Pflichtenheft für das Webprojekt „Bibliotheksverwaltung“

# Zielbestimmung

## Musskriterien

* Büchersuche über Volltext
* Login für Bibliothekare
* Ausleihe und Rückgabe, wobei der Zugriff nur für Bibliothekare gestattet ist
* Buchanlage bei Neuanschaffung (ebenfalls nur für Bibliothekare zugänglich)
* Buchdaten ändern (nur Bibliothekare)
* Buch ausscheiden (nur Bibliothekare)

## Wunschkriterien

* Büchersuche über verschiedene Kategorien
* Büchersuche über verschiedene Verlage
* Responsive Web-Design

# Produkteinsatz

## Anwendungsbereiche

Grundsätzlich soll die Web-App die Verwaltung von Büchern vereinfachen. Nach erfolgreichem Login kann auf Daten über Bücher und Ausleihen, welche in einer Datenbank gespeichert werden, zugegriffen werden.

## Zielgruppen

Einerseits ist die App auf Bibliothekare ausgerichtet, die einfach über den Browser Bücherausleihen verwalten können. Andererseits können reguläre Benutzer nach Büchern suchen und filtern.

## Betriebsbedingungen

Im Rahmen des Projekts wird die Anwendung ausschließlich lokal auf einem PC oder Notebook laufen. Aufgrund der Aufgabenstellung ist es nicht unbedingt notwendig und hinsichtlich der zu benützenden Technologien aufwendig, die App zu hosten.

# Produktumgebung

## Software

* Betriebssystem: Windows 11
* Datenbanksystem: MySQL + XAMPP
* NodeJS
* NPM (Node Package Manager)
* Frontend: JS / Vue3 (JS Framework), CSS / TailwindCSS, HTML
* Backend: ExpressJS (NodeJS Framework)

## Hardware

* Clientrechner (Hardware mit Win11 kompatibel)
* Genügend Speicherplatz (Festplattenspeicher), um die Datenbank betreiben zu können

## Produktschnittstellen

* Im Falle des Frontends mit Vue3 werden die Daten per http APIs im Backend abgefragt. Hierbei bietet es sich an, die Daten im JSON Format zu übertragen und diese dann im Frontend angemessen zu visualisieren.

# Produktfunktionen

## Büchersuche

Diese Funktion kann öffentlich von jedem benutzt werden. Bücher sollen über folgende Arten gesucht und gefiltert werden können:

* Über Texteingabe (Titel und Beschreibung)
* Über Kategorien (Wunschkriterium)
* Über Verlage (Wundschkriterium)

## Login

Um den Zugriff auf Bibliothekare zu beschränken, muss man sich einloggen, um nachfolgende Funktionen benutzen zu können. Im Großen und Ganzen können nach dem Login Bücher verwaltet werden können.

## Ausleihe und Rückgabe

In der Datenbank werden Daten über die Ausleihe und Rückgabe eines Buches festgehalten. Bibliothekare können in der App eintragen, wenn eine Person ein Buch ausleiht, bzw. zurückgibt und in welchem Zeitraum er dieses Buch behalten darf.

## Buchanlage und Neuanschaffung

Zudem haben Bibliothekare die Möglichkeit, neue Bücher in das System einzutragen. Dies wird über eine Zeile in einer Tabelle in der Datenbank realisiert.

## Buchdaten ändern

Wenn ein Buch eingetragen ist, kann ein Bibliothekar dieses bearbeiten.

## Buch ausscheiden

Sollte ein Buch nicht mehr vorhanden sein in der Bibliothek, dann kann ein Bibliothekar dieses Buch aus dem System / der Datenbank löschen.

# Produktdaten

## Bücherdaten

* ISBN
* Titel
* Beschreibung
* Verlag
* Anschaffungspreis
* Kategorie

## Ausleihdaten

* Ausleiher
* Bibliothekar
* Buch
* Ausleihdatum
* Rückgabedatum

## Bibliothekar

* Vorname
* Nachname
* E-Mail
* Benutzername

# Benutzungsoberfläche

Im Folgenden werden Skizzen angegeben, wie die Benutzeroberfläche grob gestaltet werden könnte:

## Bildersuche



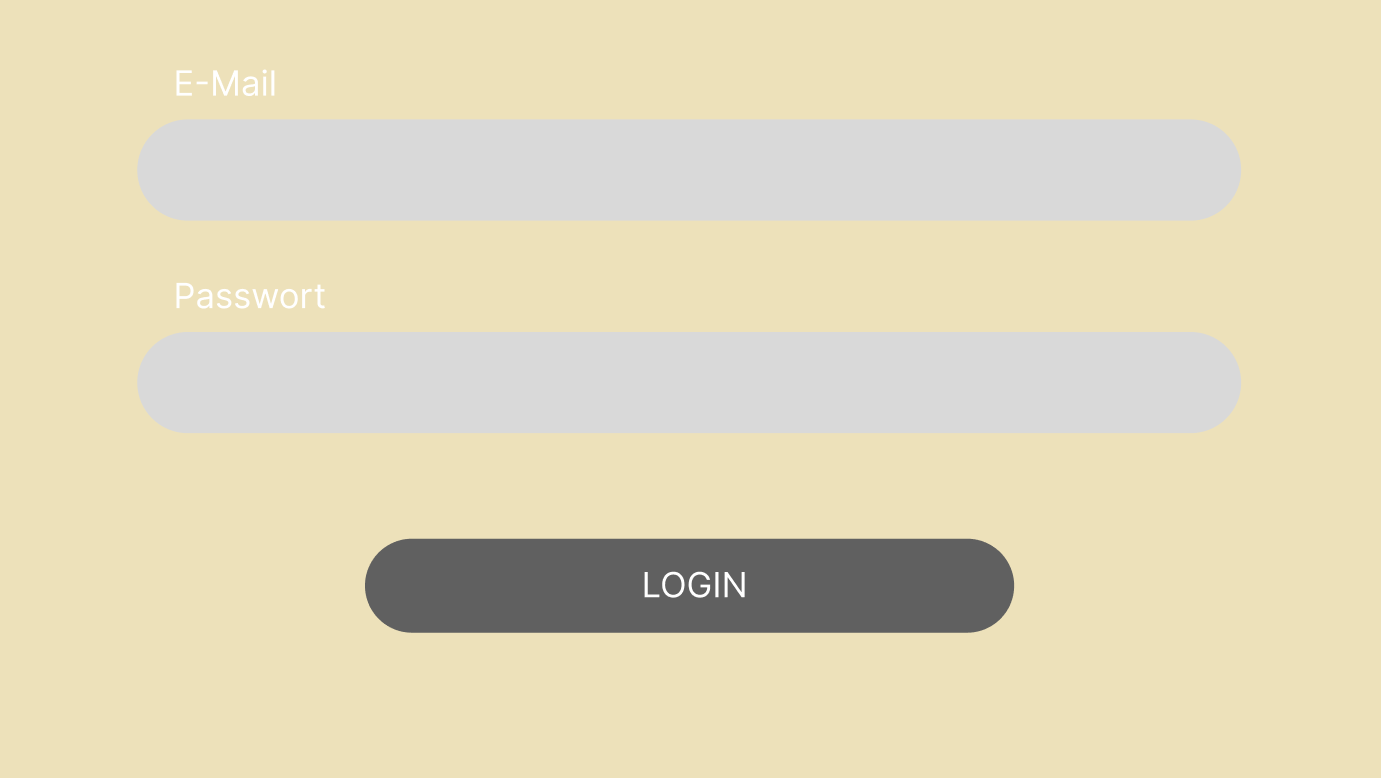
## Bücherverwaltung



## Bücherverleihung



## Login



# Qualitätsanforderungen

## Design

Vorausgesetzt wird, dass das Design für sich selbst spricht und nichts erklärt werden muss. Aus der Sicht des Benutzers müssen alle Funktionen ersichtlich sein.

# Globale Testszenarien und Testfälle

## Testfall 1

## Testfall 2

# Entwicklungsumgebung

## Software

* Betriebssystem: Windows Server 2022
* Datenbanksystem: MySQL + XAMPP
* NodeJS
* NPM (Node Package Manager)
* ExpressJS (NodeJS Framework)

## Hardware

* Schulserver
* Ausreichend Speicherplatz (Festplattenspeicher), um die Datenbank betreiben zu können

## Entwicklungsschnittstellen

* Wie auch in der Produktumgebung muss das Frontend mit dem Backend kommunizieren können, um den aktuellen Stand zu testen. Wie beschrieben werden mit Vue3 die Daten per http Schnittstellen bereitgestellt.