**Word Hunt**

**Popis a analýza riešeného problému**

**Špecifikácia zadania :**

Aplikácia Word Hunt je hľadacie slovné puzzle, ktorého hlavným cieľom je poskytnúť používateľom zábavnú a stimulujúcu skúsenosť pri hľadaní slov v mriežke písmen.

Hlavným problémom, ktorý aplikácia rieši, je nedostatok vhodných prostriedkov na zábavu a trávenie voľného času prostredníctvom slovných hier. S Word Hunt môžu používatelia rozvíjať svoju slovnú zásobu, zdokonaľovať svoje schopnosti v hľadaní slov a relaxovať v interaktívnom prostredí.

Aplikácia je určená pre všetkých, ktorí majú záujem o slovné hry a chcú si vylepšiť svoje slovné schopnosti. Môže slúžiť ako skvelý spôsob trávenia voľného času a vytvorenie výzvy pre používateľov rôznych vekových kategórií.

Hlavné ciele a funkcie aplikácie Word Hunt sú:

1. Hľadanie slov: Používatelia môžu prechádzať mriežku písmen a hľadať slová v rôznych smeroch (horizontálne, vertikálne, diagonálne).
2. Skórovanie: Aplikácia udržuje skóre používateľa na základe nájdených slov a ich obtiažnosti.
3. Rôzne úrovne obtiažnosti: Aplikácia ponúka rôzne obtiažnosti, od jednoduchých až po náročné, aby vyhovovala rôznym úrovňam schopností používateľov.

Očakávaným výsledkom aplikácie Word Hunt je poskytnúť používateľom príjemnú a vzrušujúcu skúsenosť pri hľadaní slov a stimulovať ich mozgovú činnosť. Taktiež by mala priniesť zábavu a relaxáciu.

**Popis a analýza projektu**

Na začiatku keď hrač klikne na aplikáciu tak sa ocitné v úvodnej obrazovke kde má na výber 3 tlačidlá hraj, hodnotenie alebo koniec

A screenshot of a phone

Description automatically generated with medium confidence

V tabuľke hodnotení teraz nie je žiadny záznam lebo si zapamätáva len hry ktoré sa hrali pri danom sedení. Keď hráč klikne na tlačidlo Hraj tak sa presunie do novej obrazovky ktorá vyzerá nasledovne.

A screenshot of a phone

Description automatically generated with medium confidence

V tejto častí aplikácie si môže používateľ zvoliť z 3 možností Ľahká, Stredná alebo Ťažka ktorá mu určí ako obťiažne bude mať dané mriežky. Napr. keď si vyberie ťažkú tak bude mať mriežku 8x8 v ostatných by mal menšiu a bude musieť najsť viacej slov aby si navýšil svôj čas aby mu hra neskončila.

A screenshot of a crossword

Description automatically generated with medium confidenceKeď si hráč vyberie obtíažnosť tak prejde sem.Pozrite sa teraz na obrázok čislo 1 Ako môžeme vidieť tak sme si zvolili ťažku obtiažnosť takže máme len 2 minuty na najdenie až 4 slov v danej tabulke. Možeme vidieť že v pravom hornom rohu je napísaná obtiažnosť aku hráč hraje.V strede je počet bodov a vpravo je násobič s ktorými sa nasobí najdene slovo. Slovo je za 100 bodov. Keď hráč najde slovo tak sa mu zvýrazní zelenou farbou a zmizne z hladaných slov. Keď hrač najde všetky slová tak sa mu zväčší jeho čas a vygenerujú sa nové slová. Nazorná ukážka bude na obrázku 3.

Obrázok 2 Nájdené slovo

Obrázok 1 Úvodná obrazovka

A screenshot of a game

Description automatically generated with medium confidence

A screenshot of a cell phone

Description automatically generated with low confidence

Obrázok 4 Koniec hry

Obrázok 3 Vygenerovaný novy grid

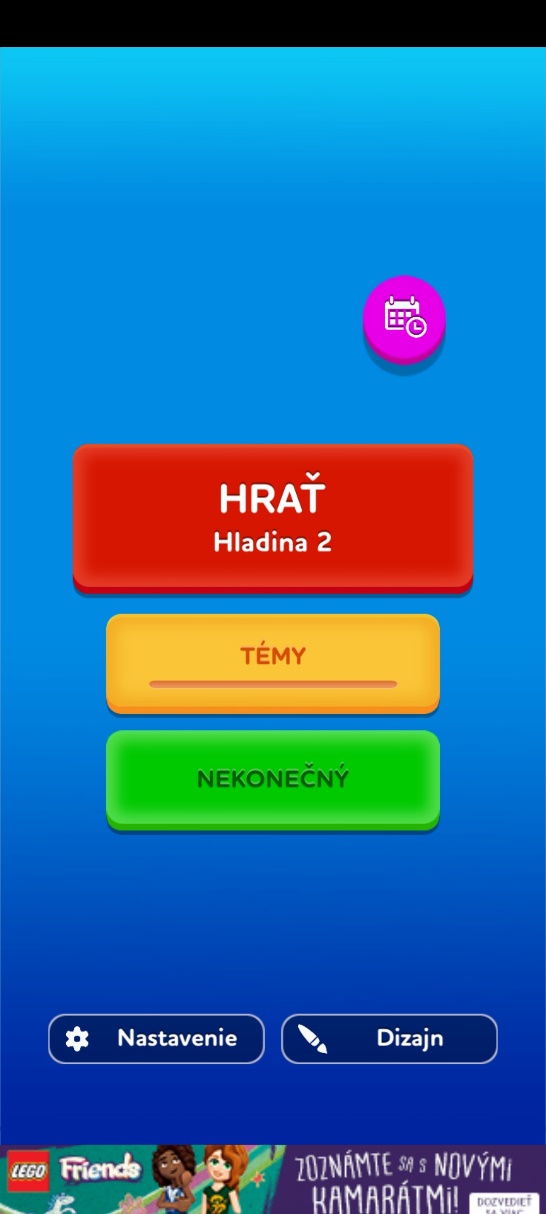
Ked hráčovi vyprší čas tak bude presunutý do obrazovky EndGameFragment. Kde sa vypíše dosiahnuté skóre a zobrazia sa nové možnosťi či chce hráč začat hrať odznova s tým istým nastavením aký si vybral keď si zvolil danú obtiažnosť alebo pôjde do hlavnej obrazovky kde sa bude môcť následne pozrieť na svoje skóre.

A picture containing text, screenshot, software

Description automatically generatedAko môžeme vidieť tak sa zapísal posledná hra do jeho tabuľky v zázname obťiažnosť a skóre.

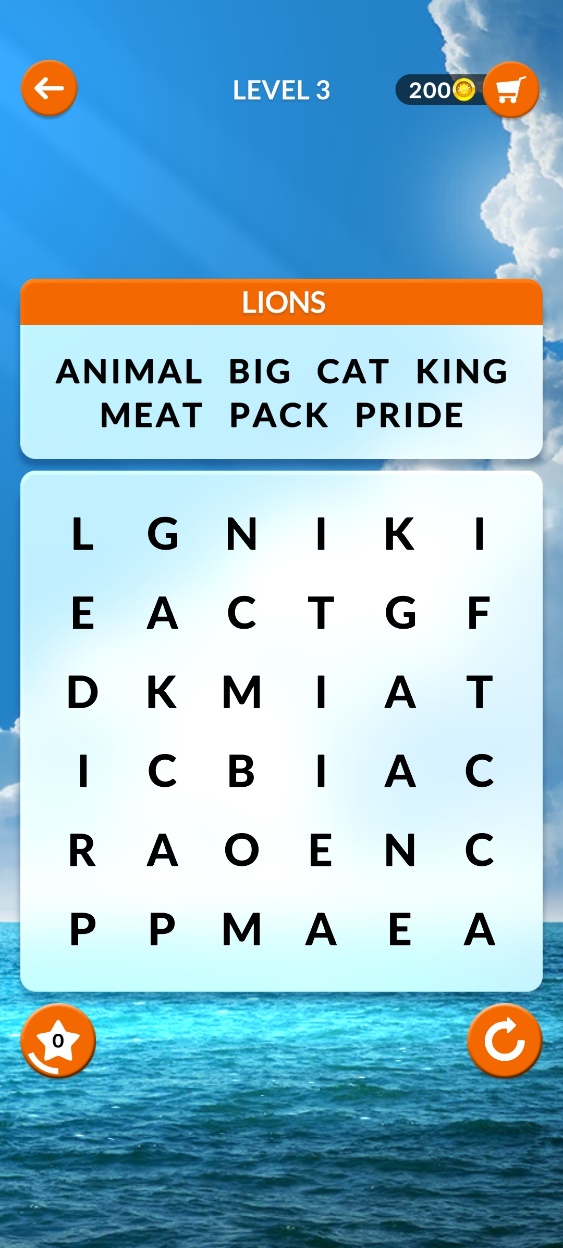
Obrázok 5 Hodnotiaca Tabuľka

A picture containing text, font, screenshot

Description automatically generated

Hra Word Search – Word Puzzle rieši podobný problém aplikácia Word Hunt ale z veľkým rozdielom že hra nemá obtiažnosti a hra nikdy neskončí dokým hráč nepovie tzn. že nemá časovač a hra je rozdelená skorej na nejaké leveli ktoré hráč plní ktoré sa týkajú nejakej témy napr. jedlo. Samozrejme následne nemá ani hodnotiacu tabuľku kde hráč vidí svoje úspešne hry. Jednou z najväčších slabín čo vidím na danej aplikácií je priveľa reklám že som sa ani nemohol do aplikácie dostať. A po každom leveli to isté čo sa v aplikácií Word Hunt nemôže stať.

A screenshot of a game

Description automatically generated with low confidenceA screenshot of a phone

Description automatically generated with low confidence

Toto je aplikácia Word Search Explorer. Aplikácia rieši ten istý problém ale zasa ako môžeme vidieť tak pracuje len z pred vytvorenými levelmi takže niekedy musia leveli dôjsť alebo sa budú opakovať čo sa v aplikácií Word Hunt nemôže a hráč má vždy nepredvídateľnú hraciu plochu. Taktiež tam nie sú leveli obťiažností a žiadne hracia skóre.

A picture containing text, diagram, plan, parallel

Description automatically generated

UML

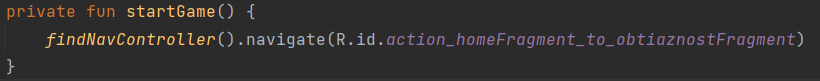
**Implementácia aplikácie**  
V tejto časti sa pozrieme ako bol wordHunt naprogramovaný. Začal by som asi z MainActivity:  
Jej úlohou je inicializovať a spravovať navigačný kontrolér, ako aj konfiguráciu App Baru pre navigáciu v rámci fragmentov.

Toto je ukážka pohybu jedného fragmenu z nav\_graph.xml podľa ktorého sa vie pohybovať.

A screen shot of a computer program

Description automatically generated with low confidence

Každému fragmentu u ktorého chceme aby sa niekam vedel presunúť potrebujeme vytvoriť akciu ktorej pridelíme id a destináciu kam by mal cestovať do iného fragmenu to docielime vložením jeho id.



Obrázok Ukážka v homeFragmente po kliknuti na hraj

**HomeFragment:**  
Tento fragment predstavuje úvodnú obrazovku aplikácie ktorú vidí hráč keď ju zapne. Hrač môže vidieť 3 buttony hraj, hodnotenie, koniec. Keď klikne na hraj tak sa vykoná metóda startGame() a bude presunutý do Obťiažnosť Fragmentu. Keby klikol na hodnotenie tak by sa presunul do recycleView kde sa ukladajú jednotlivé záznamy o hraných hrách. Koniec ukončí aplikáciu.   
Zaujímavou častou v homeFragmente je asi keď som skúšal animovanie daného textu pomocou knižnice AnimatorSet, ObjectAnimator.  
A screen shot of a computer program

Description automatically generated with low confidence

A screen shot of a computer program

Description automatically generated with low confidence

Tu som vytvoril niekoľko animácií ktoré som spojil do jedného celku a spustil som to pre daný text. Konečný výsledok ste možno už videli.

**Obtiažnosť Fragment:**

Tento fragment slúži na zvolenie obtiažnosti pomocou troch buttonov Ľahká ,Stredná, Ťažká. Keď hráč klikne na daný button tak sa nastaví objektu jeho parametre a s tými bude pracovať ďalej GameFragment, EndGameFragment a HodnotenieFragment.  
A screenshot of a computer program

Description automatically generated with medium confidence

Takto vyzerá daný objekt.  
A screen shot of a computer

Description automatically generated with medium confidence

Príklad ľahkej obťiažnosti  
**SharedViewModel:**Tato trieda sa používa na zdielanie dát mezdi fragmentmi GameFragment a HodnotenieFragment. keď hra skončí tak zavolá metódu addScore ktorá pridá záznam do recycleView v HodnotenieFragment.

A screen shot of a computer program

Description automatically generated with medium confidence

**HodnotenieFragment:**Fragment HodnotenieFragment slúži na zobrazenie skóre v RecyclerView.

Využíva ViewModel pre získanie zoznamu skóre z gameFragmentu.

Trieda ScoreAdapter je adaptérom pre zobrazenie skóre v RecyclerView. Rozširuje triedu RecyclerView.Adapter, ktorá poskytuje potrebné metódy na správu dát a vytváranie/viazanie view holderov.

A picture containing text, screenshot, font

Description automatically generated

Trieda ScoreViewHolder: Je vnorená trieda v rámci triedy ScoreAdapter. Rozširuje triedu RecyclerView.ViewHolder a reprezentuje jedno zobrazenie položky v RecyclerView. Obsahuje odkazy na zobrazenia v rámci rozloženia položky a poskytuje metódu bind na aktualizáciu zobrazení s príslušnými dátami skóre.

A screenshot of a phone

Description automatically generated with medium confidence  
takto vyzerá recycleView po dvoch hrách.

**GameFragment:**Fragment GameFragment je zodpovedný za spracovanie hry. V tomto fragmente sa zobrazuje hracia plocha s mriežkou písmen, zoznam vybraných slov, obtiažnosť hry, skóre a časovač.

Hra končí, keď uplynie časovač. Časovač je nastavený podľa obtiažnosti: ľahká (3 minúty), stredná a ťažká (2 minúty).

A screenshot of a crossword

Description automatically generated with medium confidenceAko môžeme vidieť tak hracia plocha je spravená cez gridLayout. Kde sme nastavili tieto parametre.A screen shot of a computer code

Description automatically generated with low confidence

Hra môže byť maximálne o veľkosti 10x10 aj keď hra využíva len 8x8. zároveň sme nastavili pozadie grid\_cell\_background ktorý je ale len celý biely.

Hracia plocha

Dôležité Metódy:

onViewCreated: Zobrazuje informácie o obtiažnosti, skóre, násobku skóre a čase. Nastavuje časovač podľa obtiažnosti. Nastavuje veľkosť hracej mriežky. Vytvára hraciu mriežku, inicializuje zoznam slov, vkladá slová do mriežky a povoľuje výber slov.

initializeWordList: Inicializuje zoznam slov na základe obtiažnost s ktorým bude pracovať. Z 60 slov si vyberie náhodné slová podľa obtiažnosti ľahká 2 stredná 3 ťažká 4.

createGrid: Vytvára hraciu mriežku a inicializuje každy jednotlivý grid cell ako vyzerá čo obsahuje a pod.

startTimer: Spúšťa časovač z knižnice CountDownTimer a aktualizuje zostávajúci čas zobrazený na obrazovke. Po uplynutí času sa odosiela skóre do sharedViewModel a prechádza sa na EndGameFragment.

A screen shot of a computer program

Description automatically generated with low confidence

placeWordsInGrid: Vkladá vybrané slová do hracej mriežky. Slová sa vkladajú náhodne v rôznych smeroch. Funguje to na princípe že sa snaží najprv vložíť slová ktoré boli dané pomocou metódy initializeWordList. Vždy si vyberá náhodnú pozíciu a smer ktorým bude vkladať dané slovo ak nastane error že by sa dané slová mali stretnúť tak si vyberie novú pozíciu kde položí dané slovo takto opakuje dovtedy dokedy sa mu to nepodarí. Keď vloží všetky slová tak zaplní prázdne mriežky náhodnými písmenami.

findCellTextView je metóda, ktorá sa používa na nájdenie TextView (zobrazenie textu) v rámci mriežky na základe poskytnutých súradníc. Kontroluje dotykové súradnice s porovnaním s danými bunkami pričom sa ale berie tolerancia -40 aby hra nebola veľmi citlivá na dotyk tým myslím že by hrač pohybom prsta vybral všetky bunky v ceste jeho prsta aj tie ktoré by nechcel.

clearGridAndPlaceWord je Metóda ktorá služí na vyčistenie obrazovky a upraví čas podľa obtiažnosťi aj multiplier a spustí upraveny časovač a spustí metódy   
  
createGrid()

initializeWordList()

placeWordsInGrid()  
  
ktoré vytvoria novú tabuľku na hľadanie po úspešnom nájdení všetkých slov.

Pri tejto metóde som spolupracoval aj s internetom konkrétne kde som zobral nápady a zistenia ako by to mohlo fungovať a išlo by to nejako urobiť :

EnableWordSelection: Metóda enableWordSelection() je zodpovedná za povolenie výberu slov hráčom použítím prsta

Funguje na princípe že používateľ potiahne prsť po bunkách a výtvara tím slovo ak našiel dané slovo tak sa zvýrazní (zelenou farbou) Ak nie tak sa vráti do pôvodného stavu ako neoznačená bunka. Používateľ môže takisto pracovať aj už s najdenými bunkami a keď tak vytvoriť nové slovo.

**EndGameFragment:**Toto je posledný fragment do ktorého sa dostaneme keď vyprší čas a máme dve možnosti vrátiť sa do hlavnej obrazovky alebo spustiť hru odznova s tými istými nastaveniami ako sme vybrali v ObtiaznostFragment.

Zdroje ktoré som používal

<https://stackoverflow.com/questions/47170075/kotlin-ontouchlistener-called-but-it-does-not-override-performclick>

<https://gist.github.com/emedinaa/135f89d288ba64db0fe21951b396c58c>

<https://proandroiddev.com/android-touch-system-part-3-motionevent-listeners-d4def814eba5>