|  |  |  |  |
| --- | --- | --- | --- |
| **Übersicht** | | | |
| **Projekt** |  | | |
| **Projektleiter** |  | **Datum** |  |

Inhalt

[1. Zieldefinition 2](#_Toc191655099)

[2. Anwendungsfälle im Überblick 3](#_Toc191655100)

[3. Anwendungsfälle im Detail 5](#_Toc191655101)

[3.1. Anwendungsfälle Benutzerkonto 5](#_Toc191655102)

[3.2. Anwendungsfälle Karteikarten 6](#_Toc191655103)

[3.3. Anwendungsfälle 3 7](#_Toc191655104)

[4. Daten und Fakten als Grundlage der Qualitätsanforderungen 13](#_Toc191655105)

[5. Anforderungen im Überblick 14](#_Toc191655106)

[6. Qualitäts-Anforderungen im Detail 15](#_Toc191655107)

[7. Übersicht der relevanten Daten 15](#_Toc191655108)

[8. Glossar 15](#_Toc191655109)

[9. Ausfüllhilfe 16](#_Toc191655110)

[Anwendungsfälle 16](#_Toc191655111)

[Anforderungen 17](#_Toc191655112)

# Zieldefinition

***S****pecific (spezifisch)  
Ziele müssen eindeutig und verständlich sein.*

Eine Webanwendung entwickeln, die es Nutzern ermöglicht, digitale Karteikarten zu erstellen, zu verwalten und mit diesen zu lernen.

***M****easurable (messbar):   
Ziele müssen messbar sein*

Das Projekt gilt als erfolgreich, wenn die Karteikartenapp innerhalb des vorgegebenen Zeitraums erfolgreich implementiert und getestet wurde.

***A****ppropriate /* ***A****ttractive (angemessen/wünschenswert):  
Ziele müssen den Aufwand rechtfertigen.*

Die Anwendung bietet einen Mehrwert durch intuitive und personalisierte Karteikartenverwaltung. Auswählbare Lernmethoden (z. B. Leitner-System, Spaced Repetition, Interleaving) erhöhen die Lerneffizienz und sorgen für einen USP(Alleinstellungsmerkmal).

***R****ealistic (realistisch):   
Ziele müssen mit den gegebenen Mitteln erreicht werden können.*

Der Einsatz von etablierten Technologien wie HTML, CSS, TypeScript, einer Backend-Lösung (Java) und einer Datenbank (MySQL) stellt sicher, dass die Kernfunktionen auch durch ein kleines Entwicklerteam innerhalb des vorgegebenen Zeitrahmens realisiert werden können.

***T****ime-based (zeitbasiert):   
Ziele müssen in absehbarer Zeit erreicht werden können*

Die Implementierung erfolgt in 4 Phasen.

* 1.⁠ ⁠Zwischenabgabe am 30.01.2025: GitHub Repository mit (leerem) Projekt, funktionierender GitHub Action für den Build und GitHub Pages (oder einem Äquivalent).
* 2.⁠ ⁠Zwischenabgabe: Frontend (Ende erstes Semester)
* 3.⁠ ⁠Zwischenabgabe: Festlegen von Backend Technologien (Anfang des zweiten Semesters)
* 4.⁠ Finale Abgabe Quellcode + Dokumentation

Alle Zwischenstände müssen eine Woche vor der festgelegten Abgabefrist fertiggestellt sein.

# Anwendungsfälle im Überblick

B = Benutzerkonto, K= Karteikarten, V = Verwaltung

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Titel | Beschreibung | Prio | Kritikalität |
| AF-B-01 | Registrierung | Ein neuer Benutzer kann sich mit E-Mail und Passwort registrieren | Mittel |  |
| AF-B-02 | Anmeldung | Ein Benutzer kann sich mit seinen Zugangsdaten anmelden. |  |  |
|  |  |  |  |  |
| AF-K-01 | Lernmodus | Benutzer können mit verschiedenen Lernmodi Karteikarten lernen | Hoch | Kritisch |
| AF-K-02 | Set erstellen | Benutzer muss eine Art Ordner erstellen, worin Karteikarten gespeichert werden. |  |  |
| AF-K-03 | Set bearbeiten | Benutzer kann Sets bearbeiten und löschen. |  |  |
| AF-K-04 | Karteikarten erstellen | Benutzer können neue Karteikarten mit Vorder- und Rückseite anlegen. |  |  |
| AF-K-05 | Karteikarten bearbeiten | Benutzer können bestehende Karteikarten bearbeiten und löschen. |  |  |
| AF-K-06 | Kategorisierung | Kategorisierung mit Karteikartensets ermöglichen zur besseren Struktur im Lernbereich | Niedrig | Unkritisch |
|  |  |  |  |  |
| AF-V-01 | Fortschritt speichern | Lernfortschritte werden im Benutzerprofil gespeichert. |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Anwendungsfälle = was machen anwender themengebiet anzeigen ist kein use case aber themengebiet anlegen wäre

Eins/ muss in aktiver form geschrieben werden benutzer legt neues unterthema an z.b

Themengebiet anlegen sortieren löschen bearbeiten, karteikarte löschen sortieren registrieren einloggen einstellungen wie sprache umstellen wäre auch ein anwendungsfall.

Anforderung und anwendungen klar voneinander trennen

Ablauf als Ziffern runternummerieren.

Jeder Alternative path ist auch ein use case nur namen von dem use case angeben

Ein abbruch use case damit man es nicht immer wieder erwähnen muss.

# Anwendungsfälle im Detail

## Anwendungsfälle Benutzerkonto

|  |  |
| --- | --- |
| ID | AF-X-01 |
| Titel | Registrierung |
| Beschreibung | Ein neuer Benutzer kann sich mit E-Mail und Passwort registrieren. |
| Akteure | Benutzer (Kunde) |
| Vorbedingungen  *(Was muss erfüllt sein, bevor der Use Case beginnt?)* | Nutzer noch nicht angemeldet |
| Nachbedingungen  *(Was muss erfüllt sein, wenn der UseCase erfolgreich beendet ist?)* | Nutzerdaten müssen gespeichert werden. Rückmeldung vom System, dass es funktioniert hat. |
| Ablauf/Main Path *(Der erwartete oder gewöhnliche Weg des Use Case)* | Nutzer gibt seine persönlichen Daten ein. Danach bestätigt er seine Angaben. Diese werden verarbeitet, und es wird dem Nutzer Feedback gegeben. Weiterleitung zur Startseite. |
| Erweiterung/ Alternative Path  *(Ablauf für definierte Ausnahmen)* | Wenn Registrierung fehlschlägt, erfolgt keine Weiterleitung, sondern er bleibt auf der Seite und bekommt eine Fehlermeldung. |
| Priorität / Kritikalität | Prio: 3 und Kritikalität: 5 |

|  |  |
| --- | --- |
| ID | AF-X-01 |
| Titel | Anmeldung |
| Beschreibung | Ein Benutzer kann sich mit seinen Zugangsdaten anmelden. |
| Akteure | Benutzer (Kunde) |
| Vorbedingungen  *(Was muss erfüllt sein, bevor der Use Case beginnt?)* | Registrierung erfolgreich abgeschlossen, Nutzerdaten im Backend gespeichert, noch nicht angemeldet. |
| Nachbedingungen  *(Was muss erfüllt sein, wenn der UseCase erfolgreich beendet ist?)* | Nutzer angemeldet, personalisierte Startseite, Lernsets werden angezeigt und sind nutzbar. Nutzerdefinierte Einstellungen und Lernfortschritte sind speicherbar. |
| Ablauf/Main Path *(Der erwartete oder gewöhnliche Weg des Use Case)* | Nutzer gibt seine persönlichen Daten ein. Danach bestätigt er seine Angaben. Diese werden verarbeitet, und es wird dem Nutzer Feedback gegeben. Weiterleitung zur personalisierten Startseite. |
| Erweiterung/ Alternative Path  *(Ablauf für definierte Ausnahmen)* | Wenn Anmeldung fehlschlägt, erfolgt keine Weiterleitung, sondern er bleibt auf der Seite und bekommt eine Fehlermeldung. Mögliche Passwort Zurücksetzung über Button. |
| Priorität / Kritikalität | Prio: 3 und Kritikalität: 4 |

## Anwendungsfälle Karteikarten

|  |  |
| --- | --- |
| ID | AF-X-01 |
| Titel | Lernmodus |
| Beschreibung | Benutzer können mit verschiedenen Lernmodi Karteikarten lernen |
| Akteure | Benutzer (Kunde) |
| Vorbedingungen  *(Was muss erfüllt sein, bevor der Use Case beginnt?)* | Karteikarten müssen existieren. |
| Nachbedingungen  *(Was muss erfüllt sein, wenn der UseCase erfolgreich beendet ist?)* |  |
| Ablauf/Main Path *(Der erwartete oder gewöhnliche Weg des Use Case)* | User klickt auf Karteikartenset, danach wird er auf die Lernseite weitergeleitet. Dort sieht er eine Frage aus seinem Lernset, durch draufklicken kann er sich die Antwort anzeigen lassen und bewerten, wie schwer er die Frage fand. Nach der Selbstkorrektur ob richtig oder falsch wird die nächste Frage angezeigt. |
| Erweiterung/ Alternative Path  *(Ablauf für definierte Ausnahmen)* |  |
| Priorität / Kritikalität | Prio: 10 und Kritikalität: 10 |

|  |  |
| --- | --- |
| ID | AF-Y-02 |
| Titel | Set erstellen |
| Beschreibung |  |
| Akteure | Benutzer (Kunde) |
| Vorbedingungen  *(Was muss erfüllt sein, bevor der Use Case beginnt?)* |  |
| Nachbedingungen  *(Was muss erfüllt sein, wenn der UseCase erfolgreich beendet ist?)* |  |
| Ablauf/Main Path *(Der erwartete oder gewöhnliche Weg des Use Case)* |  |
| Erweiterung/ Alternative Path  *(Ablauf für definierte Ausnahmen)* |  |
| Priorität / Kritikalität | Prio: 10 und Kritikalität: 10 |

|  |  |
| --- | --- |
| ID | AF-Y-02 |
| Titel | Set bearbeiten |
| Beschreibung |  |
| Akteure | Benutzer (Kunde) |
| Vorbedingungen  *(Was muss erfüllt sein, bevor der Use Case beginnt?)* |  |
| Nachbedingungen  *(Was muss erfüllt sein, wenn der UseCase erfolgreich beendet ist?)* |  |
| Ablauf/Main Path *(Der erwartete oder gewöhnliche Weg des Use Case)* |  |
| Erweiterung/ Alternative Path  *(Ablauf für definierte Ausnahmen)* |  |
| Priorität / Kritikalität | Prio: 10 und Kritikalität: 10 |

|  |  |
| --- | --- |
| ID | AF-Y-02 |
| Titel | Karteikarten erstellen |
| Beschreibung | Benutzer können neue Karteikarten mit Vorder- und Rückseite anlegen. |
| Akteure | Benutzer(Kunde) |
| Vorbedingungen  *(Was muss erfüllt sein, bevor der Use Case beginnt?)* | Der Benutzer ist angemeldet. Und hat ein Set erstellt worin Karteikarten gespeichert werden können. |
| Nachbedingungen  *(Was muss erfüllt sein, wenn der UseCase erfolgreich beendet ist?)* | Karten müssen im Set gespeichert und abrufbar sein. |
| Ablauf/Main Path *(Der erwartete oder gewöhnliche Weg des Use Case)* | Man klickt im Set auf „neue Karte erstellen“.  Man gibt die Frage und die zugehörige Antwort in die dafür vorgesehenen Felder ein. Und ordnet der Karte eine gewünschte Schwierigkeit zu. Auf „Karte speichern“ klicken. |
| Erweiterung/ Alternative Path  *(Ablauf für definierte Ausnahmen)* | Karte erstellen abbrechen. |
| Priorität / Kritikalität | Prio: 10 und Kritikalität: 9 |

## Anwendungsfälle 3

|  |  |
| --- | --- |
| ID | AF-X-01 |
| Titel |  |
| Beschreibung |  |
| Akteure |  |
| Vorbedingungen  *(Was muss erfüllt sein, bevor der Use Case beginnt?)* |  |
| Nachbedingungen  *(Was muss erfüllt sein, wenn der UseCase erfolgreich beendet ist?)* |  |
| Ablauf/Main Path *(Der erwartete oder gewöhnliche Weg des Use Case)* |  |
| Erweiterung/ Alternative Path  *(Ablauf für definierte Ausnahmen)* |  |
| Priorität / Kritikalität |  |

|  |  |
| --- | --- |
| ID | AF-X-01 |
| Titel |  |
| Beschreibung |  |
| Akteure |  |
| Vorbedingungen  *(Was muss erfüllt sein, bevor der Use Case beginnt?)* |  |
| Nachbedingungen  *(Was muss erfüllt sein, wenn der UseCase erfolgreich beendet ist?)* |  |
| Ablauf/Main Path *(Der erwartete oder gewöhnliche Weg des Use Case)* |  |
| Erweiterung/ Alternative Path  *(Ablauf für definierte Ausnahmen)* |  |
| Priorität / Kritikalität |  |

|  |  |
| --- | --- |
| ID | AF-X-01 |
| Titel |  |
| Beschreibung |  |
| Akteure |  |
| Vorbedingungen  *(Was muss erfüllt sein, bevor der Use Case beginnt?)* |  |
| Nachbedingungen  *(Was muss erfüllt sein, wenn der UseCase erfolgreich beendet ist?)* |  |
| Ablauf/Main Path *(Der erwartete oder gewöhnliche Weg des Use Case)* |  |
| Erweiterung/ Alternative Path  *(Ablauf für definierte Ausnahmen)* |  |
| Priorität / Kritikalität |  |

# Daten und Fakten als Grundlage der Qualitätsanforderungen

Wie viele daten muss ich verarbeiten, wie viele user hat man wie viele schnittstellen und und und

Nicht funktionale anforderungen müssen immer messbar sein Sicherheitsanforderungen sind es nicht sind funktional

Systemanforderungen/performance anforderungen(Fragebogen irgendwo)/ usability anforderungen

Technische Anforderungen:

 Performance: Die Ladezeit der Anwendung soll unter 2 Sekunden bleiben.

Funktionale Anforderungen:

 Benutzerfreundlichkeit: Die Oberfläche soll intuitiv und leicht bedienbar sein.

 Lernmethoden: Integration von Spaced Repetition und Leitner-System zur besseren Wissensvermittlung.

 Synchronisation: Alle Daten sollen in Echtzeit zwischen Geräten synchronisiert werden.

Sicherheits- und Datenschutzanforderungen:

 DSGVO-Konformität: Die App muss den Datenschutzrichtlinien der EU entsprechen.

 Authentifizierung: Nutzung von OAuth 2.0 oder JWT zur sicheren Anmeldung.

 Backups: Tägliche automatische Datensicherung für Notfälle.

4. Usability & Design Standards

Responsive Design: Optimierte Darstellung für Smartphones, Tablets und PCs.

Barrierefreiheit: Einhaltung von WCAG 2.1 für bessere Nutzbarkeit.

Dark Mode: Option für dunkles und helles Design.

5. Skalierbarkeit und Erweiterbarkeit

Modulares System: Die Anwendung soll leicht erweiterbar sein.

API-Schnittstellen: Bereitstellung einer RESTful API für externe Erweiterungen.

Cloud-Hosting: Nutzung einer skalierbaren Cloud-Lösung zur Lastverteilung.

# Anforderungen im Überblick

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Titel | Beschreibung | Prio | Kritikalität |
| A-01-F0 |  |  |  |  |
| A-01-F1 |  |  |  |  |

Anforderungen sind anwendungen in der Umsetzung und wie/wann diese stattfindet das system muss bieten

# Qualitäts-Anforderungen im Detail

Zusätzliche Informationen zu unterschiedlichen Qualitätsanforderungen wie Definition und Bewertung relevanter Qualitätsmerkmale (z.B. Anhand eines Qualiättsfragebogens)

# Übersicht der relevanten Daten

Welche Daten, Dateien, Datenbanken, Tabellen und Felder sind relevant?

Softwareengineering was brauche ich datenbanken datamodells macht matthias der idiot klassenmodell

# Glossar

**Anwendungsfall:** Sammlung aller Szenarien zur Erreichung eines Ziels aus Sicht des Akteurs)

**Funktionale Anforderung:** Eine funktionale Anforderung ist eine Anforderung bezüglich eines Ergebnisses oder eines Verhaltens, das von einer Funktion eines Systems bereitgestellt werden soll.

**Nicht-Funktionale Anforderung:**Nichtfunktionale Anforderungen beschreiben Aspekte, die typischerweise mehrere oder alle funktionalen Anfor1derungen betreffen bzw. überschneiden.

# Ausfüllhilfe

**Zieldefinition**Zieldefinition wie sie im Projektantrag zu finden ist. Bei Revisionen anzupassen

## Anwendungsfälle

**ID**Systemtische und eindeutige Nummerierung der Anwendungsfälle.

**Titel**Beschreibender Name des Anwendungsfalls für die kurze Bezugsname im Dokument.

**Beschreibung**Kurze Beschreibung des Anwendungsfalls in einem oder zwei Sätzen.

**Akteure**Personen, Systeme oder Organisatorische Einheiten die in diesem Anwendungsfall aktiv sind.

**Vorbedingungen**Zustand aller relevanten Systeme zu Beginn des Anwendungsfalls.

**Nachbedingungen**Zustand aller relevanten Systeme beim Abschluss des Anwendungsfalls im gewünschten Ablauf.

**Ablauf/Main Path**Der erwartete oder gewöhnliche Verlauf des Anwendungsfalls

**Erweiterung/Alternative Path**Ablauf für definierte Abweichungen vom Ablauf

**Priorität**Position in der Umsetzungsreihenfolge der Anwendungsfälle.

**Kritikalität**Eigenschaft, ob ein Anwendungsfall den Projekterfolg verhindern kann.

## Anforderungen

**ID**Systemtische und eindeutige Nummerierung der Anwendungsfälle inklusive ihres Typs.

**Titel**Beschreibender Name des Anwendungsfalls für die kurze Bezugsname im Dokument.

**Beschreibung**Kurze Beschreibung der Anforderung entsprechend der Satzschablone.

**Priorität**Position in der Umsetzungsreihenfolge der Anforderungen.

**Kritikalität**Eigenschaft, ob eine Anforderung den Projekterfolg verhindern kann.