

Semestrálna práca z predmetu *VAMZ*

Aplikácia milionár

**vypracoval:** Roman Kubík

**študijná skupina:** 5ZYI23

**študijný program:** informatika

**cvičiaci:** doc. Ing. Patrik Hrkút PhD.

**termín:** pondelok blok 8 - 10 v Žiline dňa 11.06. 2024

Úvod

1. Prehľad dostupných aplikácií

**Milionář – Český kvíz**

* Vyhodnocovanie výberu otázok
* Selekcia otázok vzhľadom na cenovú kategóriu
* Nepovoľuje výber témy
* Bezplatná aplikácia

Obrázok, na ktorom je text, snímka obrazovky, písmo, elektrická modrá

Automaticky generovaný popis

**Milionář 2024 český kvíz**

* Poskytuje nové a aktualizované otázky
* Za získané peniaze si hráč vie nakúpiť fiktívne nehnuteľnosti a investovať peniaze
* Aplikácia ponúka voľbu obtiažnosti kvízu
* Aplikácia zobrazuje rebríček získaných výhier v porovnaní s ostatnými užívateľmi

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

Automaticky generovaný popis

**Official Millionaire game**

* Poskytuje viac ako 4 možnosti odpovede
* Dostupný časovač odpovedí
* Všetky tri známe možnosti nápovedy / 50:50, publikum, priateľ na telefóne

Obrázok, na ktorom je text, snímka obrazovky, elektrická modrá

Automaticky generovaný popis

Podobných aplikácií je na Google Play Store je mnoho, vybrané aplikácie som analyzoval vzhľadom na podobnosť s vyššie spomenutou televíznou reláciou.

Predstavenie semestrálnej práce

Mojou semestrálnou prácou bolo vypracovanie aplikácie na motív „Ako sa stať milionárom“ v jazyku Kotlin. V hre odpovedáte na jednu zo štyroch odpovedí na otázku ktorá sa vám náhodne vyberá podľa vašich vopred zvolených parametrov. Vopred zvolenými parametrami rozumieme jednu z 5 (resp. 6) tém, a obtiažnosť otázky. Na každú otázku máte čas odpovedať 15 sekúnd, čo indikuje animovaná časomiera na vrchu kvízovej hry. Za každú správnu otázku sa vám pripočítava skóre. Hra končí v momente, keď odpoviete na otázku nesprávne, alebo keď vám vyprší čas. Hra má implementovaný rebríček, do ktorého môžete zapísať vaše výsledky po konci hry.

Návrh riešenia problému

Diagram tried

Obrázok, na ktorom je diagram, text, plán, rad

Automaticky generovaný popis

Diagram aktivít

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

Automaticky generovaný popis

Popis implementácie

Aplikácia milionár je implementovaná pomocou Jetpack Compose UI, navigácie pomocou NavHostController, lokálnej databázy otázok a skóre pomocou .json súboru, ktorý tvoria data classy Otazky, Score. O celú logiku hry sa stará funkcia MilionarApp, ktorá ako parametre dostáva vopred vytvorené ViewModely jednotlivých obrazoviek. Tieto ViewModely sú typu ThemeSelectionViewModel (od obrazovky na výberu témy otázok), GameViewModel (herná obrazovka), DifficultyViewModel(výber obtiažnosti v menu), ScoreboardViewModel(obrazovka rebríčku desiatich najvyšších skóre) a vyššie spomínaný NavHostController. Tieto ViewModely vytváram v MainActivity pri spustení aplikácie.V tejto funkcií MilionarApp najmä kontrolujem ktorá obrazovka má byť zobrazená NavHostom.

Obrázok, na ktorom je text, snímka obrazovky, softvér

Automaticky generovaný popis

Aplikácia sa spustí do MainMenu. O zvolenie a zapamätanie si zvolenej obtiažnosti sa na tejto obrazovke stará DifficultyViewModel. MainMenu je vďaka tomu odolné na zmenu polohy z výšky na šírku a naopak. Všetky obrazovky su taktiež scrollable pri zmene polohy.

Obrázok, na ktorom je text, snímka obrazovky, písmo

Automaticky generovaný popis

Vrámci načítavania skóre som zvolil spôsob vytvorenia lokálneho scores.json súboru v telefóne a opätovné načítavanie z neho. Kvôli tomuto som musel posúvať context do konštruktora ScoreManagera, aby sme vedeli spätne čítať dáta v context.filesDir. Hráč taktiež v obrazovke ScoreboardScreen dokáže celý obsah scores.json vymazať, stlačením tlačidla Reset, a premazať tak všetky výsledky.

Obrázok, na ktorom je text, snímka obrazovky, písmo

Automaticky generovaný popis Obrázok, na ktorom je text, snímka obrazovky, písmo

Automaticky generovaný popis



Trieda ScoreManager taktiež posiela notifikáciu, ak sme uložili score, ktoré má vrámci rebríčka najvyšší počet bodov. Toto skóre zakaždým zisťujeme už pri načítaní aplikácie, nie je to najvyššie skóre od spustenia aplikácie.

Obrázok, na ktorom je text, snímka obrazovky, písmo

Automaticky generovaný popis

Pre posielanie notifikácií som si vytvoril triedu NotificationHelper podľa Google Codelabs

Obrázok, na ktorom je text, snímka obrazovky, softvér

Automaticky generovaný popis

Načítavanie otázok som tiež riešil vstupným súborom otazky.json ktorý je zahrnutý v assetoch projektu. Otázky sa filtrujú podľa zvolenej obtiažnosti a témy, ktoré sú prednastavené na Random pri spustení aplikácie.

Obrázok, na ktorom je text, písmo, snímka obrazovky

Automaticky generovaný popis

Po zvolení témy a obtiažnosti, hráč púšťa hru, v ktorej mu v náhodnom poradí prichádzajú otázky zvolených parametrov. Na každú otázku je čas 15 sekúnd, to indikuje LinearProgressIndicator, ktorý sa obnovuje každú sekundu pomocou posúvania premennej animationProgress z výpočtu v LaunchedEffect composable.

Obrázok, na ktorom je text, snímka obrazovky, softvér, multimediálny softvér

Automaticky generovaný popisObrázok, na ktorom je text, písmo, snímka obrazovky, dizajn

Automaticky generovaný popis

Zoznam zdrojov

<https://developer.android.com/reference/androidx/navigation/NavHostController>

<https://developer.android.com/develop/ui/compose/side-effects>

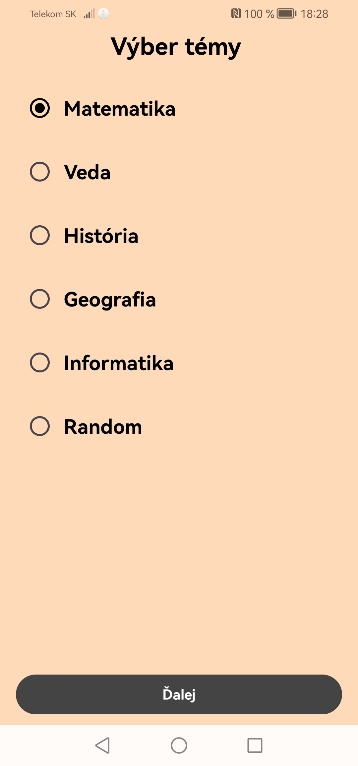
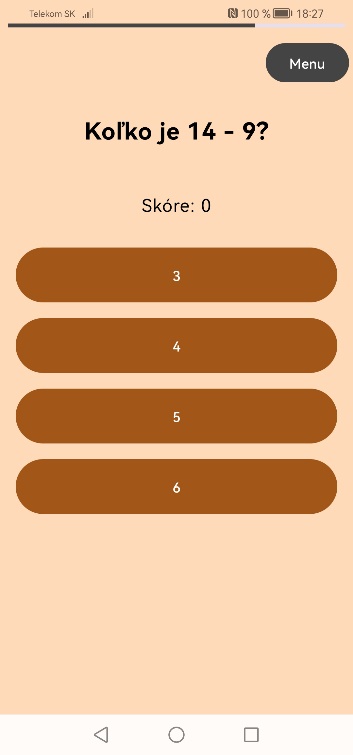
<https://kotlinandroid.org/kotlin/kotlin-read-json-file/>

<https://developer.android.com/develop/ui/views/notifications/build-notification>

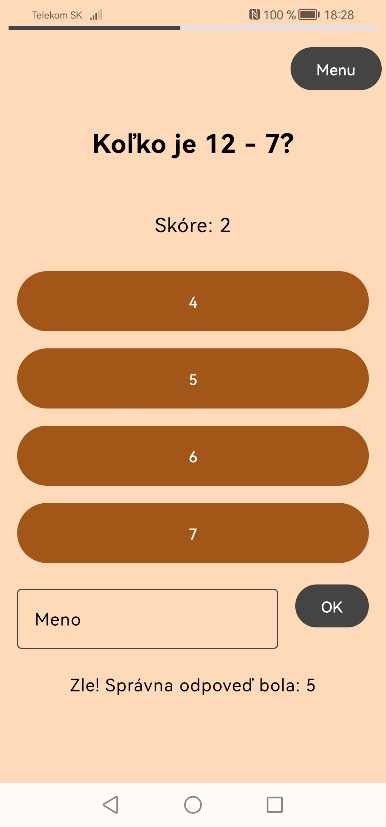
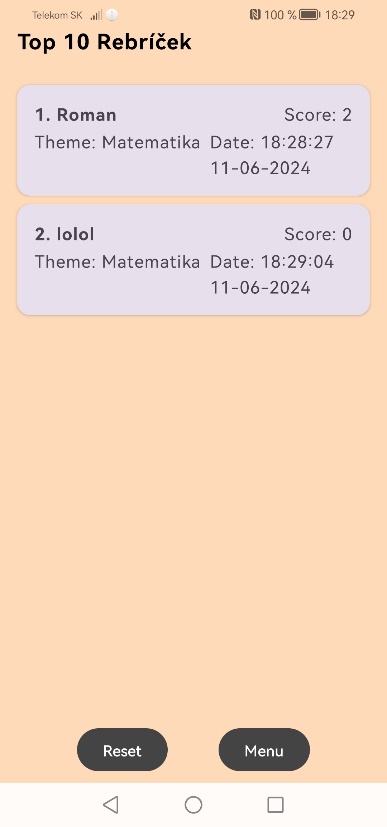
<https://google.com>/

https://stackoverflow.com/

Ukážka aplikácie

Hlavné menu Voľba témy Samotný screen kvízu

Screen po zlej odpovedi Ukážka notifikácie o highscore Prehľad rebríčka