Deckblatt

Inhaltsverzeichnis

[1. Einleitung 1](#_Toc176802718)

[1.1. Anforderungen 1](#_Toc176802719)

[1.2. Ziele und Aufbau der Arbeit 1](#_Toc176802720)

[2. Grundlagen 2](#_Toc176802721)

[3. Konzept 2](#_Toc176802722)

[3.1. Strategieentwicklung 2](#_Toc176802723)

[3.2. Design-Entwürfe 5](#_Toc176802724)

[3.3. Technische Entwürfe 8](#_Toc176802725)

[4. Umsetzung 12](#_Toc176802726)

[4.1. Systemvorbereitung 12](#_Toc176802727)

[4.2. Datenbank 12](#_Toc176802728)

[4.3. Front-End 12](#_Toc176802729)

[4.4. Back-End 12](#_Toc176802730)

[5. Systemtests 12](#_Toc176802731)

[5.1. Testfall 1 12](#_Toc176802732)

[5.2. Testfall 2 12](#_Toc176802733)

[5.3. Testfall 3 12](#_Toc176802734)

[5.4. Testfall 4 12](#_Toc176802735)

[6. Zusammenfassung 13](#_Toc176802736)

[6.1. Kritische Reflexion 13](#_Toc176802737)

[I. Quellenverzeichnis X](#_Toc176802738)

[II. Dateianhänge XI](#_Toc176802739)

Abbildungsverzeichnis

[Abbildung 1: Arbeitsprozess mit Git 2](#_Toc176797303)

[Abbildung 2: Wasserfallmodell 3](#_Toc176797304)

[Abbildung 3: Prozesslandkarte 4](#_Toc176797305)

[Abbildung 4: Wireframe\_Admin\_Lading\_Page 6](#_Toc176526479)

[Abbildung 5: Wireframe\_User\_Landing\_Page 6](#_Toc176526479)

[Abbildung 6: Wireframe\_Login\_Page 6](#_Toc176797307)

[Abbildung 7: Enity Relationship Modell 8](#_Toc176797308)

[Abbildung 8: Use Case 10](#_Toc176797309)

# Einleitung

# Anforderungen

Die allgemeinen und speziellen Anforderungen dieses Projekt sind aus dem Dokument **WEB42 – Webprojekt I Labor 2024** zu entnehmen. Die speziellen Anforderungen ergeben sich aus der Wahl des Projektes und der Anzahl der Teilnehmer dieses Projekts.  
Die speziellen Projektanforderung in diesem Fall sind:

*„2.1 Einzel-/Gruppenaufgabe: Mediendatenbank mit Web-Frontend Ausgangssituation: Es soll ein Webfrontend mit Zugriff auf eine Medien-Datenbank entwickelt werden. Die Anzahl der Medienarten, die in der Datenbank gehalten werden, sowie die konkrete Funktionalität ist abhängig von folgenden Gruppengrößen:*

*…*

*Gruppenaufgabe drei Teilnehmer: Medienarten »Foto«, »Hörbücher«, »eBooks«, »Video«; ein Administrator, mind. drei Anwender; detailliertes Vorgehensmodell, inkl. Aufgabenzuweisung Teilnehmer (selbständiges [Unter-]Kapitel, mind. zwei Seiten); detailliertes ER-Diagramm (selbständiges [Unter-]Kapitel, mind. zwei Seiten); vier dokumentierte Testfälle; Ausgabe Liste aller Medien in DB (mind. Name und Art des Medienelements; Bildschirmanzeige ausreichend) Folgende grundlegende Funktionen bietet die Webanwendung:*

* *Datenbankanbindung (Erstellen/Bearbeiten/Löschen)*
* *Accountverwaltung (über Webfrontend; Passwortschutz frei wählbar)*
* *Upload der Medien erfolgt über das Webfrontend; Mehrfachauswahl ist möglich*
* *Verschlagwortung: verwendete Zeichenketten nach Ersterfassung als Drop--Down-Menü (inkl. nachträgliches Bearbeiten oder Löschen)*
* *kombinierbare Suchfunktion über die Menge aller Objekte und deren Attribute*
* *Bildergalerie (u. a. auch über Suchfunktion generiert)*
* *Berücksichtigung anerkannter aktueller Styleguides für das Webfrontend“*

*(Schumacher, 2024, S. 4)*

# Ziele und Aufbau der Arbeit

Die Projektziele sind eine funktionstüchtige Mediendatenbank mit Web-Frontend zu erstellen, die den gestellten Anforderungen entspricht. Darüber hinaus hat diese den Anspruch zusätzliche, sinnvolle Anforderungen zu erfüllen, die sich u.a. aus der Semantik der Variablen ergeben. Diese sind im Kapitel „4.5. Testfälle“ zu finden. Um ein strukturiertes und vorzeigbares Ergebnis zu erzielen, darf natürlich ein ausgeklügeltes Konzept (siehe Kapitel 3) nicht fehlen. In diesem finden sich, außer Projektmanagement-methoden, noch Entwürfe, die z.B. Hilfestellung bei der Strukturierung der Daten oder auch beim Design geben sollen, denn auch wenn man als Entwickler einige kreative Freiheiten hat, sollte man sich an gewisse Grundsätze halten, um dem menschlichen Auge nichts Unzumutbares vorzusetzen. Die ebenfalls notwendige Dokumentation zu einem Projekt stellt dieser Laborbericht dar.

# Grundlagen

**Git**Wir erwähnen im weiteren Text, dass wir Git als Tool zur Kollaboration genutzt haben, deshalb möchten wir es an dieser Stelle kurz erläutern. Git ist ein Versionskontrollsystem, das hauptsächlich zur Verwaltung von Programmcode in Software-projekten verwendet wird. Git ermöglicht es mehreren Ent-wicklern gleichzeitig an einem Projekt zu arbeiten, ohne sich dabei gegenseitig in die Quere zu kommen.

Abbildung 1: Arbeitsprozess mit Git

# Konzept

Bevor die Umsetzung des Projekts beginnt, haben wir unsere Strategie festgelegt, den Ablauf geplant und diverse Entwürfe erstellt und somit die Grundsteine für das Projekt gelegt.

# Strategieentwicklung

Wir haben uns zur Beschreibung unserer Prozessphasen für das Wasserfallmodell (siehe Abbildung 1 auf Seite 3) entschieden. Da es sich bei unserem Projekt, um ein eher kleines Projekt handelt, mit einem klar definierten Endziel, bei dem sich die Anforderungen und Abläufe bereits in der Planungsphase genau beschreiben lassen und sich nachträglich kaum ändern, haben wir es einem Agilen-Ansatz vorgezogen.

Unsere Prozessphasen erstrecken sich von der Anforderungsformulierung bis hin zur Phase der Wartung, die wir hier aber nur Vollständigkeitshalber aufgeführt haben. Die Projektplanung umfasst die Phasen Anforderungen, Prozess-Design und Entwürfe. Die Umsetzung erfolgt in der Implementierungs- und Testphase (Tests & Fixes).

Ein Bild, das Text, Screenshot, Diagramm, Schrift enthält.

Automatisch generierte Beschreibung

Abbildung 2: Wasserfallmodell

Während der Planung- & Umsetzungsphasen ist es möglich vorherigen Phasen repetitiv zu wiederholen. Dabei wird sukzessive, ausgehend von der Phase, in der ein Problem festgestellt wurde in die nächstgelegene Phase zurückgekehrt, um den Ursprung der Fehlerquelle zurückzuverfolgen. Nach der Übergabe an den Kunden ist eine Fehlerbehebung in diesem Ausmaß nicht mehr möglich, weshalb die mögliche Rückführung in den vorherigen Schritt nicht mehr angezeigt ist.

Mit dem Fokus auf den Prozessen bzw. der Prozesskette haben wir uns zusätzlich noch für eine Prozesslandkarte entschieden, um unsere Abläufe zu optimieren, die Effizienz zu steigern und letztendlich unsere Ziele zu erreichen.

Ein Bild, das Text, Screenshot, Schrift, Diagramm enthält.

Automatisch generierte Beschreibung

Abbildung 3: Prozesslandkarte

Es empfiehlt sich der Top down Ansatz, dem wir gefolgt sind, d.h. wir schauen erst auf die Unternehmung als Ganzes um anschließend in die Prozessketten zu gehen. Da es bei einer Prozesslandkarte darum geht, „wie“ etwas gemacht wird und nicht „wer“ genau „was“ macht, haben wir zusätzlich noch einen Zeitplan (siehe Zeitplan.pdf der anhängten Dokumente) erstellt. In diesem haben wir die identifizierten Aufgaben festgehalten und die Ressourcen zugewiesen. In diesem Fall heißt das, wir haben die Teammitgliederkapazität den offenen To-Do´s zugeteilt. Die drei größten Wertschöpfungsprozesse (Oberfläche design (Front-End), Funktionen integrieren (Back-End) und die Erstellung der Datenbank) haben wir grob gesehen jeweils einem Teammitglied zugeteilt. In der Spalte „Pers.“ Des Zeitplanes ist zu sehen, wie wir die identifizierten Aufgaben dazu aufgeteilt haben. Die Koordination der Kollaboration haben wir gemeistert, in dem wir uns zweimal die Woche ausgetauscht haben. Bei diesem Austausch wurden Ergebnisse besprochen ggf. Veränderungen vorgenommen und das weitere Vorgehen geplant. Bei der Planung wurden konkrete To-Do´s benannt, deren Umsetzung jeweils bis zum nächsten Meeting geplant war. Die Meetings entsprechen demnach unseren Meilensteinen. Die Zeit bis dahin könnte man als Sprint betrachten. Einer unserer wertvollsten Tools der Unterstützungsprozesse ist die Nutzung von Git, was zum einen unsere Ergebnisse versioniert sichert und zusätzlich beim Austausch gemeinsamer Daten unterstützt. Der geplante zeitliche Verlauf bis zur Fertigstellung der konkreten Aufgaben ist im Zeitplan durch die farbliche Markierung der Wochenspalten eindeutig erkennbar. Beides zusammen, Prozesslandkarte und Zeitplan, bilden eine gute Kombination, um das „wie“, „was“ und „wer“ abzubilden.

# Design-Entwürfe

Für das Thema des Designs haben wir uns entschieden, dass unser „Space“, also unsere Mediendatenbank das Thema „Space“ bekommen soll, weil uns die Analogie marketing-technisch passend vorkam. Bevor wir konkrete Entwürfe zu diesem Thema entwerfen können, müssen aber noch grundlegendere Aspekte geklärt werden z.B. welche Ansichten gibt es? Welche Elemente werden benötigt und wo befinden sie sich?  
Die Frage, die wir zuerst beantworten müssen, ist, welche Ansichten es gibt. Bei uns gibt es eine Ansicht für alle (normalen) Benutzer. Diese werden nicht weiter unterschieden. Und die Ansicht des Administrators. Sie unterscheiden sich u.a. in den Aktionen, die die Nutzeroberfläche bietet. Weitere Informationen dazu unter „3.3. Technische Entwürfe“ zu finden. Hier möchten wir aber weiter auf die Design-Entwürfe eingehen. Wireframes sind ein nützliches Hilfsmittel, um die Anordnung der benötigen Elemente der verschiedenen Ansichten zu visualisieren.

Abbildung 3 – 5 geben einen Vorgeschmack auf diese Wireframes. Um sie besser betrachten zu können, nutzen Sie bitte die Dokumente Wireframe\_Login\_Page.pdf, Wireframe\_Admin\_Landing\_Page.pdf und Wireframe\_User\_Landping\_Page.pdf im Anhang.

Ein Bild, das Text, Screenshot, Rechteck, Diagramm enthält.

Automatisch generierte Beschreibung Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Abbildung 4: Wireframe\_Admin\_Landing\_Page Abbildung 5: Wireframe\_User\_Landping\_Page

Die Admin Landing Page soll Aktionen bereitstellen zur Verwaltung von Nutzern und Schlagwörtern und außerdem ein Dashboard zur Visualisierung von nützlichen Informationen, wogegen die User Ansicht den Fokus auf den Umgang mit den Medien legt.

Ein Bild, das Text, Screenshot, Rechteck, Diagramm enthält.

Automatisch generierte BeschreibungEinige Seiten, wie zum Beispiel die Login Page sind bei beiden Sichten identisch. Diese ehergroben Entwürfe der Seiten lassen sich durch Mockups weiter ausbauen. Man orientiert sich dabei an den Wireframes und fügt Details hinzu. Man sollte sich dabei vorab damit auseinander-setzen, welches Farbschema verwendet werden soll.Das Farbschema unseres Projekts hat sich daraus ergeben, dass wir von einem Hintergrund überzeugt waren (siehe Anhang „Background.jpg) und die restlichen Elemente darauf abgestimmt haben. Dadurch, dass der Hintergrund sehr dunkel ist, ist es für die Lesbarkeit essenziell, dass eine helle Schriftfarbe verwendet wird. Für Akzente bieten sich Grün, Blau oder Schwarz an, da diese Farben im Hintergrund enthalten sind. Wenn man etwas besonders hervorheben will, könnte man auch Komplementärfarben nutzen. Von der Farbe Blau wäre das beispielsweise Orange. Da unser Space ein eher ruhiges, homogenes Gesamtbild liefern soll, haben wir uns für Akzente im selben Farbton entschieden. Unser Styleguide im Überblick zeigt Tabelle 1.

Abbildung 6: Wireframe\_Login\_Page

|  |  |  |  |
| --- | --- | --- | --- |
| **Akzente** | **Farbe** | **Farbcode** | **Deckkraft** |
| Rahmen | Türkis | #02BFBF | 100% |
| Buttons | Türkis | #02BFBF | 50% |
|  | | | |
| Schrift | Farbe | Farbcode | Schriftart |
| Allgemein | Weiß | #FFFFFF | PangMenZhengDao\* |
| Ausgewählt | Türkis | #02BFBF | PangMenZhengDao\* |
| Auf weißen Hintergrund | Grau | Türkis | #555555 | #02BFBF | PangMenZhengDao\* |
| \*bei einigen Elementen wie dem Auswahlmenu auch Roboto | | | |
|  | | | |
| Bilder | Größe (in px) | Name im Anhang | Bild |
| Logo | 100x100 | Rakete.png |  |
| Upload-Button | 200x200 | Upload-Button.png |  |

Tabelle 1: Styleguide

Kernstück dieses Gesamtbildes soll der Upload-Button sein, weshalb wir uns bei diesem für ein auffälliges Design entschieden haben statt eines türkisen Standard-Buttons. Die dazugehörigen Mockups sind im Anhang zu finden unter den Namen Login-Page\_Mockup.jpg, Admin-Landinpage\_Mockup.jpg und User-Landingpage\_Mockup.jpg. Die Mockups wurden mit der Seite https://rp.mockplus.com/ erstellt. Bei der Implementierung ergab sich leider das Problem, dass uns die Schriftarten nicht zur Verfügung standen, weshalb wir auf die nächstähnliche Schriftart zurückgreifen mussten.

# Technische Entwürfe

Wir arbeiten in unserem Projekt mit der Datenbanksprache SQL (näheres dazu unter Systemvorbereitung). Es handelt sich also um eine relationale Datenbank. Um eine relationale Datenbank zu erstellen eignet sich ein E/R-Modell hervorragend um vorab alle Entitäten, also Dinge der realen Welt, in Entitätstypen zusammenzufassen und die Beziehung zwischen den einzelnen Entitätstypen zu beschreiben.

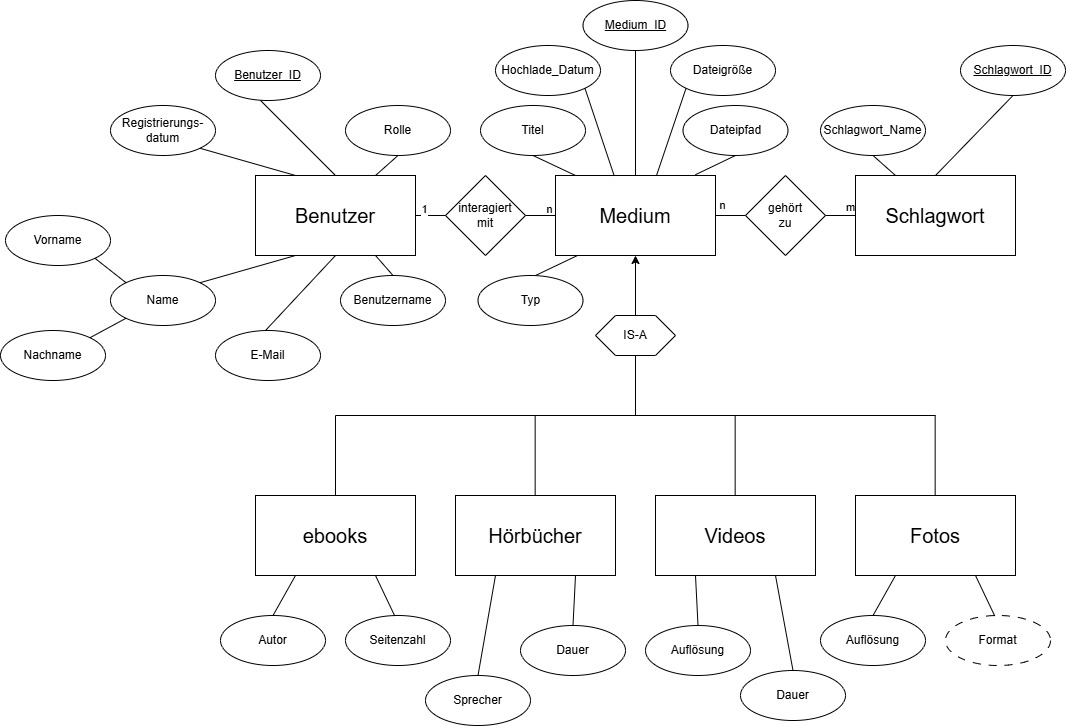


Abbildung 7: Enity Relationship Modell

Für eine größere Darstellung nutzen Sie bitte ER-Diagramm.jpg im Anhang.   
 Das E/R Diagramm beschreibt unsere Entitätstypen, wobei „ebooks“, „Hörbücher“, „Videos“ und „Fotos“ Spezialisierungen des Typs „Medium“ sind. Jede dieser Spezialsierungen besitzt die Attribute des General-Typs „Medium“ und zusätzliche noch ihre individuellen Attribute. Das individuelle Attribut „Format“ des Typs „Fotos“ ist ein abgeleitetes Attribut. Es ist kein für sich selbst stehendes Attribut, sondern es lässt sich aus dem Attribut Auflösung berechnen. Von Medium ausgehend stehen zwei weitere Entitätstypen in Beziehung zu diesem mit jeweils den Kardinalitäten einer 1 zu n und einer n zu m Beziehung. „Benutzer“ steht zu „Medium“ in einer 1 zu n Beziehung, da 1 Benutzer mit mehreren Medien interagieren kann (genaueres zu den Interaktionen der Benutzer unter „Use Case“ im übernächsten Abschnitt dieses Kapitels) aber ein Medium nur von einem Benutzer verwendet werden kann. Anders verhält sich das zwischen Schlagwörtern und Medien. Hier handelt es sich um eine n zu m Beziehung, denn ein Schlagwort kann zu mehreren Medien gehören und ein Medium kann mehrere Schlagwörter besitzen. Bei den unterstrichenden Attributen handelt es sich jeweils um die Primärschlüssel des Entitätstyps. Wir haben uns dafür entschieden diesen immer als ID, also eine fortlaufende, einzigartige Nummer, auszuführen, obwohl z.B. bei Benutzer ebenfalls der „Benutzername“ in Frage kommen würde, da man in SQL mit dem „auto increment“ Parameter arbeiten kann, was bei der Implementierung einiges erleichtert.

Aus diesem abstrakten Modell lässt sich ein konkreteres Modell ableiten: das relationale Modell. Aufgrund der Größe steht es nur im Anhang zur Verfügung (relationales Datenmodell.xlsx). Während der Konzeptphase haben wir im Team über die drei Lösungsvarianten debattiert.

Lösung 1: Eine Tabelle für jede Spezialisierung mit Basistabelle

Lösung 2: Eine Tabelle für jede Spezialisierung ohne Basistabelle

Lösung 3: Eine Tabelle für alle Entitäten

Wir haben uns dabei für Lösung 2 entschieden, da es eine gute Kompromisslösung aus den Nachteilen von Lösung 1 und 3 ist. Lösung 1 hat den Nachteil, dass relativ aufwändige Joins notwendig sind, wenn ein oder mehrere bestimmte Tupel ausgewählt werden sollen. Lösung 3 hat den Nachteil, dass die spezialisierten Attribute mit „null“ ausgefüllt werden müssen, wenn sie nicht von der Entität verwendet wird. Lösung 2 reduziert die Anzahl der Joins durch das Ausbleiben der Basistabelle, muss dafür zwar redundante Attribute in Kauf nehmen, kommt aber dafür ohne leere Felder aus.

Ein weiterer technischer Entwurf widmet sich den bereits erwähnten Interaktionen. Einen abstrakten, aber dafür übersichtlichen Überblick bietet der „Use Case“ siehe Abbildung 8.

Ein Bild, das Text, Fastfood, Essen enthält.

Automatisch generierte Beschreibung

Abbildung 8: Use Case

Wie wir in den Design-Entwürfen bereits festgestellt haben, gibt es in dem Projekt zwei Sichten, die sich durch die Verfügbarkeit unterschiedlicher Aktionen unterscheiden. Der Admin verfügt über die Admin-Konsole, in der Aktionen zur Nutzer- und globalen Schlagwortverwaltung auswählen kann. Er kann auf die Benutzer-Konsole wechseln, um die gleichen Aktionen ausführen zu können wie ein Benutzer. Dem Benutzer stehen zahlreiche Interaktionsmöglichkeiten mit den eigenen Medien und dem eigenen Account zur Verfügung inklusive eigens gewählter Schlagwörter. Eine zentrale Datenbank stellt das Herzstück dieser Interaktion dar. Sie wird verwaltet von einem Datenbankadministrator.

Hier kommt noch das Ablaufdiagramm von Yannik

Und das Klassendiagramm von Benjamin rein.

# Umsetzung

Verwendung von XAMPP etc.

# Systemvorbereitung

# Datenbank

# Front-End

# Back-End

# Systemtests

# Testfall 1

# Testfall 2

# Testfall 3

# Testfall 4

# Zusammenfassung

# Kritische Reflexion

# Quellenverzeichnis

Schumacher, Lutz (2024)  
WEB42 – Webprojekt I Labor 2024 (Zugriff 04.09.2024)

Prof. Plate, Jürgen; aktualisiert von Prof. Dr. Harwardt, Mark; Willamowski, Sonja (o.J.)  
Studienbrief INT102, Einführung in die Internetprogrammierung (Zugriff 04.09.2024)

Prof. Dr. Olderog, Torsten; Slama, Tim (o.J.)  
Studienbrief WEB602, Grundlagen der Programmierung mit PHP (Zugriff 04.09.2024)

Prof. Dr.-Ing. Timm, Heiko (o.J)  
Studienbrief DBA201, Einführung in die Datenbank-Programmierung   
mit MySQL und PHP (Zugriff 04.09.2024)

Sarah Laoyan (2024)  
Wasserfallmodell: Definition, Anwendung und Vorteile!   
URL: https://asana.com/de/resources/waterfall-project-management-methodology  
Zugriff am 19.08.2024

Julia Kupke (2023)  
Prozesslandkarten: Der Schlüssel zur Effizienzsteigerung in Unternehmen  
URL: https://www.d-velop.de/blog/prozesse-gestalten/prozesslandkarte/  
Zugriff am 19.08.2024

Simone Glitsch (o.J.)  
Wie Expert:innen eine Prozesslandkarte erstellen  
URL: https://prozessoptimierung-sprung.de/prozesslandkarte-erstellen/  
Zugriff am 19.08.2024

**Quellen der Abbildungen:**  
Abbildung 1  
https://git-scm.com/book/en/v2/Getting-Started-What-is-Git%3F  
Figure 6. Working tree, staging area, and Git directory (letzter Zugriff 09.09.2024)

Abbildung 2   
hier noch den Rest ausfüllen

# Dateianhänge

Eidesstattliche Erklärung

„Ich versichere, dass ich die beiliegende Ausarbeitung

selbstständig verfasst, keine anderen als die angegebenen Quellen und

Hilfsmittel benutzt sowie alle wörtlich oder sinngemäß übernommenen Stellen in der

Arbeit gekennzeichnet habe.“

Hamburg, 20.09.2024