalität 34](#_Toc114555108)

[Effektives Produkt 34](#_Toc114555109)

[Implementierung des Kernfeatures Request Validierung 35](#_Toc114555110)

[Funktionsweise des Kernfeatures Request Validierung 35](#_Toc114555111)

[Ziel des Kernfeatures Request Validierung 36](#_Toc114555112)

[Effektives GUI / Produkt 36](#_Toc114555113)

[Implementierung des Kernfeatures Swagger Dokumentation 37](#_Toc114555114)

[Funktionsweise des Kernfeatures Request Validierung 37](#_Toc114555115)

[Ziel des Kernfeatures Request Validierung 38](#_Toc114555116)

[Effektives GUI / Produkt 38](#_Toc114555117)

[Kontrollieren 39](#_Toc114555118)

[Beschreibung der Randbedingungen / Testanlage (Umfeld) 39](#_Toc114555119)

[Eingesetzte Testmittel und -methoden 39](#_Toc114555120)

[Beschreibung der Testszenarien 39](#_Toc114555121)

[Testprotokoll zum Testszenario User 40](#_Toc114555122)

[Testprotokoll zum Testszenario Farben 40](#_Toc114555123)

[Testprotokoll zum Testszenario Typen 40](#_Toc114555124)

[Testprotokoll zum Testszenario Kategorien 41](#_Toc114555125)

[Testprotokoll zum Testszenario Trophäen 42](#_Toc114555126)

[Bugfixing 45](#_Toc114555127)

[Testprotokoll nach dem Bugfixing 45](#_Toc114555128)

[Einführung beim Kunden 45](#_Toc114555129)

[Auswerten 46](#_Toc114555130)

[Reflexion der Vorgehensweise 46](#_Toc114555131)

[Bewertung des Produktes 46](#_Toc114555132)

[Persönliches Schlusswort und Bilanz 46](#_Toc114555133)

[Glossar 47](#_Toc114555134)

[A 47](#_Toc114555135)

[B 47](#_Toc114555136)

[C 47](#_Toc114555137)

[D 47](#_Toc114555138)

[E 47](#_Toc114555139)

[F 47](#_Toc114555140)

[G 47](#_Toc114555141)

[H 47](#_Toc114555142)

[I 47](#_Toc114555143)

[J 47](#_Toc114555144)

[K 48](#_Toc114555145)

[L 48](#_Toc114555146)

[M 48](#_Toc114555147)

[N 48](#_Toc114555148)

[O 48](#_Toc114555149)

[P 48](#_Toc114555150)

[Q 48](#_Toc114555151)

[R 48](#_Toc114555152)

[S 48](#_Toc114555153)

[T 48](#_Toc114555154)

[U 49](#_Toc114555155)

[V 49](#_Toc114555156)

[W 49](#_Toc114555157)

[X 49](#_Toc114555158)

[Y 49](#_Toc114555159)

[Z 49](#_Toc114555160)

[Quellenverzeichnis 50](#_Toc114555161)

[Literaturverzeichnis 50](#_Toc114555162)

[Internetquellen 50](#_Toc114555163)

[Abbildungsverzeichnis 51](#_Toc114555164)

[Tabellenverzeichnis 51](#_Toc114555165)

[Anhang 51](#_Toc114555166)

[Projektjournal 52](#_Toc114555167)

[Gesprächsprotokoll vom xx. Monat yyyy 52](#_Toc114555168)

[Erster Expertenbesuch 54](#_Toc114555169)

[Zweiter Expertenbesuch 54](#_Toc114555170)

[Code 55](#_Toc114555171)

[Filename.endung 55](#_Toc114555172)

[Filename.endung 55](#_Toc114555173)

[Zusätzliche Manuals, Skripts und weiteres 56](#_Toc114555174)

[Handbuch A 56](#_Toc114555175)

[Skript B 56](#_Toc114555176)

Teil 1: Umfeld und Ablauf

# Aufgabenstellung

Trophäen-Verwaltungs-App

Die TwoFold AG ist an der Entwicklung einer Trophäen APP in welcher Trophäen aus einem Sportclub ausgestellt werden. Hierfür soll eine API entwickelt werden, welches ermöglicht Trophäen zu speichern. Es werden Trophäen in einer Datenbank gespeichert. Das Projekt basiert auf der PHP Library Laravel und hat kein Frontend. Auf die Daten wird über API-Endpunkte zu gegriffen.[[1]](#footnote-1)

Ausgangslage

Wir bei der Asperger AG bieten Lernenden auf dem Autismus-Spektrum die Möglichkeit in einem geschützten Rahmen eine Lehre zu machen und fit für den ersten Arbeitsmarkt zu werden. Durch unser spezielles Setting und die Finanzierung durch die IV dürfen wir keine gewinnbringenden, externen Aufträge annehmen, weswegen wir auf interne Projekte bauen. Nebst Informatik bieten wir auch Lehren als Interactive Media Designer (IMD) an. Einer unserer sehr sportlichen IMD-Lernenden hatte sich das Konzept und das zugehörige Screendesign für eine Trophäen-Verwaltungs-App ausgedacht. Da dieser Lernende regelmässig mit seinem Sportclub an Wettkämpfen in der ganzen Schweiz teilnimmt, wollte er eine App schaffen, in welcher man die gewonnenen Trophäen und Medaillen speichern, verwalten und sich darüber auch austauschen kann. Mit dieser IPA soll jetzt das API-Backend für dieses Projekt gestartet werden.[[2]](#footnote-2)

Detaillierte Aufgabenstellung

Es soll ein API-Backend aufgebaut werden für eine Trophäen-App in welcher gewonnene Trophäen gespeichert werden können. Diese Trophäen können zudem zu Kategorien zugewiesen werden.

* Es ist ein API-Backend mit dem Laravel Framework in PHP umzusetzen
* Die Daten sollen in einer geeigneten Struktur in einer Datenbank abgelegt werden
* Die Datenbank-Struktur wird mithilfe einer Migration automatisch aufgebaut
* Die einzelnen Datenbanken werden mit Model-Klassen in Laravel abgebildet
* Für die verschiedenen Inhalte werden Routen bereitgestellt
* Die Routen werden mit den üblichen Verben angesprochen (Create = POST, Read = GET, Update
* = PUT, Delete = DELETE)
* Die API-Routen sollen in einer geeigneten Form dokumentiert werden für die künftigen Frontend Entwickler (Swagger o.ä.)
* Die Kommunikation mit der API läuft über JSON
* Anfragen werden in Request-Klassen validiert und aufbereitet
* Antworten werden in Ressourcen-Klassen aufbereitet und zurückgesendet
* Fehler in den Eingaben werden dem Nutzer zurückgeschickt
* Die Routen und Inhalte sind geschützt, sodass nur eingeloggte User Zugriff auf die eigenen Inhalte haben
* Die Routen sollen händisch getestet und in einem Testprotokoll aufgeführt werden
* Das Projekt soll lokal laufen und muss noch nicht veröffentlicht werden
* Das Projekt muss sauber, nachvollziehbar und leicht erweiterbar sein

Endpoints:

Users:

* User sollen sich registrieren und einloggen können (ohne Mail-Validierung)

Categories:

* Kategorien sollen CRUD Möglichkeiten haben
* Kategorien beinhalten einen Namen + eine Farbe (Bsp: Tennis, Blau oder Fussball, Rot)

Trophies:

* Trophäen sollen CRUD Möglichkeiten haben
* Trophäen beinhalten:
* Typ (Trophäe/Medallie)
* Titel
* Platzierung
* Datum
* Kategorie
* Ort
* Gegner
* Spielstand
* Preisgeld
* Name des Clubs
* Bild
* Trophäen können sortiert und gefiltert werden
* Sortierungen: Datum, Platzierung
* Filterung: Typ, Platzierung, Kategorien

Erwartet wird ein funktionierendes API-Backend, aufgebaut in Laravel, mit welchem bspw. über

Postman kommuniziert werden kann. Es soll eine funktionierende Registrierung eines Nutzers

möglich sein, mit anschliessendem Login. Der eingeloggte Nutzer sollte dann die Möglichkeit haben

Kategorien zu erstellen, bearbeiten und zu löschen. Zudem soll der Nutzer die Möglichkeit haben,

Trophäen zu erstellen, diese einer Kategorie zuzuweisen und mit allen Inhalten zu speichern,

bearbeiten und zu löschen. Auf der Übersichtsseite sieht der Nutzer alle seine eingetragenen

Trophäen und kann diese sortieren und filtern.[[3]](#footnote-3)

Mitteln und Methoden

XAMPP, Laravel, PHP, Swagger, API-Testingsoftware.[[4]](#footnote-4)

Neue Lerninhalte

Dem Lernenden sind grundsätzlich alle Technologien bekannt. Mit Swagger ist er aber noch unerfahren.[[5]](#footnote-5)

# Projektaufbauorganisation

Projektorganisation

|  |  |
| --- | --- |
| Auszubildender | Lehrbetrieb |
| Julien Rädler  Im Dreispitz 235  8050, Zürich | Asperger AG  Thurgauerstrasse 54  8050 Zürich |

Tabelle 1 - Adressinformation Auszubildender

Ausführungszeitraum

|  |  |  |  |
| --- | --- | --- | --- |
| Di 30. August 2022 | Do 31. August 2022 | Mo 05. September 2022 | Di 06. September 2022 |
| Mi 07. September 2022 | Di 13. September 2022 | Mi 14. Sebtember 2022 | Mo 19. September 2022 |
| Di 20. September 2022 | Mi 21. September 2022 | Mo 26. September 2022 | Di 27. September 2022 |

Tabelle 2 – Ausführungszeitraum

Termine

|  |  |
| --- | --- |
| Was? | Wann? |
| 1. Expertenbesuch | 05. September 2022 |
| 2. Expertenbesuch | 20. September 2022 |
| Präsentation, Demonstration, Fachgespräch | Noch unbekannt |

Tabelle 3 – Termine

Involvierte Personen

|  |  |
| --- | --- |
| Kontaktinformation | Rolle |
| Domenik Hofer | Hauptexperte |
| Domenik Hofer | 2. Experte |
| Domenik Hofer | Verantwortliche Fachkraft |
| Domenik Hofer | Berufsbildner |
| Julien Rädler | IPA-Prüfungskandidat |

Tabelle 4 - Involvierte Personen

# Vorkenntnisse

Der Lernende arbeitete bereits mit folgenden Web-Technologien:

- HTML (3,5 Jahre)

- CSS (3,5 Jahre)

- Javascript (3 Jahre)

- PHP (3 Jahre)

- Vue (1 Jahr)

- Laravel (1 Jahr)

- Wordpress (3 Jahre)

- MySQL / MariaDB (3 Jahre)

- Git (2 Jahre)[[6]](#footnote-6)

# Vorarbeiten

Keine.[[7]](#footnote-7)

# Benützte Firmenstandards

Als Dokumentationsvorlage habe ich die aktuelle IPA-Vorlage der TwoFold AG verwendet. Den Quellcode wie

auch die Dokumentation sichere ich auf GitHub.

# Verwendete Projektmanagemantmethode

Als Projektmanagementmethode wird IPERKA verwendet. Hierbei lässt sich das Projekt in folgende sechs

Schritte unterteilen:

* Informieren
* Planen
* Entscheiden
* Realisieren
* Kontrollieren
* Auswerte

# Versionisierung und Datensicherheit

Zur Versionierung verwende ich GitHub. Alle 4 Stunden erstelle ich mindestens einen Commit, um

Datenverlust zu vermeiden.

Git history:

|  |  |
| --- | --- |
| Nummer | Commits |
| 1 | dbd8e113513ced6fcb41b2a369c38971cd2377a9 |
| 2 | d827f47a890e00b5644d857b827b0217b98c3d6f |
| 3 | d205a92ca65465bfa7a3640638834a76e7f95707 |
| 4 | 06b4363fb0789c8515b0843e2865258fe78e3bc5 |
| 5 | 4615048d6d259c7efab88ae87bf5f94b8e6b43be |
| 6 | 83ff8faf20ae4e9d34bee70a903b18df6a886814 |
| 7 | 67829702b1a922e615f137026411078bed8e2559 |
| 8 | 41ea41e8ac92b487bc4486dcbdca7086aa242ad5 |
| 9 | 534004763d056886d66d4bb97d2b00001a5961af |
| 10 | c5016e6341fb2cc72fa1553b1d2fc31a9ec9d8c0 |
| 11 | a952a6dc17dc6bc1695f768da587e450d882ec50 |
| 12 | d0d5bbacf9d62674aa1ccfb1ba16782710f6f68e |
| 13 | a773e2c040c7aedbb723aa4146c7b9ebd077329d |
| 14 | e1170ec2269075cfb6af50740b6b08662ee047c7 |
| 15 | 06c92755b0ec635b61daa89ea04d0b5bae6fa6d4 |
| 16 | 0f3333bfbe38a8a3076dba2d0cb5b9473bd69619 |
| 17 |  |

**Zeitplan**



Abbildung 1: Zeitplan

# Arbeitsprotokoll

Dienstag, 30. August 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Aufgabenstellung lesen  Zeitplan erstellen  Kurzfassung schreiben |
| Ungeplante Arbeiten /  geleistete Überstunden | Es sind keine ungeplanten Arbeiten als solche aufgetreten. Ich habe mich aber entschieden auf Aktivitätsdiagramme zu verzichten. |
| Erreichte Ziele, Erfolgserlebnisse | Ich konnte den Zeitplan zügiger umsetzen als erwartet und liege auch sonst im Zeitplan. |
| Herausforderungen Probleme | Das Erstellen von Diagrammen hat sich als Zeitaufwändiger erwiesen als erwartet. |
| Lösungen | Keine Aktivitätsdiagramme machen. |
| Durchgeführte Tests | - |
| Wissensbeschaffung | Ich habe mich an den Vorlagen der TwoFold AG und dem Kriterienkatalog der PkOrg orientiert. |
| Beanspruchte Hilfeleistung | Domenik Hofer hat mir bei Unklarheiten mit der Aufgabenstellung und bei Fragen zu den Diagrammen weitergeholfen. |
| Vergleich mit dem Soll-Zeitplan | Ich liege im Zeitplan. |
| Persönliche Tagesreflexion | Am Anfang des Tages, kam ich mit dem Zeitplan sehr gut voran. Am Nachmittag war das Auseinandersetzen mit den Diagrammen eine Lernerfahrung. Ich bin aber zufrieden und im Zeitplan. |

Tabelle 5 - Arbeitsprotokoll Tag 1

Mittwoch, 31. August 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Kurzfassung beenden, App-Funktionen planen, Usecases beschreiben, Aktivitätsdiagram erstellen |
| Ungeplante Arbeiten /  geleistete Überstunden | Informieren-Teil der Dokumentation schreiben. Ich hatte die Vorlage der Dokumentation falsch verstanden und musste deshalb Vieles nachtragen. |
| Erreichte Ziele, Erfolgserlebnisse | Das Datenbankdiagram ist fertig geworden. |
| Herausforderungen Probleme | Die Datenbank zu planen war eine grössere Herausforderung als geplant. |
| Lösungen | Ich habe einige Fragen gestellt und mich ansonsten an den Vorgaben orientiert. |
| Durchgeführte Tests | - |
| Wissensbeschaffung | Fragen an Domenik Hofer gestellt. |
| Beanspruchte Hilfeleistung | Domenik Hofer hat einige Unklarheiten, wenn es zur Datenbank kommt, beseitigt. Weiter hat er mir Ratschläge zu den Diagrammen gegeben. |
| Vergleich mit dem Soll-Zeitplan | Ich bin weiterhin dem Zeitplan voraus. |
| Persönliche Tagesreflexion | Ich habe die Datenbank Gestaltung unterschätzt. Da ich aber gut geplant habe, bin ich noch gut im Zeitplan. Ich bin daher zufrieden mit dem was ich heute geleistet habe. |

Tabelle 6 - Arbeitsprotokoll Tag 2

Montag, 05. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Aktivitätsdiagramme erstellen, Datenbankschema erstellen, Expertenbesuch |
| Ungeplante Arbeiten /  geleistete Überstunden | Heute fand eine von der IPA unabhängigen Besprechung statt. Es gab keine Überstunden. |
| Erreichte Ziele, Erfolgserlebnisse | Ich konnte die Entscheidungsziele erreichen und sogar noch die Umgebung einrichten. |
| Herausforderungen Probleme | Durch die Besprechung und den Expertenbesuch konnte ich nur etwa zwei Stunden an der IPA arbeiten. |
| Lösungen | - |
| Durchgeführte Tests | - |
| Wissensbeschaffung | Offizielle Laravel Webseite |
| Beanspruchte Hilfeleistung | Keine |
| Vergleich mit dem Soll-Zeitplan | Ich liege weiterhin leicht vor dem Zeitplan. |
| Persönliche Tagesreflexion | Obwohl ich nicht viel Zeit hatte, konnte ich doch ein wenig Fortschritt machen. Der Expertenbesuch war auch sehr aufschlussreich. |

Tabelle 7 - Arbeitsprotokoll Tag 3

Dienstag, 06. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | API-Testtool festlegen, API-Dokumentationstool festlegen, Umgebung einrichten, Modele erstellen |
| Ungeplante Arbeiten /  geleistete Überstunden | Heute fand das monatliche Gespräch mit Domenik statt. Ich musste den Bericht für das Gespräch vorbereiten. |
| Erreichte Ziele, Erfolgserlebnisse | Die Modelle, Seeder und Migrationen sind beendet. |
| Herausforderungen Probleme | Bei der Datenbankanbindung gab es ein Problem. Es war noch der Standartname für den Tisch im .env File eingetragen. |
| Lösungen | Anpassung des .env Files. |
| Durchgeführte Tests | - |
| Wissensbeschaffung | Laravel Seite, Internet |
| Beanspruchte Hilfeleistung | - |
| Vergleich mit dem Soll-Zeitplan | Weiterhin vor dem Zeitplan. |
| Persönliche Tagesreflexion | Dem Zeitplan zu folgen, scheint bis jetzt problemlos möglich zu sein. Ich habe nur Bedenken, ob ich die Dokumentation ausreichen ausfülle. |

Tabelle 8 - Arbeitsprotokoll Tag 4

Mittwoch, 07. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Modelle erstellen, Migrationen erstellen, Routes erstellen, Usercontroller und API-Authentifizierung erstellen |
| Ungeplante Arbeiten /  geleistete Überstunden | Bilder an die API senden zu können. |
| Erreichte Ziele, Erfolgserlebnisse | Die Authentifizierung mit Sanctum funktioniert. Weiter konnte die Logik für die Meisten Controller-Aktionen erstellt werden. Für alle erstellten Endpunkte sind ausserdem Insomnia-Requests vorhanden. Alle Routes sind definiert. |
| Herausforderungen Probleme | Die Requests in Insomnia hatten die falschen Header. Ich hatte keinen Zugriff auf den aktuellen User über die Sanctum-Middleware. |
| Lösungen | Den Base Request zu einem custom Request mit JSON-Header ändern. Die Sancutm-Middleware funktioniert nur in Routes, in denen sie auch angewendet wird. |
| Durchgeführte Tests | - |
| Wissensbeschaffung | Laravel-Seite und Internet |
| Beanspruchte Hilfeleistung | Ich habe Domenik gefragt, ob eine echte Bilder-Uploadfunktion notwendig ist. Er hat dies bejaht. |
| Vergleich mit dem Soll-Zeitplan | Ich liege gut im Zeitplan. Ich bin ein wenig voraus. |
| Persönliche Tagesreflexion | Es gab zwei grössere Herausforderungen, die ich aber beide lösen konnte. Heute war der erste Tag, an dem ich komplexere Logik schreiben musste. Ich konnte gute Fortschritte machen und bin zufrieden mit meiner Leistung. |

Tabelle 9 - Arbeitsprotokoll Tag 5

Dienstag, 13. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Usercontroller und API-Authentication erstellen, Categories- und Trophiescontroller erstellen |
| Ungeplante Arbeiten /  geleistete Überstunden | - |
| Erreichte Ziele, Erfolgserlebnisse | Die komplette Validierung ist abgeschlossen. |
| Herausforderungen Probleme | Es gibt ein bekanntes Problem in Laravel, das bei PUT Requests keine http Forms verwendet werden können. Diese sind aber nötig, um Bilder uploaden zu können. |
| Lösungen | Um das Problem zu umgehen, musste ich einen POST Request verwenden und PUT als Methoden Parameter mitgeben.  Abbildung 2: Put Methode |
| Durchgeführte Tests | - |
| Wissensbeschaffung | Laravel Seite, Internet, StackOverflow: https://stackoverflow.com/questions/50691938/patch-and-put-request-does-not-working-with-form-data |
| Beanspruchte Hilfeleistung | Keine |
| Vergleich mit dem Soll-Zeitplan | Da ich heute gut vorangekommen bin, habe ich meinen Vorsprung sogar ein Wenig ausgebaut. |
| Persönliche Tagesreflexion | Es ging grössten Teils problemlos voran und wenn ein Problem auftrat, konnte ich es sehr schnell lösen. Heute konnte ich ungewöhnlich schnellen Fortschritt machen. |

Tabelle 10 - Arbeitsprotokoll Tag 6

Mittwoch, 14. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Validierung und Errors erstellen |
| Ungeplante Arbeiten /  geleistete Überstunden | Meine BMA-Gruppe bat mich ungeplant an einer Abgabe zu arbeiten. |
| Erreichte Ziele, Erfolgserlebnisse | Ich konnte trotz ungeplanter Aufgaben im Zeitplan bleiben. |
| Herausforderungen Probleme | - |
| Lösungen | - |
| Durchgeführte Tests | - |
| Wissensbeschaffung | Internet |
| Beanspruchte Hilfeleistung | - |
| Vergleich mit dem Soll-Zeitplan | Auch mit Verzögerungen, bin ich dem Zeitplan voraus. |
| Persönliche Tagesreflexion | Die grosszügige Zeitplanung und das ich gestern viel erreichen konnte hat die heutigen Verzögerungen aufgefangen. |

Tabelle 11 - Arbeitsprotokoll Tag 7

Montag, 19. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Swagger implementieren |
| Ungeplante Arbeiten /  geleistete Überstunden | - |
| Erreichte Ziele, Erfolgserlebnisse | Alle Testfälle sind definiert. |
| Herausforderungen Probleme | Der Base URL für Swagger war noch auf default gestellt. Deshalb haben die End Points zuerst nicht funktioniert. |
| Lösungen | Base URL in .env File anpassen. |
| Durchgeführte Tests | - |
| Wissensbeschaffung | Internet |
| Beanspruchte Hilfeleistung | - |
| Vergleich mit dem Soll-Zeitplan | Es konnte ein kleiner Vorsprung hinzugefügt werden. |
| Persönliche Tagesreflexion | Sehr gut. Ich bin gut vorwärts gekommen und konnte das Problem mit der Swagger Einstellung schnell lösen. |

Tabelle 12 - Arbeitsprotokoll Tag 8

Dienstag, 20. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Swagger implementieren, Authentication für alle Controller erstellen |
| Ungeplante Arbeiten /  geleistete Überstunden | - |
| Erreichte Ziele, Erfolgserlebnisse | Tests ausführen, Fehlerbehebung |
| Herausforderungen Probleme | Es gab ein grösseres Problem bei dem Erstellen und Verändern von Trophäen. Ein User konnte über den Fremdschlüssel der Kategorie in der Trophäen Tabelle, auf die Kategorien eines anderen Benutzers zugreifen. Um das zu verhindern, stellt Laravel Policies zur Verfügung. Diese konnten aber nicht benutzt werden, weil ich Version 4.7 verwende, die Aktuelle wäre 9. |
| Lösungen | Manuelle Überprüfung in den Controllern. Über einen Datenbankzugriff prüfen, welchem User welche Kategorie gehört. |
| Durchgeführte Tests | Users, Types, Colors, Categories, Trophies |
| Wissensbeschaffung | Laravel Seite, Internet |
| Beanspruchte Hilfeleistung | - |
| Vergleich mit dem Soll-Zeitplan | Knapper Vorsprung |
| Persönliche Tagesreflexion | Es war sehr aufwändig die Controller Logik selbst zu schreiben. Durch den Vorsprung im Zeitplan war es aber unproblematisch diese Zeit zu investieren. |

Tabelle 13 - Arbeitsprotokoll Tag 9

Mittwoche, 21. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Testfälle definieren, Tests ausführen, Fehlerbehebung |
| Ungeplante Arbeiten /  geleistete Überstunden | - |
| Erreichte Ziele, Erfolgserlebnisse | Fazit und Reflexion |
| Herausforderungen Probleme | - |
| Lösungen | - |
| Durchgeführte Tests | - |
| Wissensbeschaffung | Domenik Hofer |
| Beanspruchte Hilfeleistung | Ich habe Domenik Hofer nach dem Vorgehen der IPA und der Präsentation gefragt. |
| Vergleich mit dem Soll-Zeitplan | Ist entspricht dem Soll. |
| Persönliche Tagesreflexion | Ich konnte gute Fortschritte in der IPA-Arbeit selbst machen. Ich bin auf einem guten Weg um gemäss Zeitplan abzuschliessen. |

Tabelle 14 - Arbeitsprotokoll Tag 10

Montag, 26. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan |  |
| Ungeplante Arbeiten /  geleistete Überstunden |  |
| Erreichte Ziele, Erfolgserlebnisse |  |
| Herausforderungen Probleme |  |
| Lösungen |  |
| Durchgeführte Tests |  |
| Wissensbeschaffung |  |
| Beanspruchte Hilfeleistung |  |
| Vergleich mit dem Soll-Zeitplan |  |
| Persönliche Tagesreflexion |  |

Tabelle 15 - Arbeitsprotokoll Tag 11

Dienstag, 27. September 2022

|  |  |
| --- | --- |
| Tagesziele gemäss Zeitplan | Fazit und Reflexion |
| Ungeplante Arbeiten /  geleistete Überstunden | - |
| Erreichte Ziele, Erfolgserlebnisse |  |
| Herausforderungen Probleme |  |
| Lösungen |  |
| Durchgeführte Tests |  |
| Wissensbeschaffung |  |
| Beanspruchte Hilfeleistung |  |
| Vergleich mit dem Soll-Zeitplan |  |
| Persönliche Tagesreflexion |  |

Tabelle 16 - Arbeitsprotokoll Tag 12

Teil 2: Projekt

# Kurzfassung

Ausgangssituation

Die TwoFold AG arbeitet generell mit PHP und verwendet dabei häufig das Laravel Framework. Leider war zum Zeitpunkt der IPA kein Kundenprojekt verfügbar, das ein passendes Aussmas aufwies. Daher war es nötig ein typenähnliches Projekt zu eintwickeln.

Umsetzung

Als erstes plante ich den Datenbankaufbau und die Endpoints. Anschliesend konnte ich die Laravel Migrationen erstellen. Dann folten die Controller. Für die Authentivizierung benutzte ich Laravel Sanctum und für das « Frontend » Bzw. das Testing Swagger.

Ergebnis

Alle geforderten API Funktionen sind verfügbar und nur für angemeldete Benutzer zugänglich. Die eingegebenen Daten werden auf Korrektheit der Datentypen in der Datenbank geprüft.

# Informieren

## Ausgangslage

Mit einer API sollen Trophäen und dazu gehörige Kategorien erstellt und bearbeitet werden können. Die API soll das Registrieren von Benutzern erlauben. Nur registrierte Benutzer sollen Zugang zu den Endpoints der API haben.

## Abklärungen

Es ist vorgesehen, dass jeder Benutzer nur Zugang zu seinen eigenen Trophäen und Kategorien hat. Das ist ungewöhnlich, wurde aber auf Nachfrage bestätigt.

## Verstandene Aufgabenstellung und Ziel der Arbeit

Es muss ein sinnvolles Datenbankschema erstellt werden. Routes für die Funktionen der API und die dazugehörende Controller-Logik.

## Verfeinerung des Auftrages

Authentifizierung von Usern muss gewährleistet werden. Eine Userschnittstelle für Tests und Präsentationszwecke muss vorhanden sein.

## Projektumfeld und Systemgrenzen

Die API läuft auf einem XAMPP Server und soll Lokal auf demselben Computer verfügbar sein. Der Zugriff erfolgt über den Browser via Localhost.

# Planen

## Verwendete Projektmanagementmethode

I – Informieren

P – Planen

E – Entscheiden

R – Realisieren

K – Kontrollieren

A - Auswerten

## Versionierung und Datensicherheit

GiT Hub Repository

## Priorisierung der Tätigkeiten

|  |  |
| --- | --- |
| Beschreibung der Tätigkeit | Priorität |
| Datenbank planen | Sehr hoch |
| Endpoints planen | Sehr hoch |
| Usecases beschreiben | Normal |
| Authentifizierung sicherstellen | Hoch |
| Swagger implementieren | Hoch |
| Tests durchführen | Hoch |
| Dokumentation erstellen | Normal |

Tabelle 17 - Tätigkeitenliste

## Kernfeature ABC

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

### Use-Case für Abc

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

### Klassendiagramm

nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

### GUI-Design / Mockup

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

### Konzeptioneller Aufbau

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

## Kernfeature ZYZ

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

### Use-Case für zyz

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

### Klassendiagramm

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

### GUI-Design / Mockup

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

### Konzeptioneller Aufbau

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

## Geplantes Vorgehen für die Qualitätssicherung

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

## Anmerkungen zum Zeitplan

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

## Begründung für Abweichungen zum Zeitplan

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

# Entscheiden

## Datenbankfeld für Farbe

Kategorien sollen eine Farbe haben. Da nur das Backend erstellt wird, ist nicht ganz klar, wie dieser Wert im Frontend angewendet werden soll. Es ist aber davon auszugehen, dass mit diesem Wert entschieden werden soll, was für eine Farbe gerendert wird.

## Beschreibung Variante mit frei wählbarem Text

Frei wählbarer Text erlaubt für die grösste flexibilität. Es bringt aber auch das Risiko von Datenredundanz und Schreibfehlern. Das grösste Risiko ist aber, dass die Daten keine klare Bedeutung in einer Frontend-Applikation haben.

## Beschreibung Variante mit vorgegebener Auswahl an Farben

Vorbestimmte Farben schränken ganz klar die Freiheit eines User sein. Mit dieser Variante ist aber die Datenbank-Normalisierung sicher gestellt. Durch die Standartisierung wird auch die Erweiterbarkeit und die Kommunikation zwischen Front- und Backend sicher gestellt.

## Kurzbeschreibung der Entscheidungskriterien

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Kriterium | Gewichtung | Optionen | | | |
|  |  | Frei wählen | | Vorgeben | |
|  |  | Wert | Gewichtet | Wert | Gewichtet |
| Flexibilität | 10% | 10 | 100 | 2 | 20 |
| Datenredundanz | 20% | 5 | 100 | 10 | 200 |
| Erweiterbarkeit | 20% | 3 | 60 | 7 | 140 |
| Frontend Kompatibilität | 50% | 2 | 100 | 10 | 500 |
| Total |  |  | 360 |  | 860 |

Um den Farbwert sinnvoll benutzbar zu machen, muss er in irgendeiner Weise eingeschränkt werden. In diesem Kontext bei einer Auswahl, die von mir bestimmt wird.

## Datenbank Redundanzen

Die Felder für Ort, Gegner und Name des Clubs des Trophäentisches könnten sowohl mehrfach für verschieden Trophäen vorkommen als auch falsch geschrieben werden. Es ist aber in den Vorgaben nicht vorgesehen, dass diese separat definiert werden sollen. Durch separate Tabellen für diese Felder könnte Redundanz verhindert werden. Das Problem von fehlender Normalisierung, kann aber ohne eine Auswahl zu präsentieren nicht verhindert werden.

## Beschreibung Variante mit zusätzlichen Tabellen

Für jedes Feld, Ort, Gegner und Name des Clubs wird eine separate Tabelle erstellt. Wird eine Trophäe erstellt oder geändert, wird geprüft ob der Wert des Feldes bereits existiert, wenn ja wird die existierende ID verwendet, wenn nein wird ein neuer Eintrag erstellt. Dadurch erhöt sich aber die Logik in den Controllern erhablich

## Beschreibung Variante ohne zusätlzichen Tabellen

Einträge werden als Text direkt in den Trophäen gespeichert. Doppelte Einträge sind so möglich. Auch falschgeschriebene Einträge sind möglich. Es reduziert sich aber deutlich die Controller-Logik.

## Kurzbeschreibung der Entscheidungskriterien

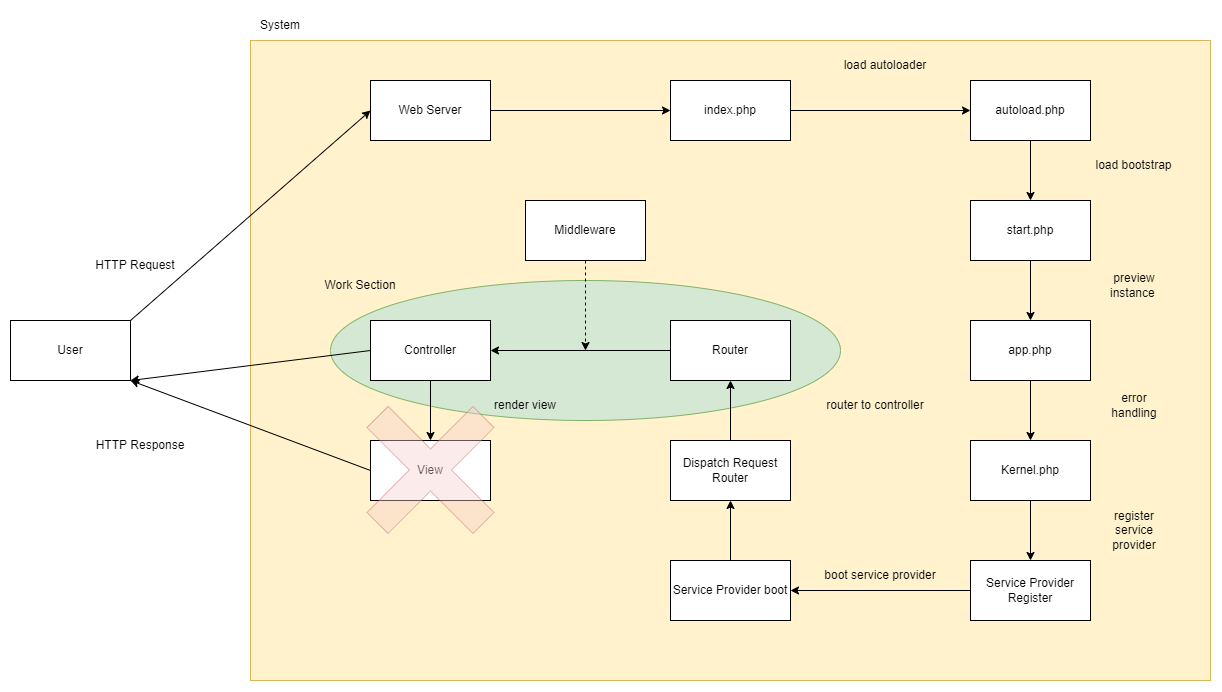
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Kriterium | Gewichtung | Optionen | | | |
|  |  | Mit Tabellen | | Ohne Tabellen | |
|  |  | Wert | Gewichtet | Wert | Gewichtet |
| Nutzen laut Vorgabe | 40% | 10 | 400 | 10 | 400 |
| Datenredundanz | 20% | 9 | 180 | 3 | 60 |
| Datennormalisierung | 20% | 3 | 60 | 3 | 60 |
| Aufwand | 20% | 3 | 60 | 9 | 180 |
| Total |  |  | 700 |  | 700 |

Obwohl die Verhinderung von Redundanz eleganter wäre, rechtfertigt es nicht den Mehraufwand. Die Funktionalität wird durch das Weglassen der Tabellen nicht beeinflusst, weil keine der Filter und Sortierungen auf diese Felder angewendet werden sollen.

# Realisieren

## Abbildung des Gesamtsystems

Laravel ist ein umfangreiches PHP-Framework. Im Rahmen dieses Projektes wird nur ein kleiner Teil seiner Möglichkeiten ausgeschöpft. In Grün ist der Bereich der in dem Eigenleistung enthalten ist. Da es sich um eine API handelt, werden keine Views erstellt werden.



## Interaktionen zwischen den Teilsystemen

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

## Vorgehensweise

Zuerst installierte ich Laravel und alle benötigten Pakete. Da ich das PHP-Packet Management System Composer bereits installierst habe, konnte ich ein neues Laravel-Projekt damit erstellen.



Für das Login wir das Sanctum-Packet von Laravel verwendet. Dies ermöglicht das einfache Erstellen von API-Tokkens und User-Management.





Zur API-Dokumentation wird Swagger verwendet. Das für Laravel angepasste Swagger-Packet ist auf Git verfügbar.



## Grundaufbau / Architetkur des Projektes

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

## Anbindung an die Datenbank

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

## Implementierung des Kernfeatures JSON Funktionalität

|  |  |
| --- | --- |
| Original public/index.php | Modifiziert public/index.php |
|  |  |
| Neue Datei Requests/BaseRequest.php | |
|  | |

### Funktionsweise des Kernfeatures JSON Funktionalität

Durch diese Änderung erwarted der HTTP header JSON anstatt einer statischen Webseite. Dazu ersetzt man den Laravel eigenen Request, durch einen selbst geschriebenen.

### Ziel des Kernfeatures JSON Funktionalität

In seiner normalen Funktionsweise, gibt Laravel statische Seiten zurück. Diese sind üblicherweise in Blade geschrieben und werden über die web-Routes abgewickelt. Da dieses Projekt aber eine JSON-API ist kann diese Annahme gelegentlich zu Problemen führen.

### Effektives Produkt

Anstatt das bei einem Authentifizierungsfehler eine statische Webseite und ein Request Header für eine Webseite geschickt wird, wird nun eine JSON-Nachricht mit einem passenden Header geschickt.

|  |  |
| --- | --- |
| Antwort für die Route api/user | |
| Vorher | Nachher |
|  |  |
|  |  |

## Implementierung des Kernfeatures Request Validierung

Die Requests die an die Datenbank gestellt werden müssen Validiert werden und bei Diskrepanzen müssen gut verständliche Error Messages generiert werden. Zwar ist durch die Verwendung von Laravel Migrations die Datenbank gegen falsche Eingaben abgesichert, es werden aber ohne Validierung nur direkte Datenbankerrors zurück gegeben.

### Funktionsweise des Kernfeatures Request Validierung

Die Kontrollerklasse verfügt über eine validate Methode. Mit dieser kann über eine Liste von Attributen geprüft werden, ob die entsprechenden Voraussetzungen im Request oder der Datenbank erfüllt sind.

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

### Ziel des Kernfeatures Request Validierung

Es werden JSON Fehlermeldungen erzeugt, die dem Benutzer den genauen Fehler mitteilen. So ist klar welcher Parameter fehlt oder falsch eingetragen wurde.

### Effektives GUI / Produkt

Ein Bild, das Text enthält.

Automatisch generierte BeschreibungValidierung um eine neue Trophäe zu erstellen :

Fehlerausgabe :

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

## Implementierung des Kernfeatures Swagger Dokumentation

Um benutzerfreundlichen Zugriff auf die API zu haben, soll Swagger als Frontend benutzt werden.

### Funktionsweise des Kernfeatures Request Validierung

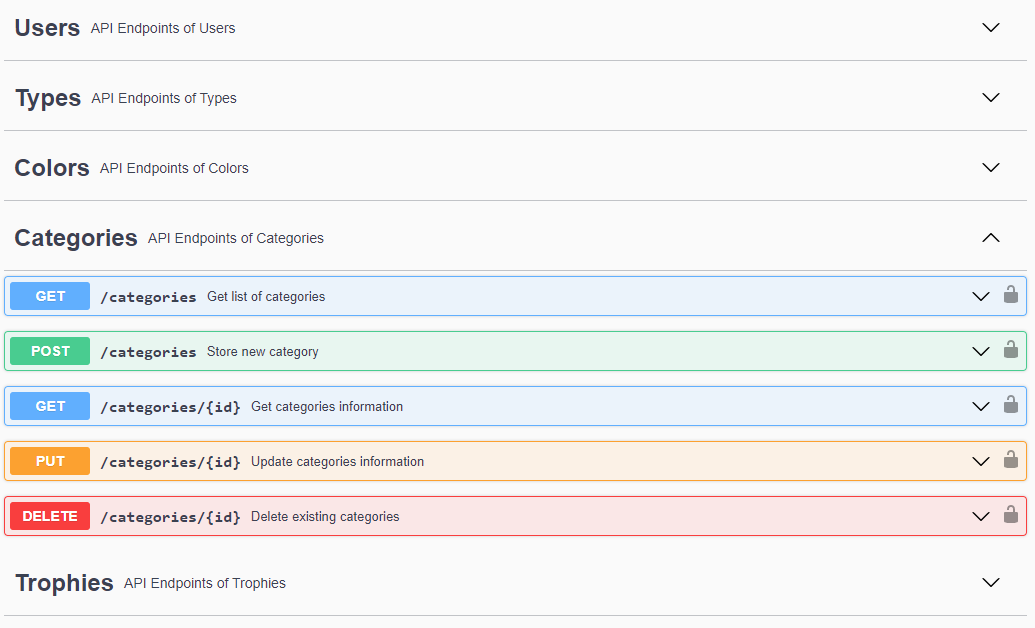
Swagger wird als Composer Packet zum Projekt hinzugefügt. Es liest dann Swagger spezifische Syntax über den Funktionen in den Controllerklassen und erstellt daraus ein GUI, das dann im Browser aufgerufen werden kann.



### Ziel des Kernfeatures Request Validierung

Die API Funktionen sollen klar ersichtlich sein und einfach ausprobiert werden können. So ist dokumentiert was mit der API möglich ist und die Routes können getestet werden.

### Effektives GUI / Produkt



# Kontrollieren

## Beschreibung der Randbedingungen / Testanlage (Umfeld)

Durch das PHP-Framework Laravel, werden Endpunkte für Benutzer, Kategorien und Trophäen bereit gestellt. Über die Implementierung von Swagger, werden diese Endpunkte dokumentiert, können aber auch händisch getestet werden.

## Eingesetzte Testmittel und -methoden

Es wird ein Testprotokoll erstellt, das dann über das Swagger-UI ausgeführt und dokumentiert wird.

## Beschreibung der Testszenarien

Ein User muss sich Registrieren und anmelden können. Dazu müssen Bearer-Tokens erstellt und zurückgegeben werden.

Ein User mit gültigem Bearer-Token muss nur seine Eigenen Kategorien Ansehen, Erstellen, Bearbeiten und Löschen können.

Ein User mit gültigem Bearer-Token muss nur seine eigenen Trophäen Ansehen, Erstellen, Bearbeiten, Filtern und Löschen können.

Jeder kann sich die möglichen Farben anzeigen lassen.

Jeder kann sich die möglichen Typen anzeigen lassen.

## Testprotokoll zum Testszenario User

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Testfall | Vorbedingung | Vorgehen, Eingaben | Erwartetes Resultat | Effektives Resultat | i.O. |
| 1 | Registrierung | Laravel-Server und Swagger müssen aktiviert sein. | Eine einmalige E-Mail-Adresse, ein Name und ein Passwort müssen eingegeben werden. | Es wird ein User in der Datenbank erstellt. Ein Bearer-Token wird erstellt und zurückgegeben. | Token erhalten.  User erstellt. | OK |
| 2 | Anmeldung | Testfall 1 | E-Mail-Adresse und Passwort werden eingegeben. | Bearer-Token wird erstellt und zurückgegeben. | Token erhalten. | OK |

Tabelle 18 - Testprotokoll für das User Feature

## Testprotokoll zum Testszenario Farben

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Testfall | Vorbedingung | Vorgehen, Eingaben | Erwartetes Resultat | Effektives Resultat | i.O. |
| 1 | Farben anzeigen | Laravel-Server und Swagger müssen aktiviert sein. | URL für Farben-End Point senden. | Liste der möglichen Farben wird zurückgegeben | Typen angezeigt. | OK |

Tabelle 19 - Testprotokoll für das Farben Feature

## Testprotokoll zum Testszenario Typen

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Testfall | Vorbedingung | Vorgehen, Eingaben | Erwartetes Resultat | Effektives Resultat | i.O. |
| 1 | Typen anzeigen | Laravel-Server und Swagger müssen aktiviert sein. | URL für Typen-End Point senden. | Liste der möglichen Typen wird zurückgegeben | Farben angezeigt. | OK |

Tabelle 19 - Testprotokoll für das Typen Feature

## Testprotokoll zum Testszenario Kategorien

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Testfall | Vorbedingung  (Für alle Endpunkte gilt:  Laravel-Server und Swagger müssen aktiviert sein.  Gültiger Bearer-Token muss mit gesendet werden.) | Vorgehen, Eingaben | Erwartetes Resultat | Effektives Resultat | i.O. |
| 1 | Vorhandene Kategorien des Benutzers werden anzeigen. | Kategorien müssen vorhanden sein. | URL für Kategorien-End Point senden. | Liste der vorhandenen Kategorien wird angezeigt. | Es werden alle Kategorien dieses Users angezeigt. | OK |
| 2 | Neue Kategorie erstellen. | - | Gültiger Name muss eingegeben werden.  Vorhandene Farb-ID muss eingegeben werden. | Kategorie in der Datenbank wird erstellt. | «Success» wird zurückgegeben und der Eintrag wird erstellt. | OK |
| 3 | Einzelne Kategorie anzeigen | Kategorie existiert in der Datenbank. | Gültige Kategorie-ID wird eingegeben. | Die Kategorie mit der entsprechenden ID wird angezeigt. | Die Kategorie mit der entsprechenden ID wird gezeigt. | OK |
| 4 | Kategorie bearbeiten | Kategorie existiert in der Datenbank. | Gültiger Name kann eingegeben werden.  Vorhandene Farb-ID kann geändert werden. | Der Eintrag der Kategorie in der Datenbank, wird mit den neuen Daten angepasst. | «Success» wird angezeigt und der Eintrag in der Datenbank wurde geändert. | OK |
| 5 | Kategorie löschen | Kategorie existiert in der Datenbank. | Gültige Kategorie-ID wird eingegeben. | Der Eintrag der Kategorie in der Datenbank wird entfernt. | Kategorie mit der entsprechenden ID wurde gelöscht. | OK |

Tabelle 19 - Testprotokoll für das Kategorien Feature

## Testprotokoll zum Testszenario Trophäen

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Testfall | Vorbedingung  (Für alle Endpunkte gilt:  Laravel-Server und Swagger müssen aktiviert sein.  Gültiger Bearer-Token muss mit gesendet werden.) | Vorgehen, Eingaben | Erwartetes Resultat | Effektives Resultat | i.O. |
| 1 | Vorhandene Trophäen des Benutzers werden angezeigt. | Es existieren Trophäen in der Datenbank. | URL für den Trophäen-Endpunkt eingeben. | Liste der vorhandenen Trophäen wird angezeigt. | Die Trophäen werden angezeigt. | OK |
| 2 | Neue Trophäe erstellen. | - | Existierende Typen-ID wird eingegeben.  Gültiger Titel wird eingegeben.  Gültiges Ranking wird eingegeben.  Gültiges Datum wird eingegeben.  Existierende Kategorie-ID wird eingegeben.  Gültiger Ort wird eingegeben.  Gültiger Gegner wird eingegeben.  Gültiger Punktestand wird eingegeben.  Gültiges Preisgeld wird eingegeben.  Gültiger Klub Name wird eingegeben.  Gültiges Bild wird eingegeben. | Eine neue Trophäe wird in der Datenbank erstellt. | Success wird zurückgegeben und ein Eintrag in der Datenbank wird erstellt. | OK |
| 3 | Einzelne Trophäe anzeigen. | Trophäe existiert in der Datenbank. | Existierende Trophäen-ID wird eingegeben. | Die Trophäe mit der entsprechenden ID wird angezeigt. | Trophäe wird angezeigt. | OK |
| 4 | Trophäe bearbeiten. | Trophäe existiert in der Datenbank. | Existierende Typen-ID kann eingegeben werden.  Gültiger Titel kann eingegeben werden.  Gültiges Ranking kann eingegeben werden.  Gültiges Datum kann eingegeben werden.  Existierende Kategorie-ID kann eingegeben werden.  Gültiger Ort kann eingegeben werden.  Gültiger Gegner kann eingegeben werden.  Gültiger Punktestand kann eingegeben werden.  Gültiges Preisgeld kann eingegeben werden.  Gültiger Klub Name kann eingegeben werden.  Gültiges Bild kann eingegeben werden. | Der Eintrag der Trophäe in der Datenbank, wird mit den neuen Daten angepasst. | Success wird angezeigt und der Eintrag wurde angepasst. Das Verwenden von fremd Kategorien wird nicht erlaubt. | OK |
| 5 | Trophäe löschen | Trophäe existiert in der Datenbank. | Gültige Trophäen-ID wird eingegeben. | Trophäe mit der entsprechenden ID wird entfernt. | Trophäe wurde aus der Datenbank entfernt. | OK |
| 6 | Trophäen nach Datum sortieren. | Es sind Trophäen in der Datenbank vorhanden. | URL für das Sortieren nach Datum wird eingegeben. | Trophäen werden nach Datum sortiert angezeigt. | Trophäen werden nach Datum sortiert angezeigt. | OK |
| 7 | Trophäen nach Platzierung sortieren. | Es sind Trophäen in der Datenbank vorhanden. | URL für das Sortieren nach Platzierung wird eingegeben. | Trophäen werden nach Platzierung sortiert angezeigt. | Trophäen werden nach Platzierung sortiert angezeigt. | OK |
| 8 | Trophäen nach Typ filtern. | Es sind Trophäen in der Datenbank vorhanden. | Gültige Typen-ID eingeben. | Trophäen des entsprechenden Typs werden angezeigt. | Trophäen des entsprechenden Typs werden angezeigt. | OK |
| 9 | Trophäen nach Platzierung filtern. | Es sind Trophäen in der Datenbank vorhanden. | Gültige Platzierung eingeben. | Trophäen mit der entsprechenden Platzierung werden angezeigt. | Trophäen mit der entsprechenden Platzierung werden angezeigt. | OK |
| 10 | Trophäen nach Kategorie filtern. | Es sind Trophäen in der Datenbank vorhanden. | Gültige Kategorie-ID eingeben. | Trophäen mit der entsprechenden Kategorie werden angezeigt. | Trophäen mit der entsprechenden Kategorie werden angezeigt. | OK |

Tabelle 19 - Testprotokoll für das Trophäen Feature

## Bugfixing

Es war möglich für Benutzer auf die Kategorien und Trophäen von anderen Benutzern zuzugreifen. Das wollte ich mit “Policies” lösen. Leider hat das nicht funktioniert, weil meine Laravel Instalation nicht auf dem neusten Stand ist. Daraufhin habe ich die Logik um das zu verhindern in den Controllern geschrieben.

## Testprotokoll nach dem Bugfixing

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Testfall | Vorbedingung | Vorgehen, Eingaben | Erwartetes Resultat | Effektives Resultat | i.O. |
| 1 | Einzelne Kategorie eines anderen Benutzers anzeigen. | Registrierter Benutzer sendet Bearer Token mit.  Es existieren Kategorien eines anderen Benutzers in der Datenbank. | Kategorien ID eines anderen Benutzers eingeben. | Forbidden und ein 403 Error Code wird zurückgegeben. | Forbidden und Error Code. | OK |
| 2 | Einzelne Trophäe eines anderen Benutzers anzeigen. | Registrierter Benutzer sendet Bearer Token mit.  Es existieren Trophäen eines anderen Benutzers in der Datenbank. | Trophäen ID eines anderen Benutzers eingeben. | Forbidden und ein 403 Error Code wird zurückgegeben. | Forbidden und Error Code. | OK |

Tabelle 20 - Testprotokoll nach Bugfixing

**Upload / Live-Schaltung des Produktes**

Das Produkt soll ausschliesslich lokal in der Testumgebung ausgeführt werden.

## Einführung beim Kunden

Es ist kein Kunde oder Endbenutzer vorgesehen.

# Auswerten

## Reflexion der Vorgehensweise

**Informieren**

Durch vorsichtiges Lesen der Vorgaben und sorgfältiges abstimmen der Mittel, die am besten zur Umsetzung geeignet sind, konnte ich mir schon vor dem Beginnen ein klares Bild machen. Dieses Vorgehen hat mir viel Zeit gespart und ermöglichte es mir, bequem im Zeitplan zu bleiben.

**Planen**

In der Planungsphase gab es einige Unklarheiten im Bezug auf die Funktionalität. Ich bekam von Domenik Hofer Antworten auf meine Fragen, die es mir erlaubte die Datenbank korrekt zu planen. Durch das erstellen von den entsprechenden Diagrammen wurden potentielle Fehlerquellen ausgeschlossen.

**Entscheiden**

Durch intensives Auseinandersetzen der Struktur meiner Applikation, konnte ich mich für Ressourcen und Vorgehensweisen innerhalb von Laravel entscheiden. Wie das Verwenden von Model Relationships um meine Controller Logik zu vereinfachen.

**Realisieren**

Die Realisierung lief dank der sorgfältigen Planung grössten Teils reibungslos ab. Das einzige grösser Problem das auftauchte, war mit dem Erstellen und Bearbeiten von Trophäen. Um die Fremdschlüssel von Kategorien korrekt zu prüfen, hätte ich Laravel Policies für eine elegante Lösung benötigt. Diese waren aber in meiner Laravel Version (4.7) nicht verfügbar (benötigt 9). Es war also nötig selber Controller Logik zu schreiben. Das war mehr Aufwand und weniger elegant.

**Kontrollieren**

Das Testen beschränkte sich auf das korrekte Funktionieren der API Routes. Da das von Anfangan mit in Betracht gezogen wurde, konnte das effizient mit Swagger ausgeführt werden.

## Bewertung des Produktes

Das Produkt erfüllt alle Forgaben. Dank der Swagger Implementation ist es auch sehr sauber Dokumentiert und Benutzerfreundlich. Die Erweiterbarkeit ist durch das einhalten von Laravel Standards gewährleistet. Einzig das Fehlen von Policy Logik macht die Controller etwas unübersichtlicher und die Erweiterbarkeit aufwändiger.

## Persönliches Schlusswort und Bilanz

Ich bin sehr zufrieden mit dem Endergebniss. Alles funktioniert tadellos Ich konnte mein Laravel Wissen vertiefen und Laravel Praktiken anwenden. Ich nehme vorallem positiv mit, wie die gute und ausführliche Planung das Arbeiten vereinfacht hat.

# Glossar

A

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

B

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

C

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

D

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

E

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

F

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

G

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

H

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

I

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

J

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

K

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

L

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

M

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

N

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

O

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

P

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

Q

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

R

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

S

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

T

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

U

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

V

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

W

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

X

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

Y

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

Z

|  |  |
| --- | --- |
| Begriff / Abkürzung | Erklärung |
|  |  |

# Quellenverzeichnis

Literaturverzeichnis

Literatur, die man für die eigene Arbeit benützt hat, muss an **2 Orten** verzeichnet werden:

**1. im Quellenverzeichnis am Schluss der Arbeit,**

**2. im laufenden Text, wo sie wörtlich oder sinngemäss zitiert wird.**

**Eintrag im Quellenverzeichnis**

Jede Arbeit weist am Ende des Textes ein Literaturverzeichnis auf, in dem **alle zitierte Literatur zum Überblick** gesammelt wird. Das Literaturverzeichnis am Schluss der Arbeit ist alphabetisch nach Nachnamen der Autor/innen bzw. Herausgeber/innen geordnet.

Neben dem Autorennnamen müssen mindestens enthalten sein: Der vollständige Titel und Untertitel, der Erscheinungsort und das Erscheinungsjahr.

In das Quellenverzeichnis gehören auch Name und Adresse der Institution, in der die interviewten Personen arbeiten.

**Beispiel für einen Eintrag eines verwendeten Buchs im Quellenverzeichnis**

Grossenbacher, René: Die Medienmacher. Eine empirische Untersuchung zur Beziehung zwischen

Public Relations und Medien in der Schweiz. Solothurn 1986

Internetquellen

Auch Zitate aus dem Internet müssen an **2 Orten** verzeichnet werden: sowohl im Literaturverzeichnis am

Schluss der Arbeit als auch im laufenden Text.

1. Vollständiger Name des Verfassers des Internet-Dokuments; *falls dieser fehlt:*

Titelzeile der Hypertextseite oder Name der Organisation/Institution, die für die Website verantwortlich ist.

2. Titel, gegebenenfalls Untertitel des Internet-Dokuments; falls diese Angaben fehlen:

Beschreibung der Website („Persönliche Homepage“, „Website der Firma“ usw.).

3. Ev. Verweis auf die Online-Zeitschrift, in der das Dokument publiziert wurde, oder weitere Angaben zur Website.

4. Erstellungsdatum oder Datum der letzten Änderung; falls diese Angaben fehlen: „o. J.“.

5. Vollständige URL (Internetadresse).

**Beispiele:**

• Beck, Daniel: Sportübertragung am Fernsehen. Vom Publikumsmagneten zum teuren Verlustgeschäft?

In: Medienheft, 6.9.2016. URL: <http://www.medienheft.ch/kritik/bibliothek/k16_BeckDaniel.html> (22.10.2014).

• Payer, Margarete: Wir katalogisieren das Internet. URL’s, URN’s und Co. Vortrag auf der 1. InetBib-Tagung, Dortmund. 21.4.2014. URL: http://users.aol.com/margpayer/urlso.html (7.11.2016).

• Pellegrino, Joseph: Persönliche Homepage. 24.9.2016. URL:

http://www.english.eku.edu/pellegri/personal.htm (7.11.2016). • SRG SSR Idée Suisse: Die Trägerschaft. Ein Verband von Vereinen. O. J. URL: http://www.srgssrideesuisse.ch/de/corporation/welcom.html (20.3.2016).

Abbildungsverzeichnis

[Abbildung 1: Zeitplan 12](#_Toc115164732)

Tabellenverzeichnis

[Tabelle 1 - Adressinformation Auszubildender 7](#_Toc529451342)

[Tabelle 2 – Ausführungszeitraum 7](#_Toc529451343)

[Tabelle 3 – Termine 7](#_Toc529451344)

[Tabelle 4 - Involvierte Personen 8](#_Toc529451345)

[Tabelle 5 - Arbeitsprotokoll Tag 1 11](#_Toc529451346)

[Tabelle 5 - Arbeitsprotokoll Tag 2 12](#_Toc529451347)

[Tabelle 5 - Arbeitsprotokoll Tag 3 13](#_Toc529451348)

[Tabelle 5 - Arbeitsprotokoll Tag 4 14](#_Toc529451349)

[Tabelle 5 - Arbeitsprotokoll Tag 5 15](#_Toc529451350)

[Tabelle 5 - Arbeitsprotokoll Tag 6 16](#_Toc529451351)

[Tabelle 5 - Arbeitsprotokoll Tag 7 17](#_Toc529451352)

[Tabelle 5 - Arbeitsprotokoll Tag 8 18](#_Toc529451353)

[Tabelle 5 - Arbeitsprotokoll Tag 9 19](#_Toc529451354)

[Tabelle 5 - Arbeitsprotokoll Tag 10 20](#_Toc529451355)

[Tabelle 5 - Arbeitsprotokoll Tag 11 21](#_Toc529451356)

[Tabelle 5 - Arbeitsprotokoll Tag 12 22](#_Toc529451357)

# Anhang

[Projektjournal 42](#_Toc529454774)

[Gesprächsprotokoll vom xx. Monat yyyy 42](#_Toc529454775)

[Erster Expertenbesuch 44](#_Toc529454776)

[Zweiter Expertenbesuch 44](#_Toc529454777)

[Code 45](#_Toc529454778)

[Filename.endung 45](#_Toc529454779)

[Filename.endung 45](#_Toc529454780)

[Zusätzliche Manuals, Skripts und weiteres 46](#_Toc529454781)

[Handbuch A 46](#_Toc529454782)

[Skript B 46](#_Toc529454783)

# Projektjournal

Gesprächsprotokoll vom xx. Monat yyyy

Teilnehmer: <Vorname Name>, <Vorname Name>

Datum, Zeit, Ort: <Date>, <Time>, <Place>

**Besprechungsnotizen**:

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

**Getroffene Entscheidungen:**

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

Erster Expertenbesuch

Teilnehmer: <Vorname Name>, <Vorname Name>

Datum, Zeit, Ort: <Date>, <Time>, <Place>

**Besprechungsnotizen**:

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

**Getroffene Entscheidungen:**

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

Zweiter Expertenbesuch

Teilnehmer: <Vorname Name>, <Vorname Name>

Datum, Zeit, Ort: <Date>, <Time>, <Place>

**Besprechungsnotizen**:

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

**Getroffene Entscheidungen:**

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.

Code

Filename.endung

<Code>

Filename.endung

<Code>

Zusätzliche Manuals, Skripts und weiteres

Handbuch A

<Code>

Skript B

<Code>

1. Aufgabenstellung Original gemäss Eingabe aus PkOrg [↑](#footnote-ref-1)
2. Aufgabenstellung Original gemäss Eingabe aus PkOrg [↑](#footnote-ref-2)
3. Aufgabenstellung Original gemäss Eingabe aus PkOrg [↑](#footnote-ref-3)
4. Aufgabenstellung Original gemäss Eingabe aus PkOrg [↑](#footnote-ref-4)
5. Aufgabenstellung Original gemäss Eingabe aus PkOrg [↑](#footnote-ref-5)
6. Aufgabenstellung Original gemäss Eingabe aus PkOrg [↑](#footnote-ref-6)
7. Aufgabenstellung Original gemäss Eingabe aus PkOrg [↑](#footnote-ref-7)