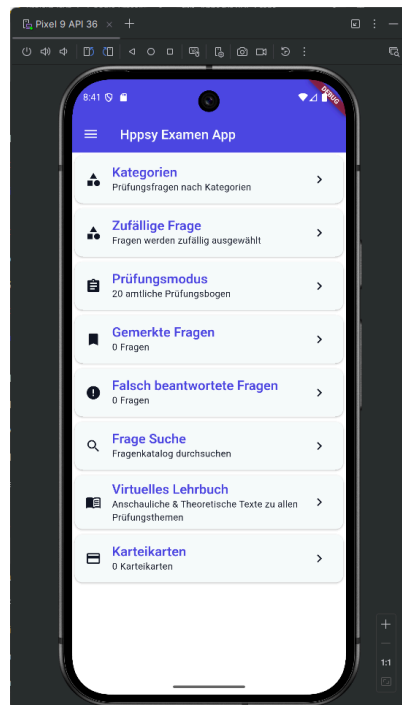


Java Projekt 3

Prüfungstrainer App
Abgabetermin 29.06.2025



Fachinformatiker Anwendungsentwicklung
Marc Böhnert

Ausbildungsbetrieb

cbm GmbH

vertreten durch die geschäftsführende Gesellschafterin Katrin Querfeld

Wegesende 3 – 4

28195 Bremen

1. Projektziele	3
2. Funktionale nicht Funktionale Anforderungen	3
3. Ausgewählte Hard- & Software	4
4. Vorgehensmodell	4
5. Zeitplan (Gant-Diagramm)	5
6. Nutzwertanalyse	6
8. SOLL & IST Analyse	8
9. Meilensteinplanung	9
10. Kostenkalkulation	9
11. Datenbank	10
12. Abschlussbericht	11
13. Qualitätssicherung und Tests	11
14. Anhang	13

1. Projektziele

Die Projektdokumentation wurde zu Übungszwecken erstellt. Sie dient der Vorbereitung auf die IHK - Abschlussprüfung im Sommer 2026.

2. Funktionale nicht Funktionale Anforderungen

Das Projektziel ist eine App zu programmieren, die es ermöglichen soll, sich auf die Prüfung zum Heilpraktiker für Psychotherapie optimal vorbereiten zu können. Dazu sollten falsch beantwortete Fragen gespeichert werden, um diese zu wiederholen. Wenn eine Frage 3x richtig beantwortet wurde, ist sie als Prüfungsreif zu markieren. Der User kann sich Fragen merken, um sie später wieder aufrufen zu können.

Die App soll für folgende Plattformen bereitgestellt werden:

- Android
- iOS
- Web

Die Rohdaten werden in ein einheitliches Format gebracht. Das von der App digital verarbeitet werden kann. Damit die App im Offline-Modus funktioniert, muss die Datenbank lokal gespeichert werden. Die Datenbank muss verschlüsselt sein, der Zugriff wird über einen API Schlüssel realisiert. Der API Schlüssel läuft wöchentlich ab und muss erneuert werden.

Der Nutzer muss sich registrieren und anmelden können. Die Registrierung soll zusätzlich über ein Google Konto möglich sein. Nutzerdaten werden auf dem Server gespeichert. Die Einhaltung des Datenschutzes ist unbedingt einzuhalten. Die Daten der Nutzer müssen so gespeichert werden, dass DSGVO-Anfragen automatisch beantwortet werden können.

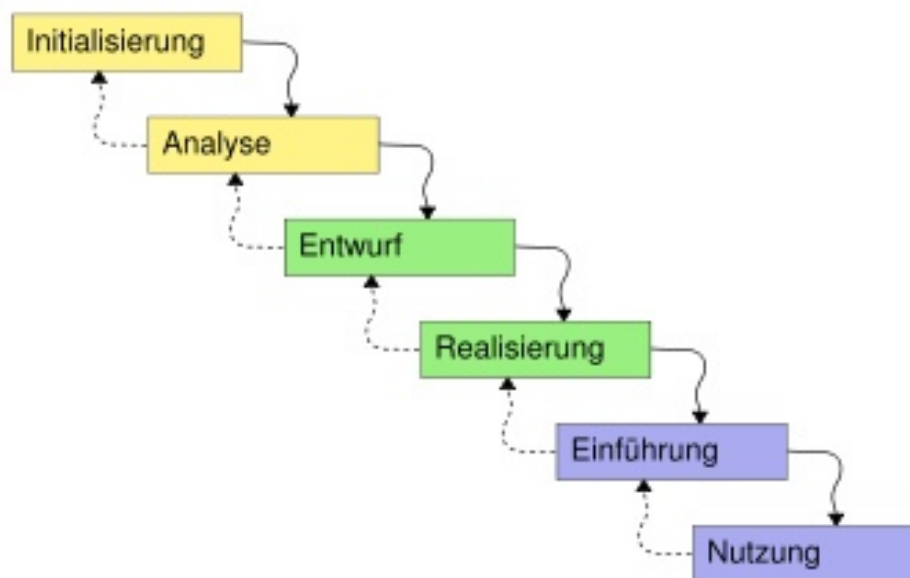
Es soll kostenpflichtige Funktionen geben. Zu diesem Zeitpunkt ist nicht klar, welche dies sein werden. Es soll im Entwicklungsprozess aber schon berücksichtigt werden. Die gesamte App soll nach modernen Software-Architektur Richtlinien umgesetzt werden.

3. Ausgewählte Hard- & Software

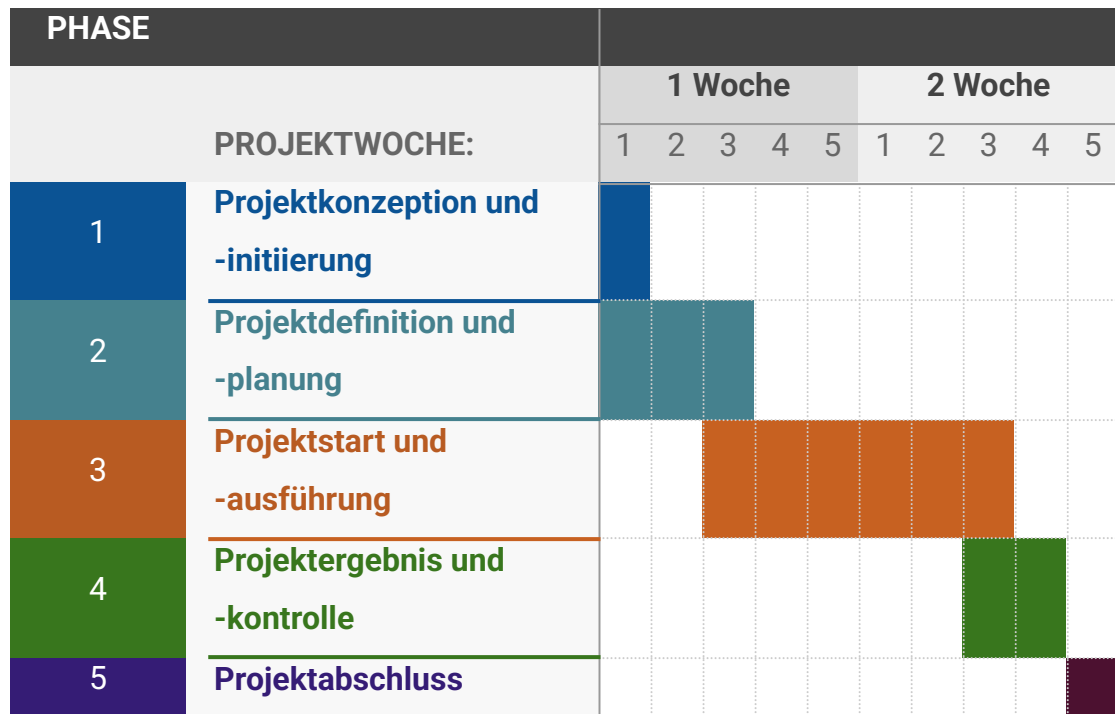
Ich habe den Computer aus der Schule zur Verfügung gehabt. Dies ist ein Thin Client mit einem aktuellen i7 Prozessor. Als Betriebssystem steht Windows 11 zur Verfügung. Sowie ausreichend Speicherplatz. Eine Lizenz von IntelliJ Idea in der Pro Version wurde durch den Auftraggeber bereitgestellt.

4. Vorgehensmodell

Als Vorgehensmodell wurde das erweiterte Wasserfallmodell gewählt. Dies wird damit begründet, dass die Anforderungen an das Projekt sehr klar sind im Vorfeld. Eine gute Kostenschätzung war auf diese Weise möglich.



5. Zeitplan (Gant-Diagramm)



6. Nutzwertanalyse

Nutzwertanalyse

Kriterium	Gewicht	IntelliJ IDEA	Visual Studio	Kotlin	Flutter	Swing
Produktivität	25 %	10	8	9	10	9
Ökosystem & Community	20 %	10	8	10	10	10
Konfiguration & Wartbarkeit	15 %	10	5	9	10	8
Performance	15 %	9	9	8	10	8
Multiplattform fähig	10 %	10	10	8	10	10
Lizenz / Kosten	5 %	1	10	10	10	10
Zukunftssicherheit	10 %	9	9	10	10	9
Gesamtnutzen	100 %	9,30	8,10	9,1	10,00	9,05

Sachmittelplanung

Am Arbeitsplatz von CBM steht ein Dell -Optiplex 7060 inklusive 2 Monitoren bereit. Für das Homeoffice steht ein Macbook M2 sowie ein zusätzlicher Monitor mit externer Maus und Tastatur bereit. Ausreichende Internetverbindungen stehen an beiden Ort zur Verfügung (mindestens DSL 16.000). Als IDE wird IntelliJ Idea in der Ultimate Variante benutzt. Als Framework wird Spring Boot verwendet, dieses Framework ist Java Open-Source-Framework

7. Funktionale und nicht funktionale Anforderung

Fragen und Antworten

Die App soll eine Sammlung von Prüfungsfragen enthalten, die der Nutzer beantworten kann Jede Frage hat 5 Antwortmöglichkeiten von den 1 bis 4 richtig sein können Der Nutzer kann die richtigen und falschen Antworten auswerten lassen.

Fragen wiederholung und Speicherung:

Falsch beantwortete Fragen sollen gespeichert werden, damit der Nutzer sie später erneut beantworten kann.

Jede Frage soll eine Zählung der richtigen Antworten haben. Sobald eine Frage 3-mal richtig beantwortet wurde, wird sie als "prüfungsreif" markiert.

Fragen merken:

Der Nutzer kann bestimmte Fragen markieren, um sie später schnell wieder aufzurufen und zu wiederholen.

Offline-Modus:

Die App muss offline funktionsfähig sein, indem alle relevanten Daten lokal gespeichert werden.

Die Datenbank muss verschlüsselt sein, um die Sicherheit der Nutzerdaten zu gewährleisten.

Datenbank:

Die Datenbank muss Fragen und Antworten in einem einheitlichen Format speichern, das von der App digital verarbeitet werden kann.

Die Datenbank muss lokal auf dem Gerät gespeichert werden, sodass sie auch im Offline-Modus verfügbar ist.

Die Datenbank muss regelmäßig mit einem API-Schlüssel abgerufen und aktualisiert werden. Der API-Schlüssel läuft wöchentlich ab und muss automatisch erneuert werden.

Benutzerregistrierung und Anmeldung:

Der Nutzer muss sich registrieren und anmelden können.

Die Registrierung kann entweder manuell oder über ein Google-Konto erfolgen. Alle Nutzerdaten müssen sicher gespeichert werden.

Datenschutz und DSGVO:

Die App muss alle relevanten DSGVO-Vorgaben einhalten. Nutzerdaten müssen so gespeichert werden, dass DSGVO-Anfragen automatisch beantwortet werden können. Eine Funktion zur Datenlöschung (auf Anfrage des Nutzers) muss eingebaut werden.

Kostenpflichtige Funktionen:

Es sollen kostenpflichtige Funktionen in die App integriert werden, die im

Entwicklungsprozess berücksichtigt werden müssen. Die Funktionen könnten auf einer Abo- oder Kaufbasis angeboten werden.

API-Integration:

Der Zugriff auf die Datenbank erfolgt über einen API-Schlüssel, der wöchentlich erneuert werden muss. Die API muss sicherstellen, dass nur berechtigte Anfragen Zugriff auf die Datenbank haben.

Multiplattform-Unterstützung:

Die App muss auf den Plattformen Android, iOS und Web bereitgestellt werden.

8. SOLL & IST Analyse

IST-Analys

Es steht eine Entwicklungsumgebung bereit. Die nötigen Fähigkeiten und das Wissen liegen dem Entwickler vor. Ein Datensatz von Fragen ist vorhanden und kann für die App verwendet werden. Der Fragensatz beinhaltet Fragen, die immer 5 mögliche Antworten beinhalten. Es können eine oder mehrere richtige Antworten möglich sein. Es besteht ein Sicherheitskonzept für die Datenbank und die API Kommunikation. Der Funktionsumfang wurde ausgearbeitet.

SOLL-Analyse

Das Hauptziel ist ein App zur Prüfungsvorbereitung für die HPP-Prüfung. Die Nutzer der App sollen sich optimal auf Ihre Prüfung vorbereiten können. Die App soll für verschiedenen Plattformen veröffentlicht werden. Dazu zählen Android, iOS und das Web. Das UI / UX soll ansprechend sein und den Nutzen in seinen Bann ziehen, damit der Nutzer mehr Zeit in der App verbringt. Es soll außerdem Benutzerfreundlich und responsive sein. Dieses Ziel soll über Gamification erreicht werden. Die DSGVO-konforme Speicherung liegt im Fokus der Entwicklung. Es soll eine Reihe von Funktionen umgesetzt werden. Ein Teil der App wird kostenpflichtig sein. Die Daten sollen lokal auf dem Gerät gespeichert werden. Die Daten in der Datenbank sind zu verschlüsseln. Die API Kommunikation mit dem Server ist ebenfalls zu verschlüsseln. Dieser API Schlüssel hat eine begrenzte Gültigkeit und muss regelmäßig erneuert werden. Als Datenbank soll eine SQLite Datenbank

verwendet werden. Somit garantieren wir eine Offline Funktionalität bei maximaler Datensicherheit. Für den Nutzer gibt es eine Benutzerregistration und eine Benutzeranmeldung. Eine Datensicherung im Nutzerkonto ist vorgesehen. Das Projekt soll nach aktuellen Architektur Standards umgesetzt werden.

9. Meilensteinplanung

Meilenstein	Inhalt	Geplanter Zeitpunkt
M1 – Projektstart	Projektdefinition, Zielsetzung, Ist-Analyse	Tag 1
M2 – Planung abgeschlossen	Pflichtenheft, Technologiewahl, Meilenstein- & Zeitplan	Tag 2–3
M3 – Implementierungsbeginn	Datenmodell, GUI-Entwurf, erste Codebasis	Tag 4
M4 – Funktionaler Prototyp	Grundfunktionen umgesetzt, erste Tests	Tag 8
M5 – Fertige Anwendung	Alle Kernfunktionen entwickelt und getestet	Tag 13
M6 – Projektabschluss intern	Code-Review, Abnahme durch Ausbilder	Tag 14
M7 – Projektdokumentation fertig	Vollständige Doku + Vorbereitung für Präsentation	Tag 15–16

10. Kostenkalkulation

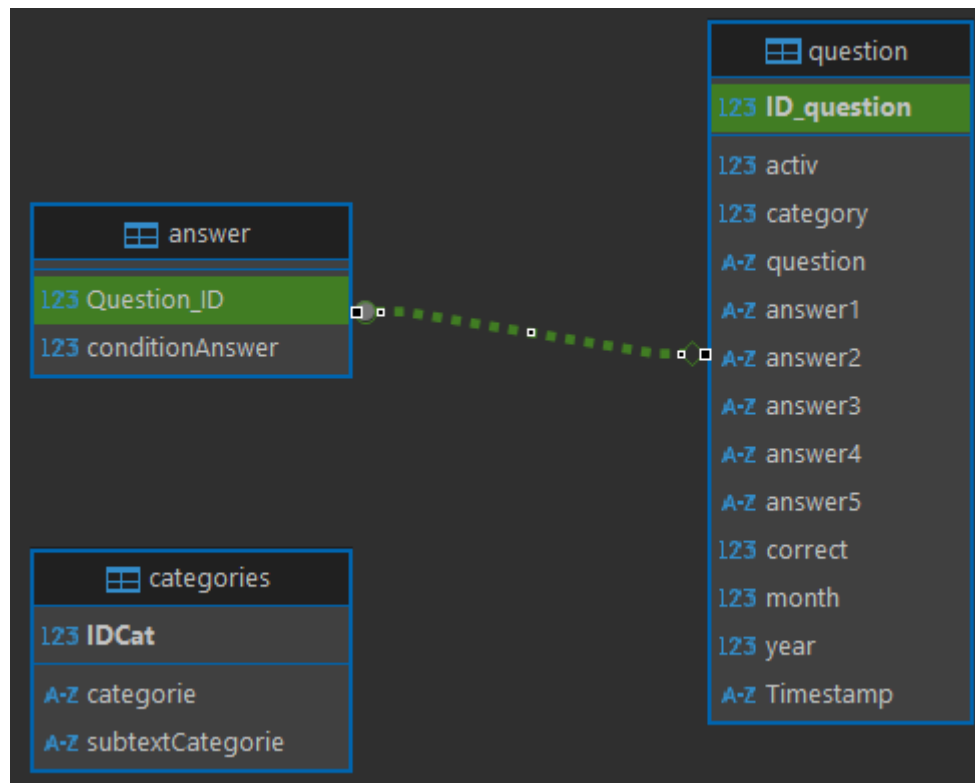
Kostenart	Menge / Dauer	Einheitspreis (€)	Gesamtkosten (€)
Personalkosten			
Auszubildender (du selbst)	80 Stunden	10,00	800,00
Ausbilder/Betreuung	5 Stunden	50,00	250,00

Sachkosten / Arbeitsmittel

Entwicklungs-PC (pauschal)	2 Wochen Nutzung	10,00 / Tag	100,00
Internet/Strom (pauschal)	2 Wochen	–	20,00
IDE / Tools (Open Source)	–	0,00	0,00
Gesamtkosten			1.170,00 €

11.Datenbank

Als Datenbank habe ich eine SQLite Datenbank gewählt. Da sie den Anforderungen an das Projekt entspricht. Sie soll dafür verwendet werden, lokale Daten zu speichern, um die Offline-Funktionalität bereitzustellen. Außerdem werden Nutzerdaten zu diesem Zeitpunkt nicht an den Server übertragen, sondern bleiben lokal auf dem Endgerät gespeichert. Es gibt eine Problematik mit der Fragen-Tabelle. Diese ist noch nicht in der 3. Normalform. Dies ist für einen späteren Zeitpunkt geplant.



12. Abschlussbericht

Ich konnte einen Teil der Software realisieren.

13. Qualitätssicherung und Tests

Ich habe manuelle Tests durchgeführt, um die einzelnen Komponenten zu testen. Zusätzlich habe ich Automatisierte Tests hinzugefügt diese sind aber noch nicht für jede Klasse vorhanden.

14. Anhang

