



Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Méréstechnika és Információs Rendszerek Tanszék

Házi feladat

Felhasználói dokumentáció

BEÁGYAZOTT RENDSZEREK SZOFTVERTECHNOLÓGIÁJA

Készítették:

Hornyák Máté Dániel

Molnár Marcell

Poleczki Ákos

2019. május 10.

1. A feladat rövid összefoglalása

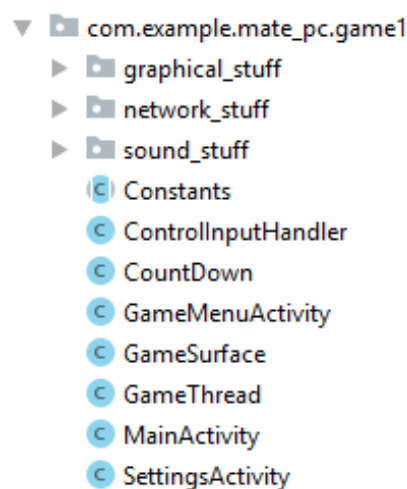
A megvalósított feladat egy kétdimenziós, oldalnézetes lövöldözős játék Android operációs rendszer alatt, mely a következőképpen működik.

Egyszerre két játékos játszik egymás ellen, összesen fejenként három életponttal, websocket alapú kapcsolaton keresztül. A játék célja az ellenfél életpontjainak lenullázása. A játék elindításakor a felhasználó egy felugró menüpontban adhatja meg a csatlakozni kívánt szerver IP címét, illetve lehetősége van ezen kívül az alapvető beállításokat változtatni a játék megkezdése előtt. Tehát a vezérlési módok és az egyéni ízlésnek megfelelő háttér kiválasztására, valamint a háttérzene ki-be kapcsolására. A beállítások elvégzésével és a csatlakozás gomb megnyomásával a felhasználó csatlakozik a megadott szerverhez. A beállításokat az alkalmazás két indítása között is megőrzi. Amennyiben mindkét játékos sikeresen csatlakozott, a játék egy vissza-számlálást követően elkezdődik.

A beállítások alkalmával megadott vezérlő módok alapján vagy gombokkal vagy joystick segítségével van lehetőség a karakter mozgatására, az ellenfél megsebzésére pedig egy dedikált nyomógomb szolgál. A nyomógomb lenyomásakor egy számláló indul el, mely néhány másodpercig blokkolja az újratüzelést, ezzel megakadályozva sorozatos támadást. A játék dinamikájának kiszélesítésének érdekében a blokkoláson kívül akadályok is elhelyezésre kerülnek a pályán, amelyekre akár fel is lehet ugrni a karakterrel.

2. A program felépítése

Az programot alkotó osztályok az alábbi képen látható struktúra szerint épülnek fel.



1. ábra. Az osztályok struktúrája

A fő package-en belül található:

- a *MainActivity*,
- a beállításokért felelős *SettingsActivity*,
- a főmenüt megjelenítő *GameMenuActivity*,
- a felhasználói bemeneteket kezelő *ControlInputHandler*,
- a *GameThread* osztály, amely egy számban megfelelő időközönként kirajzoltatja az aktuális nézetet,
- a *GameSurface* osztály, amely a kirajzolást és a játék logikáját valósítja meg az egyes játék objektumok állapotának frissítésével,
- a *CountDown* osztály, amely egy lebegő dialógus (*Dialog*) segítségével egy visszaszámlálót valósít meg mielőtt a játék elkezdődne,
- és az absztrakt *Constants* osztály, amelyben az alkalmazáson belül több helyen használt konstansokat tárolja.

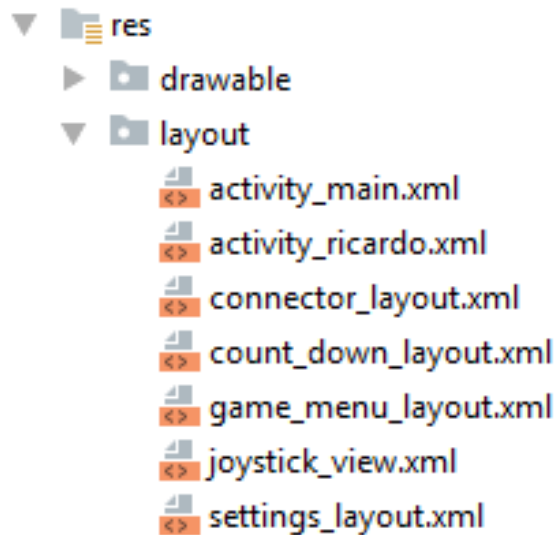
További három package-et találunk a források között, amelyek egyrészt a grafikus megjelenítés, a hálózati kommunikáció és az audió effektek szempontjából lényegesek:

- a *graphical_stuff*-ban találhatók a játékosok karaktereit, az akadályokat és a lövedékeket reprezentáló osztály leírások,
- a *network_stuff*-ban a hálózatkezelésért felelős socket osztály, a program indulásakor használatosa a hálózati kapcsolat kiépítéséért felelős osztály és egy külön szálat futtató osztály található. Ez utóbbi periódikusan küldi át a lényeges információkat a szerver felé.
- a *asound_stuff*-ban a háttérzenét kezelő és a különböző hangeffekteket lejátszó osztályok vannak.

3. A nézetek

4. MainActivity

A program indulásakor a MainActivity-ből fog egy példány létrejönni. Ennek az osztálynak a nézetében (*activity_main* található



2. ábra. Az *activity*-kben használt *layout*ok

- egy *LinearLayout*, amelyre a játék elemeit rajzoljuk ki,
- egy *LinearLayout*, amely a joystick alapú irányításnál fogja a *TouchEvent*ket "elkapni",
- valamint 4 gomb, amelyek a lövésért és a 3 irányú mozgásért felelősek.

A bemenő eseményeket egy külön osztályban kezeljük, ezért a gombokra és a joystick nézetére olyan *OnTouchListener*-t hozunk létre, amely egy függvényhívással továbbadja az eseményeket a *ControlInputHandler* osztálynak.

Létrehozzuk a *GameSurface* osztályt (amely a *SurfaceView* osztályból származik), és játék felületét megtestesítő *LinearLayout* nézetéhez hozzáadjuk.

Ezután elindítjuk a csatlakozásért felelős nézetet egy *ConnectorClass* formájában, majd várjuk a felhasználói interakciót.

5. GameSurface