

Kapj el! App Inventor alkalmazás

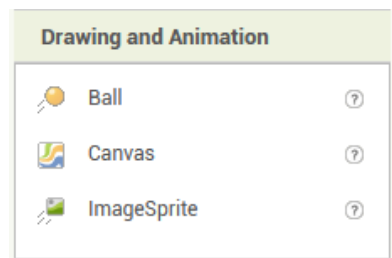
1. Kezdeként az App Inventor főoldalán -> Tutorials -> Mole Mash opciónál letöltjük a szükséges képet az alkalmazásunkhoz.

Getting Started

Connect to the App Inventor web site and start a new project. Name it "MoleMash". Open the Blocks Editor and connect to the phone.

Also download this [picture of a mole](#) and save it on your computer.

2. Létrehozunk egy új projectet KapjEl néven.
3. Screen1 Title tulajdonság -> Kapj el!
4. Szükségünk van egy Canvas komponensre. Neve: cvJatekTer. Width tulajdonsága Fill parent. Height->300 pixels.
5. Canvas komponensre ráteszünk egy ImageSprite komponent. Ez lesz az a komponens, ami a vakond képét fogja hordozni és mozog jobbra-balra. Neve: isVakond. A komponens Picture tulajdonságához feltöltjük az elején letöltött vakond képet.



6. A Canvas komponens alá elhelyezünk egy Layout->HorizontalArrangement komponent. Neve: haEredmenyek. Width->Fill parent.
7. Elhelyezünk a HorizontalArrangement komponensre User Interface->Label komponenseket (6 darabot). A számokat tartalmazó Label-ek nevét át is nevezzük, ugyanis ezeket az értékeket a későbbiekben lekérdezzük és módosítjuk a program futása közben. (IITalalatok, IIMelle, IIdozito). Minden Label Text tulajdonságát az alábbi képen látható módon átnevezzük.

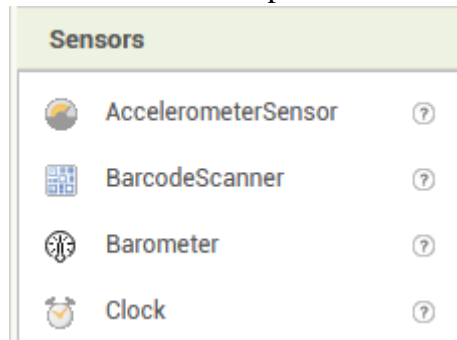
Találatok: 0 Mellé: 0 Időzítő: 1000

8. Elhelyezünk egy újabb HorizontalArrangement komponent. Neve: haVezerlogombok. Ezen a komponensen fognak elhelyezkedni a nyomógombok. Width->Fill parent. AlignVertical->center.
9. User Interface->Button komponent helyezünk el az előbbi komponensen. Neve: btNullaz. Text tulajdonsága->Nulláz. Második nyomógombot is elhelyezünk az előző nyomógomb mellé. Neve: btKilep. Text tulajdonsága->Kilép.
10. Ha nem szeretnénk, hogy túl közel legyen a kettő nyomógomb egymáshoz, akkor elhelyezünk egy Label komponent a kettő gomb közé. Távolságát pedig szabályozhatjuk azzal, hogy a Label Text tulajdonságához hány darab szóközt helyezünk.

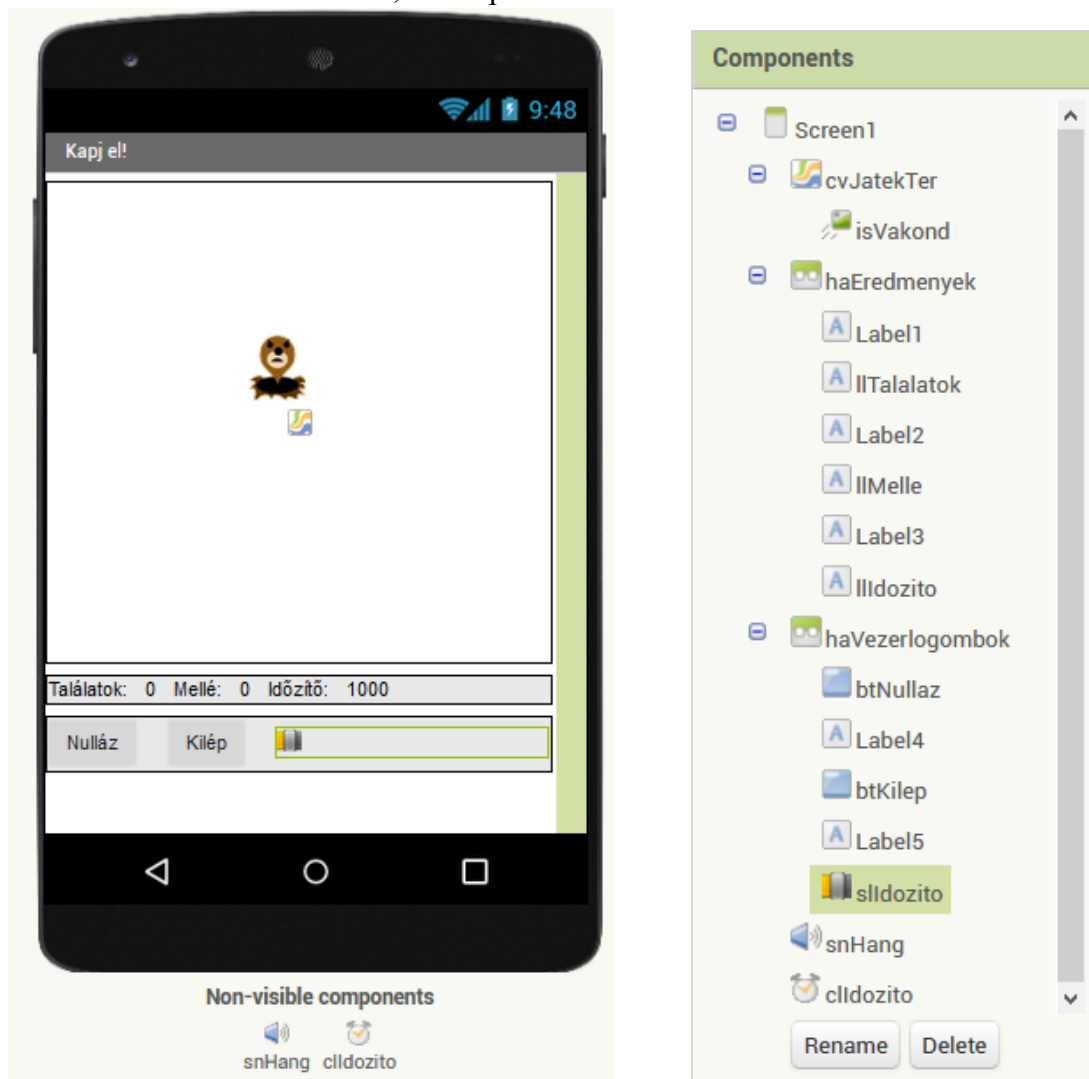
11. Végül a nyomógombok mellé elhelyezünk egy csúszkát is. User Interface->Slider.
Neve: slIdozito. A későbbiekben ezzel állítjuk a vakond mozgásának a gyorsaságát.
Width->Fill parent. MinValue->100, MaxValue->3000.



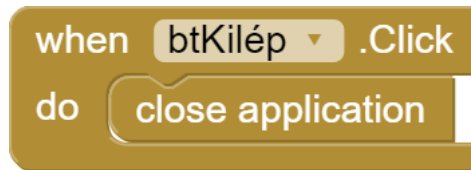
12. Media->Sound komponenst helyezünk még el. Neve: snHang. Ez a komponens segítségével érhetjük majd el, hogyha sikeresen rányomtunk a vakondra, akkor rezegjen a telefon.
13. Emellett elhelyezünk egy Sensors-Clock komponenst is. Neve: clIdozito.



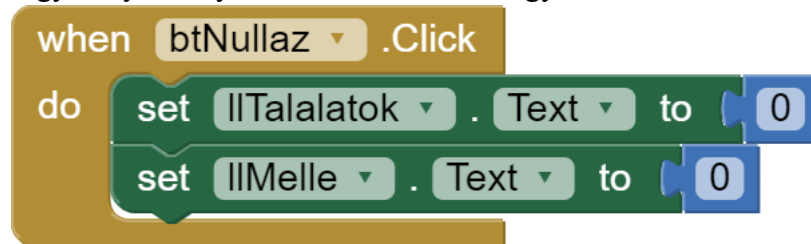
14. Ezzel elkészültünk a felülettel, és átlépünk a Blocks nézetbe.



15. Elkészítjük a Kilép gomb eseménykezelőjét. A Kilép gomb megnyomása után az alkalmazásunkat bezárjuk.



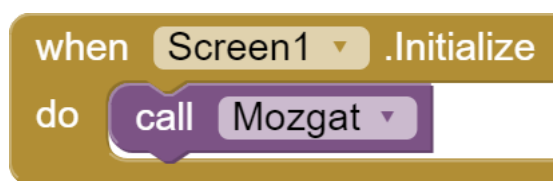
16. A Nulláz gomb eseménykezelője. A kettő Label komponens értékét, amelyek számolják, hogy hányszor nyomtunk a vakondra, vagy mellé kinullázzuk.



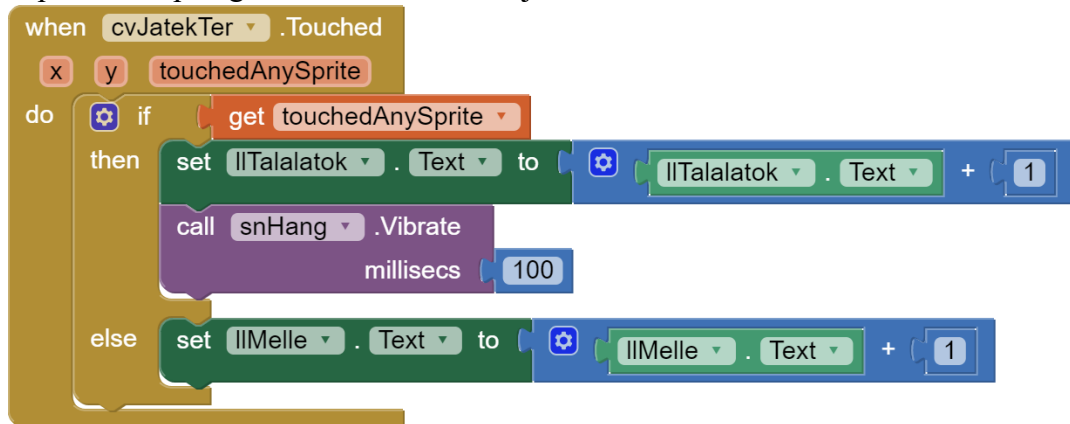
17. A vakond áthelyezése egyik helyről a másikra. Procedures->to do. Neve: Mozgat. Meghívjuk a vakond ImageSprite MoveTo metódusát és új x,y koordinátát kell adnunk neki. Csak a Canvas területén mozoghat. A Canvas-nak a bal felső sarka a 0,0 koordinátával rendelkezik. Mivel a vakond képét nem szeretnénk levinni a látható játéktérből, ezért a Játéktér szélesség és magasság méretéből ki kell vonnunk a vakond szélességét és magasságát.



18. Kettő esetben van szükségünk a Mozgat metódusra. Amikor elindítom az alkalmazást vagy amikor letelt az időzítő által meghatározott intervallum.



19. A „Tappizás” kezelése. Ha eltaláljuk a vakond képét, akkor a Találatok számát növeljük 1-el. Emelett találat esetén rezegjen is a telefon. Ha nem találjuk el a vakond képét, akkor pedig a Mellé értékét növeljük.



20. A „csuszka” eseménykezelője. Ha megváltoztatjuk az időzítőt, akkor először le kell állítanunk majd utána állítjuk át az értékét. Ezek után újraindítjuk az időzítőt. Végül az lIdozito label értékét (Text tulajdonságát) is átállítjuk a pillanatnyi csuszka alapján.

