Obrázok, na ktorom je text, písmo, snímka obrazovky, grafika

Automaticky generovaný popis

Semestrálna práca z predmetu VAMZ

**Quizzify**

**Autor:** Šimon Lušňák

**Štúdijná skupina:** 5ZYI22

# Popis a analýza riešeného problému

## Špecifikácia zadania a definovanie problému

Aplikácia *"Quizzify"* je jednoduchá aplikácia pre Android zariadenia, ktorá poskytuje užívateľom možnosť hrať rôzne typy kvízov. Tento projekt sa zameriava na poskytnutie zábavy a vzdelávania prostredníctvom interaktívnych kvízov z rôznych oblastí.

## Podobné aplikácie a ich rozdiely

**QuizzLand. Quiz & Trivia game**: QuizzLand. Quiz & Trivia game je populárna aplikácia na mobilné zariadenia, ktorá ponúka širokú škálu kvízov z rôznych tém. Oproti nej sa aplikácia *"Quizzify"* zameriava na obmedzený počet kategórií kvízov a zameriava sa na jednoduché a zábavné otázky.

Obrázok, na ktorom je text, snímka obrazovky, animák

Automaticky generovaný popis

**Trivia Crack**: Trivia Crack je ďalšia populárna kvízová aplikácia s rôznymi kategóriami otázok. Na rozdiel od nej, *"Quizzify"* ponúka menej kategórií, ale sa zameriava na zlepšenie vedomostí v konkrétnych oblastiach.

# Návrh riešenia problému

## Krátka analýza

Aplikácia "Quizzify" sa bude sústrediť na poskytnutie zábavy a vzdelávania pomocou kvízov. Každá kategória bude zahŕňať otázky z určitej oblasti, čo umožní užívateľom zlepšiť svoje vedomosti a zároveň sa zabaviť.

## Návrh aplikácie

Aplikácia bude mať jednoduché a intuitívne používateľské rozhranie.

Na úvodnej obrazovke budú tlačidlá “Sign In” a “Register” , ktoré budú slúžiť na registráciu alebo prihlásenie sa použivateľa. Po úspešnom prihlásení sa používateľ dostane na hlavnú obrazovku, kde má možnosť zobraziť svoj profil v ktorom si môže zmeniť používateľské meno alebo heslo, taktiež tam je zobrazené jeho hodnotenie. Ďalej môže používateľ z hlavnej obrazovky prejsť na rebríček top 100 používateľov, ktorý umožnuje aj následné zobrazenie detailov o používateľovi z rebríčka.

Tiež je možné zmeniť farebnú schému aplikácie po navigovaní z hlavnej obrazovky do nastavení.

Po kliknutí na tlačidlo „Play“ na hlavnej obrazovke sa používateľ dostane na obrazovku s kategóriami kvízov, kde si môže zvoliť jednu z kategórii alebo aj resetovať svoj pokrok v danej kategórii.

Po zvolení kategórie sa používateľ dostáva na obrazovku s otázkami z danej kategórie na ktoré odpovedá stlačením jedného z tlačidiel s odpoveďami.

Obrázok, na ktorom je text, snímka obrazovky, diagram, plán

Automaticky generovaný popis

# Popis implementácie

Použil som architektúru MVVM (Model-View-ViewModel) pri implementácii aplikácie. Táto architektúra zabezpečuje oddelenie logiky obchodného modelu (Model), prezentačnej vrstvy (View) a komponentov, ktoré spravujú tok dát a biznis logiku (ViewModel). Každá časť má svoju špecifickú úlohu:

Model: Reprezentuje údaje a obchodnú logiku aplikácie. V tvojom prípade sú to databázové triedy ako User, Category, Question, Answer, atď.

View: Je zodpovedná za zobrazovanie údajov používateľovi. V tvojom prípade by to mohli byť rôzne obrazovky, ktoré zobrazujú informácie o používateľovi, kategóriách, otázkach a odpovediach.

ViewModel: Obsahuje logiku, ktorá je nezávislá na konkrétnej implementácii používateľského rozhrania. Zodpovedá za správu údajov a komunikáciu medzi modelom a zobrazením. V tvojom prípade to sú triedy ako UserViewModel, CategoryViewModel, QuestionViewModel, atď.

MVVM architektúra umožňuje oddeliť prezentačnú logiku od zobrazovacej logiky, čo zjednodušuje testovanie, znovupoužiteľnosť a údržbu kódu.

**Obrazovky**

**InitialScreen**

Popis: Táto obrazovka je vstupným bodom aplikácie. Zobrazuje logo aplikácie a tlačidlá na prihlásenie alebo registráciu.

**SignInScreen**

Popis: Táto obrazovka umožňuje užívateľovi prihlásiť sa pomocou užívateľského mena a hesla. V prípade úspešného prihlásenia sa používateľ presunie na hlavnú obrazovku.

**RegisterScreen**

Popis: Na tejto obrazovke noví používatelia môžu vytvoriť účet vyplnením formulára s užívateľským menom a heslom.

**MainMenuScreen**

Popis: Hlavná obrazovka aplikácie zobrazuje hlavné možnosti ako začať hru, zobraziť rebríček alebo upraviť nastavenia.

**LeaderboardScreen**

Popis: Táto obrazovka zobrazuje rebríček najlepších hráčov. Používateľ môže vybrať iného hráča a zobraziť jeho profil.

**ProfileScreen**

Popis: Profilová obrazovka zobrazuje informácie o používateľovi ako je užívateľské meno a hodnotenie, ako aj možnosť zmeny hesla alebo mena.

**ChangePasswordScreen**

Popis: Umožňuje používateľovi zmeniť svoje heslo.

**ChangeUsernameScreen**

Popis: Umožňuje používateľovi zmeniť svoje užívateľské meno.

**SettingsScreen**

Popis: Obrazovka nastavení umožňuje používateľovi zmeniť niektoré nastavenia aplikácie, ako je napríklad tmavý režim alebo notifikácie.

**UserDetailsScreen**

Popis: Táto obrazovka zobrazuje podrobnosti o konkrétnom používateľovi, ako je jeho užívateľské meno a hodnotenie.

**QuestionScreen**

Popis: Obrazovka s otázkou zobrazuje otázku a možnosti odpovedí. Používateľ môže vybrať jednu z možností a pokračovať ďalšou otázkou.

**Funkcie**

**performSignIn**

Popis: Funkcia, ktorá vykonáva proces prihlásenia používateľa pomocou zadaného užívateľského mena a hesla.

**performRegistration**

Popis: Funkcia, ktorá vykonáva proces registrácie nového používateľa s užívateľským menom a heslom.

**changeUsername**

Popis: Funkcia, ktorá umožňuje zmenu užívateľského mena používateľa.

**changePassword**

Popis: Funkcia, ktorá umožňuje zmenu hesla používateľa.

**Implementácia databázy**

Entita **Answer**

Answer je tabuľka v databáze reprezentujúca odpovede na otázky.

Obsahuje stĺpce:

answerId: Unikátny identifikátor odpovede (primárny kľúč).

questionId: Identifikátor otázky, na ktorú je odpoveď zodpovedaná (cudzí kľúč).

answerText: Text odpovede.

isCorrect: Určuje, či je odpoveď správna.

DAO rozhranie **AnswerDao**

Obsahuje metódy pre prístup k údajom v tabuľke Answer.

Metódy zahŕňajú vkladanie nových odpovedí, získavanie odpovedí podľa identifikátora otázky a získavanie všetkých odpovedí synchronne.

Repozitár **AnswerRepository**

Zodpovedá za správu údajov Answer a poskytuje prístup k nim.

Obsahuje metódy na získanie odpovedí podľa identifikátora otázky a vloženie nových odpovedí.

Entita **Category**

Category je tabuľka v databáze reprezentujúca kategórie otázok.

Obsahuje stĺpce:

categoryId: Unikátny identifikátor kategórie (primárny kľúč).

categoryName: Názov kategórie.

description: Popis kategórie.

imageId: Identifikátor obrázku pre kategóriu.

DAO rozhranie **CategoryDao**

Obsahuje metódy pre prístup k údajom v tabuľke Category.

Metódy zahŕňajú vkladanie nových kategórií a získavanie všetkých kategórií synchronne.

Repozitár **CategoryRepository**

Zodpovedá za správu údajov Category a poskytuje prístup k nim.

Obsahuje metódy na získanie všetkých kategórií a vloženie nových kategórií.

**User**

Reprezentuje používateľa aplikácie s atribútmi ako userId, username, password, rating, darkMode, atď.

**UserDao**

Rozhranie definujúce prístup k databáze pre operácie týkajúce sa používateľov ako vkladanie, autentifikácia, aktualizácia a získanie používateľov.

**UserRepository**

Obsahuje metódy na manipuláciu s údajmi o používateľoch.

Implementuje logiku pre vkladanie, autentifikáciu, aktualizáciu a získanie používateľov z databázy.

**UserProgress**

Reprezentuje postup používateľa pri riešení otázok, vrátane údajov ako progressId, userId, questionId a isAnswered.

**UserProgressDao**

Rozhranie definujúce prístup k databáze pre operácie týkajúce sa pokroku používateľa, ako vkladanie, aktualizácia a získanie informácií o pokroku.

**UserProgressRepository**

Implementuje logiku pre manipuláciu s údajmi o pokroku používateľa, ako vkladanie, aktualizáciu a získanie informácií o pokroku.

Databázová trieda **AppDatabase**

Obsahuje definície databázových tabuliek a definície DAO rozhraní.

Poskytuje statickú metódu *getDatabase()*, ktorá vráti inštanciu databázy. Inštancia databázy je inicializovaná pomocou návrhového vzoru ***Singleton***.

**Implementácia ViewModelov**

**UserViewModel**

Slúži na riadenie údajov používateľa, ako je prihlasovanie, registrácia, získavanie a aktualizácia používateľských informácií.

Obsahuje metódy na overenie používateľa, registráciu používateľa, získavanie používateľa podľa používateľského mena, aktualizáciu používateľského mena, aktualizáciu hesla a aktualizáciu nastavení tmavej témy a notifikácií.

**QuestionViewModel**

Riadi prístup k údajom týkajúcim sa otázok a odpovedí.

Obsahuje metódy na získavanie otázok podľa identifikátora kategórie a získavanie odpovedí na otázky.

**CategoryViewModel**

Slúži na správu kategórií otázok.

Poskytuje metódu na získanie všetkých kategórií a metódu na získavanie počtu otázok v danej kategórii.

**UserProgressViewModel**

Jeho úlohou je riadiť prístup k údajom týkajúcim sa pokroku používateľa v hre.

Obsahuje metódy na aktualizáciu pokroku používateľa, resetovanie pokroku používateľa v kategórii a inicializáciu pokroku používateľa.

**ThemeViewModel**

Jeho úlohou je riadiť nastavenia témy v aplikácii.

Obsahuje metódu na nastavenie tmavej témy.

# Zoznam použitých zdrojov

<https://developer.android.com/>