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

**Dokumentation zur schulischen Projektarbeit  
Thema: Entwicklung einer Koch-App mit Gamification-Aspekt im Rahmen der Vorbereitung zur Abschlussarbeit der Ausbildung**

Florian Diehle, Cornelius Müller, Niklas Soika, Jonas Wolf

FIA22B

Lernfeld 12

Andreas-Gordon-Schule Erfurt

Erfurt, 27.02.2025

**Inhaltsverzeichnis**

[1. Einleitung 4](#_Toc191556764)

[1.1 Projektbeschreibung 4](#_Toc191556765)

[1.2 Projektziel 4](#_Toc191556766)

[1.3 Projektumfeld 5](#_Toc191556767)

[1.4 Projektbegründung 5](#_Toc191556768)

[1.5 Projektschnittstellen 5](#_Toc191556769)

[1.6 Projektabgrenzung 5](#_Toc191556770)

[2 Projektplanung 6](#_Toc191556771)

[2.1 Projektphasen 6](#_Toc191556772)

[2.2 Abweichungen vom Antrag 6](#_Toc191556773)

[2.3. Diagramme 6](#_Toc191556774)

[2.3 Ressourcenplanung 7](#_Toc191556775)

[3 Analysephase 7](#_Toc191556776)

[3.1 Ist-Analyse 7](#_Toc191556777)

[3.2 Wirtschaftlichkeitsanalyse 8](#_Toc191556778)

[3.2.1 „Make or Buy“-Entscheidung 8](#_Toc191556779)

[3.2.2 Projektkosten 8](#_Toc191556780)

[3.2.3 Amortisationsdauer 9](#_Toc191556781)

[3.2.4 Nutzwertanalyse 9](#_Toc191556782)

[4 Entwurfsphase 9](#_Toc191556783)

[4.1 Zielplattform 9](#_Toc191556784)

[4.2 Entwurf der Benutzeroberfläche 10](#_Toc191556785)

[5 Projektdurchführung 11](#_Toc191556786)

[5.1 Erstellen der Mockups 11](#_Toc191556787)

[5.2 Erstellen der Datenbank 11](#_Toc191556788)

[5.3 Erstellen der API 12](#_Toc191556789)

[5.4 Testen der API 13](#_Toc191556790)

[5.5 Fehlschläge 13](#_Toc191556791)

[5.6 Frontend-Programmierung 13](#_Toc191556792)

[6 Testen der App 14](#_Toc191556793)

[6.1 Test der Anmeldefunktion 14](#_Toc191556794)

[6.2 Test der Registrierungsfunktion 14](#_Toc191556795)

[7. Projektreflexion 15](#_Toc191556796)

[8. Quellenverzeichnis 15](#_Toc191556797)

[Anhang 17](#_Toc191556798)

[9. Glossar 25](#_Toc191556799)

[Eidesstattliche Erklärung 26](#_Toc191556800)

# 

# 1. Einleitung

In der vorliegenden Projektdokumentation wird der Ablauf des Gruppenprojekts beschrieben, das im Rahmen des Berufsschulunterrichts an der Andreas-Gordon-Schule durchgeführt wurde. Das Projekt dient als Vorbereitung auf das Abschlussprojekt in der Ausbildung zum Fachinformatiker mit der Fachrichtung Anwendungsentwicklung. Ziel dieses Gruppenprojekts war es, erworbene Kenntnisse umzusetzen und zu vertiefen und gegebenenfalls neue zu erwerben.

### 1.1 Projektbeschreibung

Das Projekt besteht aus einer Android-Koch-App mit verschiedenen zusätzlichen Funktionen und einem ausgeklügelten Gamification-Aspekt, um das Kocherlebnis für weniger Motivierte zu erleichtern und den Kochprozess mit Spaß zu verbinden. Diese App setzt genau da an, indem klassische Rezeptfunktionen mit spielerischen Elementen wie dem Sammeln von Erfolgen oder dem Leveln von Zutaten kombiniert werden. Für Personen, die zum Beispiel gar nicht wissen, was überhaupt gekocht werden soll, können die Rezeptvorschläge basierend auf dem Spielerlevel Inspiration bieten. Zusätzlich kann der Kochprozess über eine Einkaufsliste und einen Barcode-Scanner vorbereitet werden.

### 1.2 Projektziel

Das Hauptziel der App und des Projekts ist es, den Kochprozess attraktiver zu gestalten, unter anderem durch die Gamification, durch Erfolge und dem Leveln von Zutaten. Um neue Zutaten freizuschalten und zu leveln werden die Nutzer ermutigt, andere Rezepte auszuprobieren und regelmäßig zu kochen. Außerdem wird dazu aufgerufen, eigene Rezepte hochzuladen, sodass andere diese ebenfalls nachkochen können. Die freigeschalteten Erfolge können die Nutzer auf ihrem eigenen Profil ansehen, was die Motivation zum Kochen fördert. Mit dem Barcode-Scanner und der Einkaufsliste können Vorbereitung für den Einkauf getroffen werden.

### 1.3 Projektumfeld

Das Projekt wird im Rahmen eines schulischen Teams mit vier Mitgliedern umgesetzt und ist betriebsunabhängig. Es gibt keinen externen Auftraggeber oder Kunden, sondern das Projekt entsteht aus Eigeninitiative. Die Zielgruppe umfasst Hobbyköche, Kochanfänger und alle, die sich für spielerisches Kochen begeistern können. Da es sich um ein Schulprojekt handelt, werden alle erforderlichen Ressourcen eigenständig bereitgestellt. Die Entwicklung erfolgt auf privaten Endgeräten mit frei zugänglichen Software-Tools und Entwicklungsumgebungen. Die Ergebnisse des Projekts werden innerhalb der Schule präsentiert.

### 1.4 Projektbegründung

Viele Menschen haben wenig Motivation oder Zeit, regelmäßig zu kochen, obwohl sie gerne neue Rezepte ausprobieren würden. Eine spielerische Herangehensweise kann helfen, den Kochprozess interessanter zu gestalten und Nutzer dazu zu ermutigen, öfter selbst zu kochen. Ebenfalls kann die App dabei helfen, mithilfe der eingebauten Filter-Funktion gesünder zu kochen. Das Leveln der Zutaten und die Achievements fürs Kochen können dabei helfen die Benutzer zu ermutigen mehr und vielfältiger zu kochen.

### 1.5 Projektschnittstellen

Da das Projekt eigenständig entwickelt wird, gibt es keine externen technischen Schnittstellen oder Anbindungen an bestehende Systeme. Die Umsetzung erfolgt in Eigenverantwortung der vier Teammitglieder ohne externe Finanzierung.

### 1.6 Projektabgrenzung

Das Projekt wird nicht der Öffentlichkeit zur Verfügung gestellt, sondern wird zuerst nur für den schulischen Zweck entwickelt.

# 2 Projektplanung

### 2.1 Projektphasen

Für das Projekt standen insgesamt 40 Stunden zur Verfügung. Diese wurden vor dem Projekt auf alle 4 Schüler gerecht aufgeteilt. Hierfür wurde das Projekt in einzelne Projektphasen geteilt.

|  |  |
| --- | --- |
| Projektphase | Geplante Zeit |
| Analysephase | 4h |
| Entwurfsphase | 10h |
| Implementierungsphase | 21h |
| Erstellung der Dokumentation | 5h |

### 

### 2.2 Abweichungen vom Antrag

Das Endprodukt weist Abweichungen zur Planung auf. Aufgrund von dem gleichzeitigen Ausfall der Gruppe in 50% der Zeit fehlen schlussendlich die folgenden Funktionen: Der Barcodescanner, das Leveling-System, die Kommunikation mit der OpenFoodFacts-API und Rezeptbilder.

### 2.3. Diagramme

Zur Planung des Projekts wurden ein Klassendiagramm, ein Relationales Datenbankmodell und ein Projektstrukturplan erstellt. Diese sind unter den Abbildungen 14 bis 16 zu finden.

### 2.3 Ressourcenplanung

Im Rahmen der Projektplanung wurde ein Überblick der verwendeten Ressourcen erstellt.

|  |  |
| --- | --- |
| Verwendete Ressourcen |  |
| Hardware | |
| Laptop | 4x |
| Android-Tablet | 1x |
| Software | |
| Visual Studio 2022 | 4x |
| MariaDB-Datenbank | 1x |
| API-Eigenentwicklung | 1x |
| GitHub zur Sicherung der Daten | 1x |
| Postman | 1x |
| Personal | |
| Azubi | 4x |

# 3 Analysephase

### 3.1 Ist-Analyse

Die Projektgruppe plant die Erstellung einer Kochapp für Androidgeräte. Im aktuellen Markt existieren viele bereits etablierte andere Kochapps mit verschiedensten Funktionen und Features. Jedoch gibt es keinerlei Apps die die Funktionen eines Barcodescanners, eines Einkaufswagens, eines Rezeptbuchs und gleichzeitig die einer spielerischen Möglichkeit vereinen, was die Idee der Gruppe beschreibt. All diese zuvor aufgelisteten Features existieren einzeln in eigenen Programmen, aber nicht in einer App die alle Aspekte vereint.

### 3.2 Wirtschaftlichkeitsanalyse

Dadurch, dass keine App auf dem Markt existiert, die die Anforderungen der Funktionen des Projektes erfüllen, ergibt sich eine wirtschaftliche Möglichkeit, die ausgenutzt werden kann. Jedoch muss gesagt werden, dass es im Bereich der Kochbuchprogramme viele langjährig etablierte Apps mit ähnlichen Funktionen gibt. Dies könnte bei einer fiktionalen Markteinführung zu erheblichen Marketingkosten führen oder sogar zum Aus des Projekts führen. Ob es sinnvoll ist diese App selbst zu entwickeln oder die Idee zu verkaufen, soll in den folgenden Abschnitten geklärt werden.

### 3.2.1 „Make or Buy“-Entscheidung

Aufgrund dessen, dass keine Projektlösung mit den Anforderungen an Funktion, Design oder Innovation existiert und das Projektteam über keinerlei reelles Budget verfügt, kann kaum von einer „Buy-Entscheidung“ gesprochen werden. Deshalb wurde entschieden das Programm in Eigenentwicklung zu erstellen, ohne andere Dienstleister mit einbeziehen zu müssen. Auch alle Softwarelösungen zur Erstellung des Projekts sind kostenlos und andere Hilfstechniken, wie zum Beispiel die OpenFoodFacts API, sind frei zugänglich und kostenfrei.

### 3.2.2 Projektkosten

Im Laufe des Projektes sind keine Kosten angefallen da Software sowie Hardware bereits vorhanden waren.

### 3.2.3 Amortisationsdauer

Die Amortisationsdauer dieses Projektes kann mit 0 angegeben werden da keine Kosten angefallen sind.

### 3.2.4 Nutzwertanalyse

Für die Bestimmung des richtigen Frameworks wurde eine Nutzwertanalyse durchgeführt.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Gewich-tung | React Native |  | Flutter |  | .NET MAUI Blazor |  |
| Erlernbarkeit | 6 | 0,5 | 3 | 0,5 | 3 | 2 | 12 |
| Performance | 8 | 4 | 32 | 5 | 40 | 4 | 32 |
| Community | 3 | 4 | 12 | 4 | 12 | 4 | 12 |
| Wartbarkeit | 5 | 3 | 15 | 4 | 20 | 4 | 20 |
| Entwicklungsge-schwindigkeit | 8 | 4 | 32 | 3 | 24 | 4 | 32 |
|  |  |  | 94 |  | 99 |  | 108 |

Basierend auf dieser Grundlage wurde sich für .NET MAUI Blazor entschieden.

# 4 Entwurfsphase

### 4.1 Zielplattform

Das Abschlussprojekt soll als mobile Anwendung mit einer zentralen und einer lokalen Datenbank realisiert werden. Für die plattformübergreifende Entwicklung wurde .NET MAUI Blazor als Framework gewählt, da es eine moderne und performante Benutzeroberfläche bietet. Dieses Framework ermöglicht eine einheitliche Entwicklung für verschiedene Plattformen, wobei sich das Projekt zunächst auf Android-Geräte konzentriert. Als zentrale Datenbank wird MySQL eingesetzt, da sie sich durch hohe Stabilität und Skalierbarkeit auszeichnet. Zusätzlich wird eine lokale Datenbank implementiert, um Offline-Zugriffe und eine nahtlose Nutzererfahrung zu gewährleisten. Die Kommunikation zwischen der App und der zentralen Datenbank erfolgt über eine eigens entwickelte API, die eine effiziente und sichere Datenübertragung ermöglicht.

Die Wahl der Technologien erfolgte auf Basis der Anforderungen an Skalierbarkeit, Performance und einfache Wartbarkeit. .NET MAUI wurde aufgrund seiner plattformübergreifenden Entwicklungsmöglichkeiten gewählt, während MySQL als bewährtes relationales Datenbanksystem dient. Die Kombination einer zentralen und einer lokalen Datenbank gewährleistet eine flexible Nutzung, auch ohne durchgehende Internetverbindung. Zukünftig kann die App auf weitere Plattformen wie iOS ausgeweitet werden.

### 4.2 Entwurf der Benutzeroberfläche

Die Benutzeroberfläche unserer Koch-App wurde mit dem Ziel entwickelt, eine intuitive und ansprechende Nutzererfahrung zu bieten. Dabei wurde auf eine klare Strukturierung der Inhalte und eine einfache Bedienbarkeit geachtet. Um die Navigation für die Nutzer möglichst komfortabel zu gestalten, wurde eine mobile Benutzeroberfläche mit einer modernen grafischen Gestaltung (GUI) gewählt. Für die visuelle Gestaltung wurden mithilfe von Draw.io Mockups erstellt, die das Layout der Anwendung veranschaulichen.

Die Startseite enthält einen Anmelde- und Registrierungsbereich, über den sich Nutzer in die App einloggen können. Nach der Anmeldung gelangt man auf die Hauptseite, die die wichtigsten Funktionen der App bereitstellt. Die Navigation erfolgt über eine seitliche Menüführung, die Zugriff auf zentrale Bereiche wie das Kochbuch, die Einkaufsliste, Erfolge und Einstellungen ermöglicht. Die Hauptansicht bietet eine Rezeptübersicht mit Suchfunktion und personalisierten Empfehlungen. Im Benutzerprofil können Nutzer ihren Fortschritt verfolgen, gesammelte Erfolge einsehen und ihre Zutaten verwalten. Das Kochbuch dient zur Speicherung von Rezepten und ermöglicht das Hinzufügen neuer Gerichte. In der Einkaufsliste lassen sich benötigte Zutaten verwalten und nach Kategorien sortieren. In den Einstellungen können Nutzer das Design der App anpassen und die Sprache ändern. Das Farbschema setzt sich aus einem dunklen Hintergrund mit grünen Akzenten zusammen, um eine moderne und ansprechende Optik zu gewährleisten. Die Benutzeroberfläche wurde so gestaltet, dass sie sowohl für Kochanfänger als auch für erfahrene Hobbyköche leicht verständlich ist. Klare visuelle Hierarchien und eine einheitliche Menüstruktur sorgen für eine intuitive Nutzung. Große Buttons und Icons erleichtern die Bedienung auf Touchscreens, während eine kontrastreiche Farbgestaltung die Lesbarkeit verbessert.

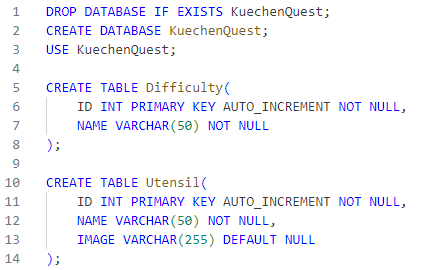
# 5 Projektdurchführung

### 5.1 Erstellen der Mockups

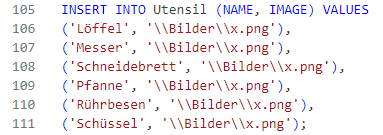
Die Mockups wurden gemeinsam in der Gruppe erstellt, sie sind unter Abbildung 1 bis 13 zu finden.

### 5.2 Erstellen der Datenbank

Mit dem Wissen aus vorherigen Projekten wurde ein Installations-Skript für die Datenbank erstellt. Dieses Skript beinhaltet die SQL-Anweisungen zur Erstellung der Datenbank sowie aller Tabellen, außerdem werden bereits Testdatensätze eingefügt, um ein Arbeiten mit der Datenbank direkt zu ermöglichen.



In der Abbildung sieht man den Start des Skriptes. Falls eine Datenbank mit dem Namen “KuechenQuest” vorhanden sein sollte wird diese gelöscht, deswegen sollte man dieses Skript nur ausführen, wenn man grundlegende Änderungen an den Tabellen vornehmen will. Nach der Erstellung der Datenbank wird in den neu erstellten Kontext gewechselt, wo anschließend alle Tabellen angelegt werden. Testdaten werden am Ende des Skriptes eingefügt, wie man an diesem Beispiel sehen kann.



### 

### 5.3 Erstellen der API

Zur Programmierung der API wurde die Programmiersprache C# genutzt, da der Großteil der Gruppe bereits Erfahrung damit hat.

Die API besteht aus 4 Controllern die Anfragen bearbeiten:

* Benutzer-Controller (UserController)
* Rezept-Controller (RecipeController)
* Hilfsmittel-Controller (UtensilController)
* Zutaten-Controller (IngredientController)

Die Controller haben jeweils Endpoints, welche die CRUD-Anfragen bearbeiten. Der Datenbankzugriff der Controller erfolgt über eine Selbstentwickelte Database-Klasse, in dieser Klasse sind alle SQLs gespeichert und dort erfolgt auch die gesamte Fehlerbehandlung.



In der obigen Abbildung kann man exemplarisch den Login-Endoint des Benutzer-Controllers sehen. Die Klasse *MyLoginRequest* ist eine Kapselklasse, die den Benutzernamen und das Passwort enthält. Zunächst wird überprüft, ob der Request nicht leer ist. Anschließend wird die Login-Funktion der Datenbank aufgerufen. Bei erfolgreicher Anmeldung wird der Benutzer zurückgegeben, andernfalls wird *null* zurückgegeben. Auf Basis des Rückgabewerts wird dann ein entsprechender HTTP-Response-Code generiert, der an den Client gesendet wird.

### 5.4 Testen der API

Die API wurde bereits während der Entwicklungsphase mithilfe der Software Postman umfassend getestet, um ihre Funktionalität und Korrektheit sicherzustellen.

### 5.5 Fehlschläge

Die Kommunikation mit der OpenFoodFacts API konnte aufgrund von Krankheit leider nicht mehr implementiert werden.

### 5.6 Frontend-Programmierung

In der Planungsphase des Projektes wurde diskutiert, welche Technologie für die App verwendet werden sollte, dazu wurde auch eine Nutzwertanalyse erstellt (vgl. 3.1.4 Nutzwertanalyse). Schlussendlich wurde sich dazu entschieden, .NET MAUI Blazor zu verwenden.

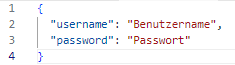
Die Frontend-Programmierung begann mit dem Erstellen der Login-Seite, da diese als Standardseite fungiert, wenn es zu Fehlern in der App kommt. Weiterführenden Seiten wie z.B. die Startseite oder die Profilseite wurden danach von einzelnen Teammitgliedern parallel entwickelt.

# 6 Testen der App

### 6.1 Test der Anmeldefunktion

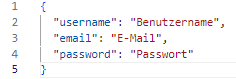
Die Login-Funktion wurde während der Entwicklung des Back- und Frontends zweimal unabhängig voneinander getestet. Im Vorfeld der Entwicklung wurde ein Kommunikationsformat vereinbart (siehe nachfolgende Abbildung).

In Zukunft ist die Übermittlung des Passwortes in gehashter Form geplant um die Sicherheit der Userdaten weiter zu erhöhen.



### 6.2 Test der Registrierungsfunktion

Die Registrierungsfunktion ähnelt der Login-Funktion in ihrer Funktionsweise. Beim Registrieren eines Benutzers muss hier zusätzlich zu Benutzername und Passwort eine einzigartige E-Mail vergeben werden.



# 7. Projektreflexion

Die Planung des Projektes innerhalb des Projektteams verlief gut, allerdings war der selbstgesteckte Zeitplan etwas zu gering bemessen. Der erste Schritt der Projektdurchführung war das Erstellen der Mockups. Währenddessen konnte sich bereits ein Teammitglied mit der Erstellung des Backends befassen, ohne den Prozess zu verlangsamen. Für die Durchführungsphase hat sich das Projektteam in zwei kleinere Teams unterteilt (Frontend, Backend). Diese Aufteilung hat sehr gut funktioniert, da das Frontend Team bereits viele Funktionen des Backends benutzen konnte, obwohl die Entwicklung noch nicht abgeschlossen war. Aufgrund eines Komplettausfalls des gesamten Teams in der zweiten Hälfte des Projektzeitraums konnten nicht alle geplanten Funktionen umgesetzt werden. Durch eine außerplanmäßige Verlängerung des Projektzeitraumes konnte dennoch ein vorzeigbares Produkt entwickelt werden. Abschließend lässt sich sagen, dass die geplanten Funktionen den individuellen Fähigkeiten des Projektteams entsprechen und die Implementierung aller Funktionen bei ausreichender Zeit möglich gewesen wären.

# 8. Quellenverzeichnis

Für das Projekt wurden ausschließlich Dokumente aus dem Unterricht sowie eigenes Wissen verwendet.9. Abbildungsverzeichnis

[Abbildung 1 - Mockup der Login-Seite 17](#_Toc191554670)

[Abbildung 2 - Mockup der Registrierungsseite 17](#_Toc191554671)

[Abbildung 3 - Mockup der Rezeptübersicht 18](#_Toc191554672)

[Abbildung 4 - Mockup der Navigations-Sidebar 18](#_Toc191554673)

[Abbildung 5 - Mockup der Profilseite 19](#_Toc191554674)

[Abbildung 6 - Mockup der Erfolgsübersicht 19](#_Toc191554675)

[Abbildung 7 - Mockup der Zutatenlevel 20](#_Toc191554676)

[Abbildung 8 - Mockup der Einkaufsliste 20](#_Toc191554677)

[Abbildung 9 - Mockup der Kochbuchseite 21](#_Toc191554678)

[Abbildung 10 - Mockup der Rezepterstellung 21](#_Toc191554679)

[Abbildung 11 - Mockup des Barcode-Scanners 22](#_Toc191554680)

[Abbildung 12 - Mockup der Einstellungsseite 22](#_Toc191554681)

[Abbildung 13 - Mockup des Ausloggen-Buttons 22](#_Toc191554682)

[Abbildung 14 - Klassendiagramm des Projekts 23](#_Toc191554683)

[Abbildung 15 - Relationales Datenbankmodell des Projekts 23](#_Toc191554684)

[Abbildung 16 - Projektstrukturplan des Projekts 24](#_Toc191554685)

# Anhang



Abbildung 1 - Mockup der Login-Seite

Abbildung 2 - Mockup der Registrierungsseite

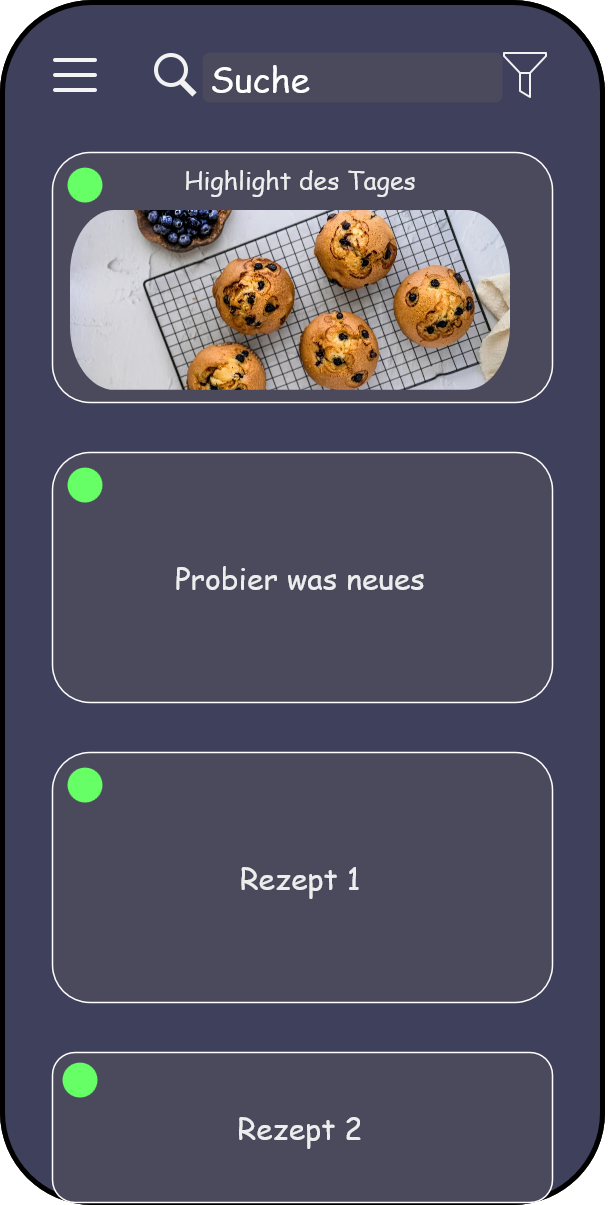


Abbildung 3 - Mockup der Rezeptübersicht

Abbildung 4 - Mockup der Navigations-Sidebar

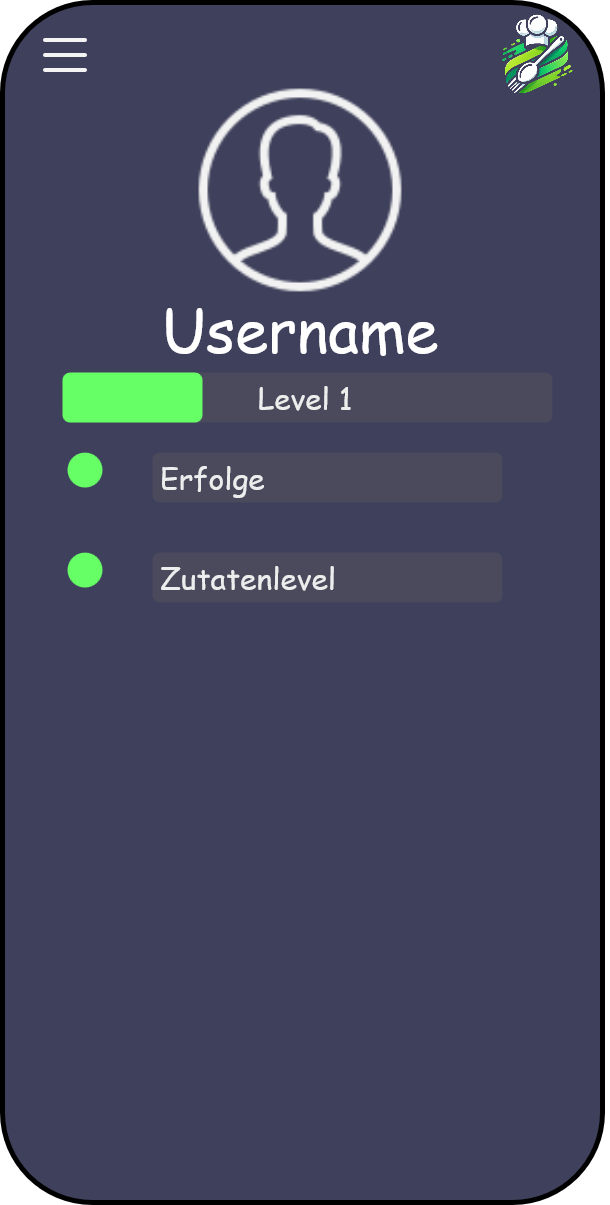


Abbildung 5 - Mockup der Profilseite

Abbildung 6 - Mockup der Erfolgsübersicht

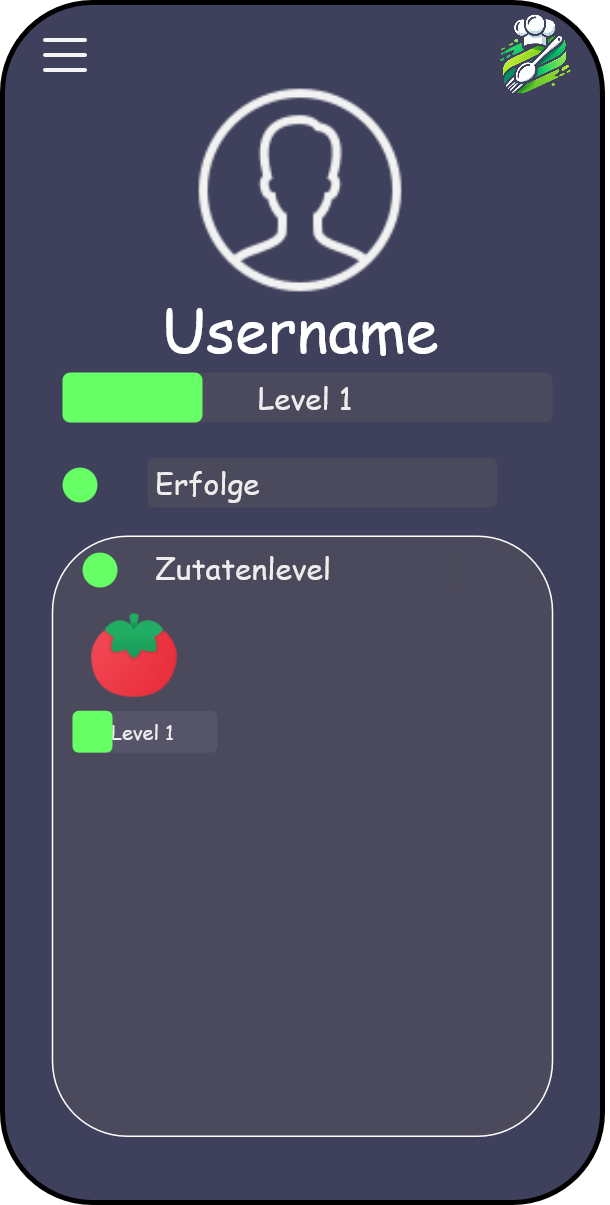


Abbildung 7 - Mockup der Zutatenlevel

Abbildung 8 - Mockup der Einkaufsliste

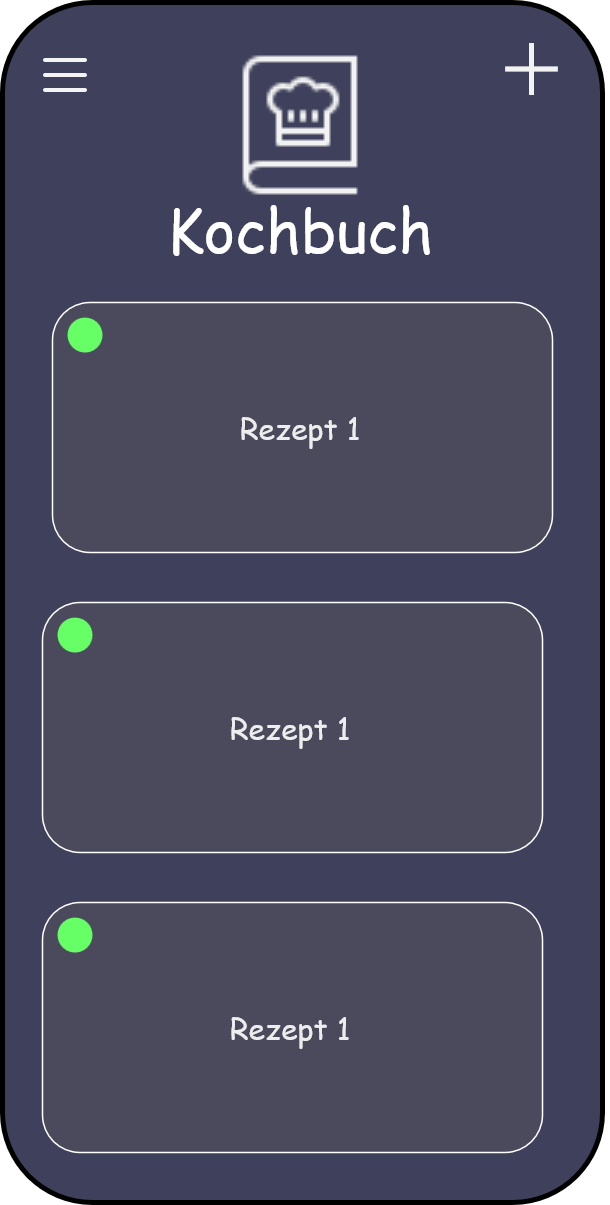
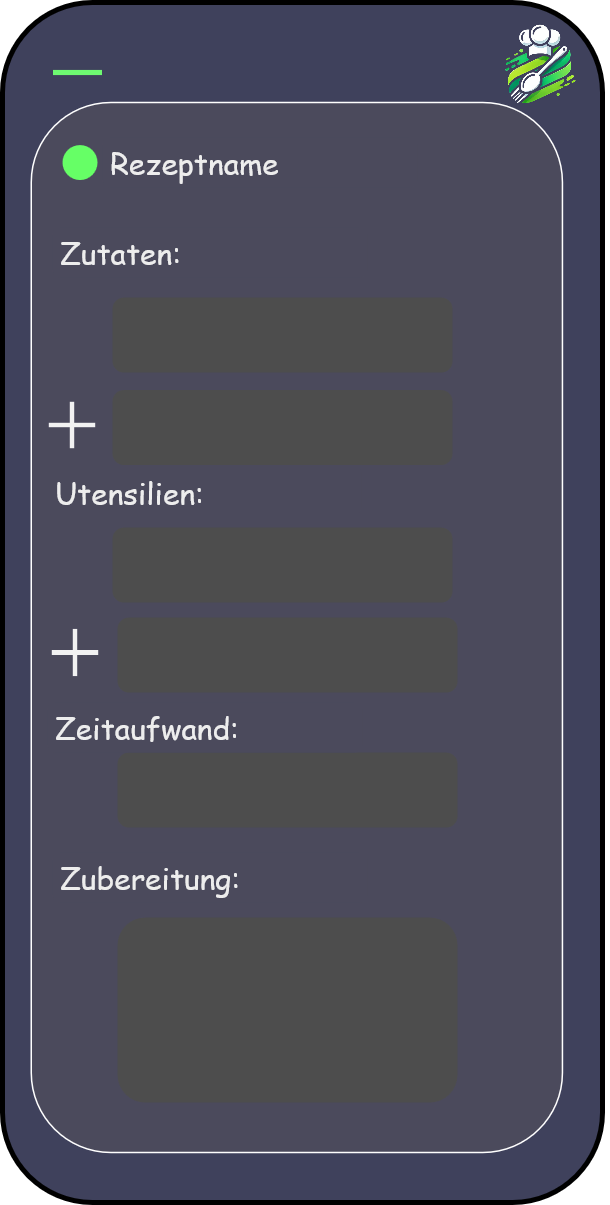


Abbildung 9 - Mockup der Kochbuchseite

Abbildung 10 - Mockup der Rezepterstellung

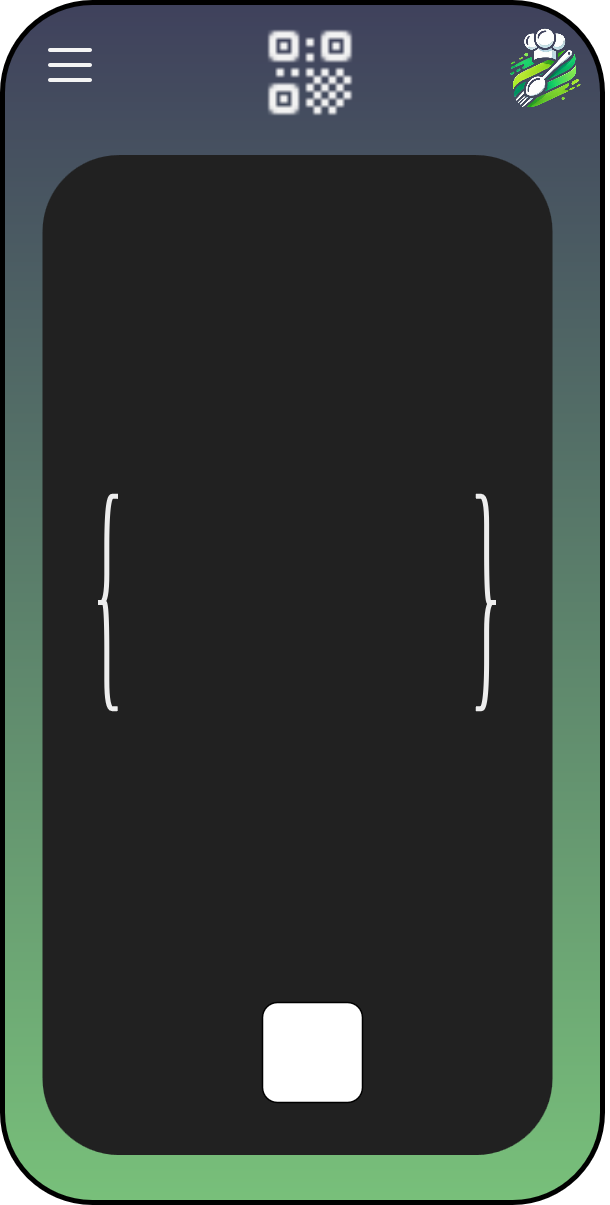
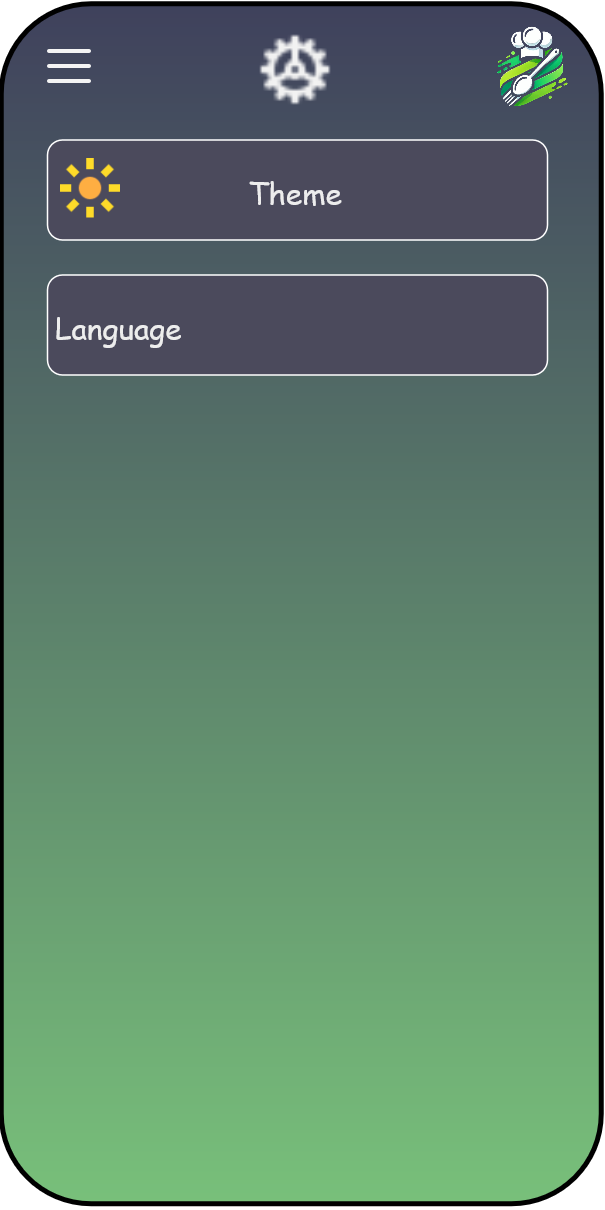


Abbildung 11 - Mockup des Barcode-Scanners

Abbildung 12 - Mockup der Einstellungsseite

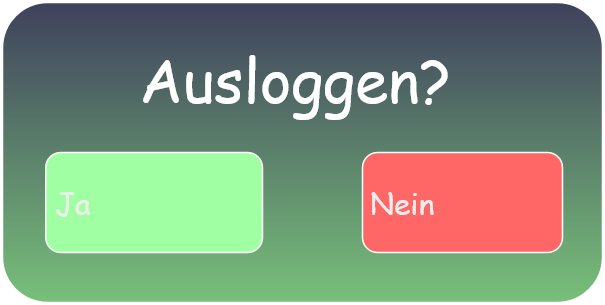


Abbildung 13 - Mockup des Ausloggen-Buttons

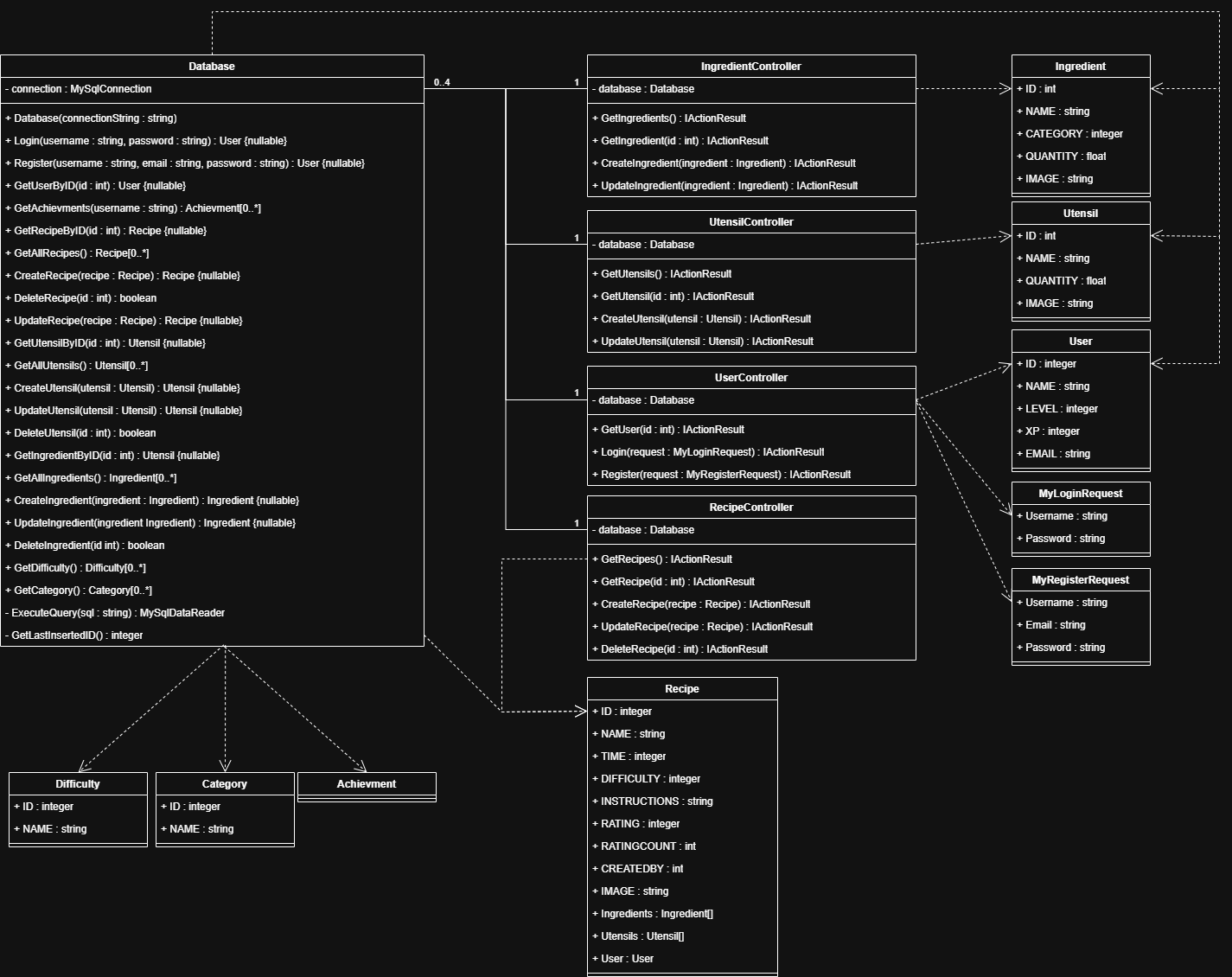


Abbildung 14 - Klassendiagramm des Projekts

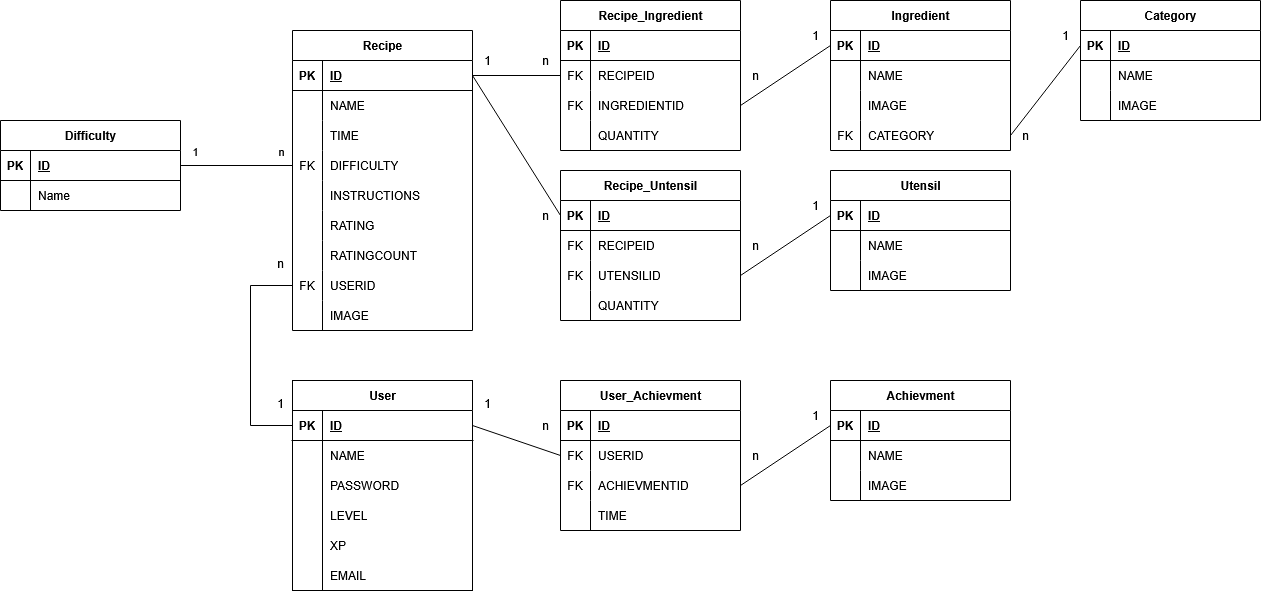


Abbildung 15 - Relationales Datenbankmodell des Projekts

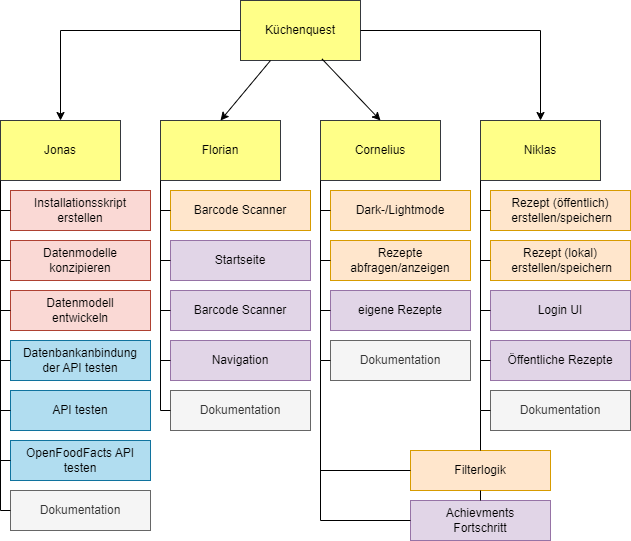


Abbildung 16 - Projektstrukturplan des Projekts

# 9. Glossar

|  |  |
| --- | --- |
| **API**  **Backend**  **CRUD**  **Frontend**  **Gamification**  **GitHub**  **GUI**  **Postman**  **Visual Studio 2022**  **.NET Maui Blazor** | Application Programming Interface  Systemseitige Programmierungsebene der Anwendung  Grundoperatoren einer API; Create, Read, Update und Delete  Nutzerseitige Programmierungsebene der Anwendung  Die Einbindung von videospieltypischen Elementen in einem spielfremden Kontext  Plattform zur Codeverwaltung und Synchronisierung  Graphical User Interface  Open-Source-Anwendung zum Senden von API-Requests  Integrierte Entwicklungsumgebung für C# und andere Programmiersprachen  Framework zur Entwicklung der Android-App |
|  |  |

# Eidesstattliche Erklärung

Wir versichern an Eides statt, dass wir die Arbeit selbständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt haben. Alle Stellen, die wörtlich oder sinngemäß aus Veröffentlichungen entnommen sind, haben wir als solche gekennzeichnet.

Datum, Unterschrift