Android – Feladat

Csináljunk TicTacToe alkalmazást!!!4!négy

1. feladatrész

- TicTacToe
- Layout alap
- Kattintásokra képcsere, játékosváltás

Resource-k / Erőforrások

- Töltsünk le a zinternetről
  - Egy X ikont
  - Egy kör ikont
  - Nevezzük el őket androidosan
    - ic_x.png / ic_x.jpg
    - Ic_circle.png / ic_circle.jpg

Layout

- Vertikális LinearLayout
  - Benne három horizontális LinearLayout
  - Mindegyik magassága 0dp és a súlya 0.3
  - Legyen mindnek ID-ja! (row1, row2, row3)
    - Bennük legyen 3 ImageView
      - Mindegyik szélessége 0dp és a súly 0.3
      - ID nekik nem fog kelleni
  - Minden más szélesség, magasság match_parent

Activity Osztályváltozók

- A feladat későbbi kiegészítésének megkönnyítése érdekében a sorok legyenek privát osztályváltozók
- private LinearLayout row;
- Valamint kelleni fog nekünk, hogy melyik játékos van soron
- private boolean currentPlayer;

Activity - OnCreate

- Az onCreate-ben szokásosan meg kell találni a képeket, hogy azokra OnClickListenereket pakolhassunk.
- Ehhez először meg kell találni a három sorunkat, például:
- LinearLayout row = (LinearLayout) findViewById(R.id.row);
- (Később majd bővítjuk a játékot, ezért a sorokat az osztály változói közé mentsük)
- A LinearLayout a JAVA kódban pont úgy viselkedik (számunkra fontos szempontból), mint egy ArrayList, amiben valamilyen View-k vannak (amikről tudjuk, hogy a mi képeink lesznek)
- Így hát nincs más dolgunk, mint mindhárom sor összes elemén végigmenni és beállítani a gyerek View-knak OnClickListenert

Activity – OnCreate II.

- A parancsok azért nem ugyanazok
- row.getChildCount() – hány View van benne (ez 3 lesz most minden esetben)
- row.getChildAt(i) – az i. elem a sorban
- OnClickListenert pedig a már megszokott
- valami.setOnClickListener(this)
- Paranccsal állítunk be, aztán az Activity-t kiegészítjük(ALT+Enter, Make the Activity implement…)
- Ha minden jól ment, ezzel kaptunk egy onClick metódust

OnClick(View amireKattintottunk)

- 2 dolgot kell itt elsőkörben végrehajtanunk
- Ha jól beállítottuk az onClickListenereket, akkor csak képre kattintva hívódhat meg ez a függvény
- És ezt a képet kényelmesen meg is kapjuk a függvény paraméterében
- Ezzel nincs más dolgunk, mint ImageView-á alakítani, majd beállítani neki egy erőforrások közül kitúrt képet (nyilván attól függően x-et vagy kört, hogy melyik játékos van soron), pl:
- imageView.setImageDrawable(getResources.getDrawable(R.drawable.ic_x))
- Valamint az aktuális játékost átállítani

II. feladatrész

- Új játék gomb megvalósítása
- Tegyük fel a Layoutra
  - Ezen kívül minden más legyen GONE
  - Rákattintva ez tűnjön el és jöjjön fel a játéktér

III. feladatrész

- Eredmény tárolása, nyertes kiírása
- Osztályváltozó: eredmény 3x3 int tömb, aktuális bökött X és Y (a kép pozíciója), ez is nyilván int lesz
- Érdemes készíteni egy függvényt, amely az aktuális játékos számát adja vissza a boolean alapján, amit utána majd a tömbben tárolunk
- onCreate kiegészítése
  - Ahol az onClickListenerek be vannak állítva, ott beteggelhetjük a képeinket a sorszámaikkal, hogy utána könnyen tudjuk csak az onClickben kapott View paraméteréből, hogy az pontosan hanyadik kép is volt, amire böktünk.
  - Erre van egy egyszerű módszer minden View-nak van egy tag változója, amibe bármit bele tehetünk a setter-ével és kivehetjük a getterével:
  - view.setTag(i), int i = (Integer) view.getTag()

III. Feladatrész - folytatás

- onClick:
  - Itt meg kell határoznunk az aktuális X-et és Y-t, hogy az eredmény tömbe beleírhassuk a játékosunk számát, aki épp soron van
  - Az egyikhez csak ki kell olvasni a már jó előre elteggelt integerünket
  - A másikhoz meg kell néznük, hogy a paraméterül kapott megbökött View, melyik sorba tartozik
  - view.getParent() == row
  - Ezután csak beírjuk az eredmény tömbe, az aktuális játékos számát a kiszámolt pozicióba
  - Majd meghívunk egy nyertes ellenőrző függvényt (amit létre is kell hoznunk)

III. Feladatrész – A nyertes ellenőrzése

- Hogyan tudjuk leellenőrizni, hogy vége-e a játéknak?
- (Nyilván elég ennyit tudni, hiszen, ha vége, akkor az nyert, aki épp most tett, teljes indukcióval bizonyítható, házi feladat ☺)
- 4 dolog van, amit ellenőriznünk kell
  - Az aktuális sor
  - Az aktuális oszlop
  - Az egyik átló
  - A másik átló

III. Feladatrész – 4 dolog

- Az aktuális sorhoz lehet csinálni egy for ciklust ami fut 0-2-ig és ha a sor minden elemében az a játékos szerepel, aki most épp soron van, akkor ő nyert.
- Az aktuális oszlophoz lehet csinálni egy for ciklust ami fut 0-2-ig és ha az oszlop minden elemében az a játékos szerepel, aki most épp soron van, akkor ő nyert.
- Az egyik átlóhoz lehet csinálni egy for ciklust ami fut 0-2-ig és ha az átló minden elemében az a játékos szerepel, aki most épp soron van, akkor ő nyert.
- A másik átlóhoz lehet csinálni egy for ciklust ami fut 0-2-ig és ha az átló minden elemében az a játékos szerepel, aki most épp soron van, akkor ő nyert.

III. Feladatrész – 1 dolog

- Hasonlóak, nem?
- Most akkor csináljak 4 for ciklust?
- Csinálhatsz, de az egész megoldható egyben is.
- ☺

Ne felejtsétek el nekem elküldeni levélben (mármint az én csoportom :D)

http://erdekes.herokuapp.com

gyulavari.adam+petrik@gmail.com