

Házi feladat dokumentáció

Android alapú szoftverfejlesztés

[Félév: 2020 ősz]



Snake Game

Smodics Roland – (V4UL2C)

smodics.roland@gmail.com

Laborvezető: Rádi József Róber



Bemutató

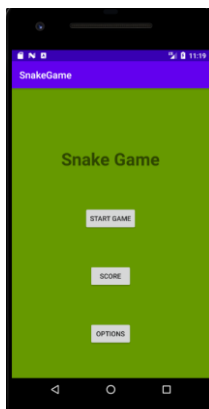
A program a régi Snake játékhoz hasonlít, amelyben egy kígyóval lehet mozogni egy adott pályán, meg lehet enni ételeket, és ettől nő a kígyó mérete. A játékos meghal, ha beleütközik saját magába, és ezzel a játéknak vége. A kígyó méretének növekedésével arányosan a sebessége is nő, így nehezítve a játékot.

Főbb funkciók

Az alkalmazásban alapvetően 3 Activity van. Van egy főmenü, ahonnan el tudjuk indítani a játékot, megnézhetjük a rekordokat, vagy beléphetünk a beállításokba. Van egy Options menü ahol a kígyó színét, a játék nehézségét vagy háttérét állíthatjuk, a beállítások között lapozva, és van egy játéktér activity, ahol maga a játék zajlik. A vezérlés gombok alapján működik. A játék végét pedig toast üzenet jelzi. A Score fragmentben láthatjuk az eddigi elért rekordokat, amiket egy külön adatbázisból olvas ki a program.

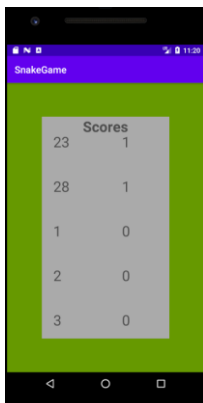
Felhasználói kézikönyv

Start Menü:



A játék launcher activityje, innen lehet a játékot elindítani, vagy kiválasztani, hogy melyik almenübe szeretnénk továbblépni.

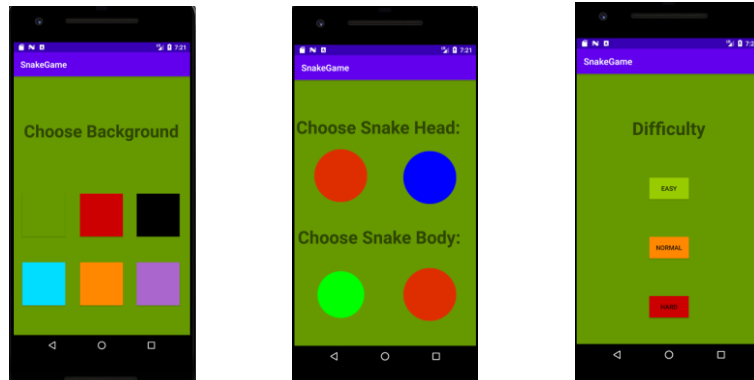
Score Fragment:



A főmenüből lehet megnyitni, és a korábban elért eredményeket listázza.

A Score Fragment egy ScoreAdaptert használ a RecyclerView megjelenítéshez, az adatokat pedig egy Room adatbázisból nyeri ki. Az adatbázis egy táblát tárol 2 oszloppal. Minden score-nak van egy ID-ja (primary key) és egy Value-ja. A megjelenítésnél Value alapján rendezve jelenik meg.

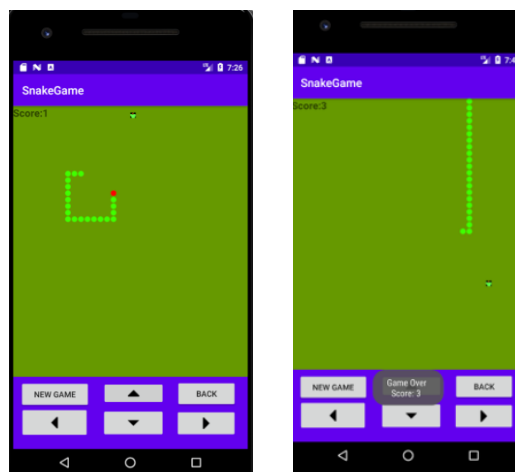
Options Menü:



Az Options menu szintén a főmenüből érhető el, és 3 oldala van amik között lapozni lehet. (Háttérszín, Kígyó és nehézségi szint beállítás)

Az options Activity egy FragmentPagerAdapterrel kezeli le a három fragmentet (OptionsAdapter).

Game Activity:



A játék menü két részből áll: Egy vezérlési panelből, és egy játéktérből. A nyilakkal mozgatható a kígyó, ami ha magába, vagy a játék falába ütközik meghal. A játék végét Toast üzenet jelti, és kiírja az elért score-t is.

Felhasznált technológiák

Itt kell felsorolni minden technológiát, technikát, külső könyvtárat, komplexebb algoritmust, ami növeli az alkalmazás értékét. Osztályzáskor ezt a fejezetet nézzük meg először.

Külső osztálykönyvtár használata esetén a könyvtár neve legyen link, ami annak elérhetőségére mutat.

A kulcsszavak legyenek **félkövér** betűtípussal szedve.

Például:

- **Activity:** 3 Activity található a játékban: Main, Options, Game
- **Fragmentek, Adapterek:** Score fragment, Options fragmentek, ScoreAdapter, OptionsAdapter
- **RecyclerView, LinearLayout:** A Score Fragment **RecyclerView** alapján jeleníti meg az értékeket, az Options menü **LinearLayout**ot használ
- **Room db** alapú adattárolás
- **Notifications:** A játék végén **Toast** üzenet jelzi az eredményt és értesíti a felhasználót a végéről.

Fontosabb technológiai megoldások

A legnehezebb rész számomra az adatbázis bekötése, illetve az Activityk közötti kommunikáció kitalálása volt.