

MIT APP INVENTOR 2

Kezdő lépések

Sziasztok!

Virga Kriszta vagyok, informatika tanár. Ezt az anyagot azért készítettem, hogy a teljesen kezdő, legelső lépésektől mutassam be egy nagyon egyszerű kis applikáció fejlesztésén keresztül az MIT App Inventor2 használatát. Ezen tizenpár oldal megírása SOKKAL több időbe került ☺, mint az applikáció elkészítése (ami egyébként kb. 10 perc lenne, de a leírása, a képernyőképek összeszerkesztése viszont jó pár órát tett ki).

Nézzétek el, kérlek, a kinézetet és a biztosan elő-előforduló fogalmazási hibákat, pont olyan leterhelt vagyok, mint bármelyik szakmais infótanár, szóval ezt itt leginkább csak munkaanyagnak szánom. ☺

Aki gondolja és szeretné, az átnézi, kinyomtatja (jó, inkább ne, mert az nem túl környezetbarát megoldás) és lépésről lépésre megcsinálhatja első applikációját. Amint ráéreztek Ti is és a diákok is a fejlesztés ízére, gyorsan fogtok haladni. Később, amint lesz időm, összeszerkesztem és kiteszem néhány kész, egyszerű applikációnk képernyőképét is, hogy legyen pár ötlet, amin tovább haladhattok. Ezen kívül természetesen mindenki használhatja az MIT App Inventor tutoriálját is, nagyon jó kis applikációk vannak fent.

Amint látjátok, nem csináltam egyetemi székfoglaló szintű anyagot, bár nyilván büszkébb lennék rá. De ebből az is látszik, hogy bárki képes ezt a szintet megugrani és másnak segítő, támogató anyagot adni, szóval hajrá, készítsünk minél több ilyen (vagyis ennél sokkal jobb) segédletet, amiről csak tudunk.

Még egy tanács! Az applikáció telefonra történő másolásához (erről a legvégén, az Applikáció-építés részben részletesebben is írok, de jobb, ha előre is jelzem), legyen a diákoknál adatkábel.



Az anyagban ilyen helyes kis figyelmeztető háromszögekkel jeleztem, ha valami tipikus hibába bele lehet futni (már pedig mi belefutottunk néhányszor)

Tartalom

1. Keressük fel az MIT App Inventor2 honlapját	2
2. Belépés	2
2.1.) Gmail fiókos bejelentkezés	2
2.2.) Engedélyezések, visszaigazolások	2
3. Applikáció fejlesztés	3
3.1.) DESIGNER használata – nézet kialakítása	4
3.2.) BLOCKS használata - programozás, blokkok használata	9
5.) Applikáció-építés	14

KEZDŐ LÉPÉSEK

1. Keressük fel az MIT App Inventor2 honlapját

MIT App Inventor 2

ai2.appinventor.mit.edu/ ▾ Oldal lefordítása

To use App Inventor for Android, you must use a compatible browser. Currently the supported browsers are: Google Chrome 29+; Safari 5+; Firefox 23+.

<http://ai2.appinventor.mit.edu/>

2. Belépés

2.1.) Gmail fiókos bejelentkezés

Az App Inventor2 egy gmail-es fiókot fog kérni tőlünk, melyen keresztül elérhetjük a programot. (Ha van már gmail-es fiókunk, lépünk be azzal, ha nincs, hozzunk létre egyet. Mi, a suliban, a diákokkal létrehozatunk olyan gmail-es fiókot, melyet aztán a különböző programokhoz használunk regisztrációs fiókként).

Innentől kezdve ezen a fiókon keresztül érhetjük el az összes programunkat, melyet ide fejlesztünk.

2.2.) Engedélyezések, visszaigazolások

Miután bejelentkeztünk a fiókba, következő lépésként engedélyezzük, hogy az App Inventor2 használhassa a fiókot annak érdekében, hogy innentől kezdve ezen keresztül szerkeszthessük az applikációinkat.

Google fiók

An application is requesting permission to access your Google Account.

Kérjük, válassza ki a használni kívánt fiókot.

• regi.stracio01@gmail.com

Google is not affiliated with the contents of the application or its owners. If you sign in, Google will share your email address with the application but not your password or any other personal information.

Engedélyezés

Köszönöm, nem

[Bejelentkezés másik fiókba](#)

☒ Jegyezze meg ezt az engedélyt 30 napig

Utána már csak egy felkérés következik, hogy a fejlesztések érdekében töltsünk ki egy rövid felmérést. Választhatjuk azt, hogy most töltsük ki (Take Survey Now), később (Take Survey Later) vagy soha (Never Take Survey).Érdemes azért később ennnyivel is támogatni a fejlesztéseket, de az elején még nyugodtan hagyjuk ezt ki.

Welcome to App Inventor!

Please fill out a short voluntary survey so that we can learn more about our users and improve MIT App Inventor.

Take Survey Now

Take Survey Later

Never Take Survey

Ha mindezekkel megvagyunk, jöhet is a szórakozás ☺, vagyis az applikáció készítés

3. Applikáció fejlesztés

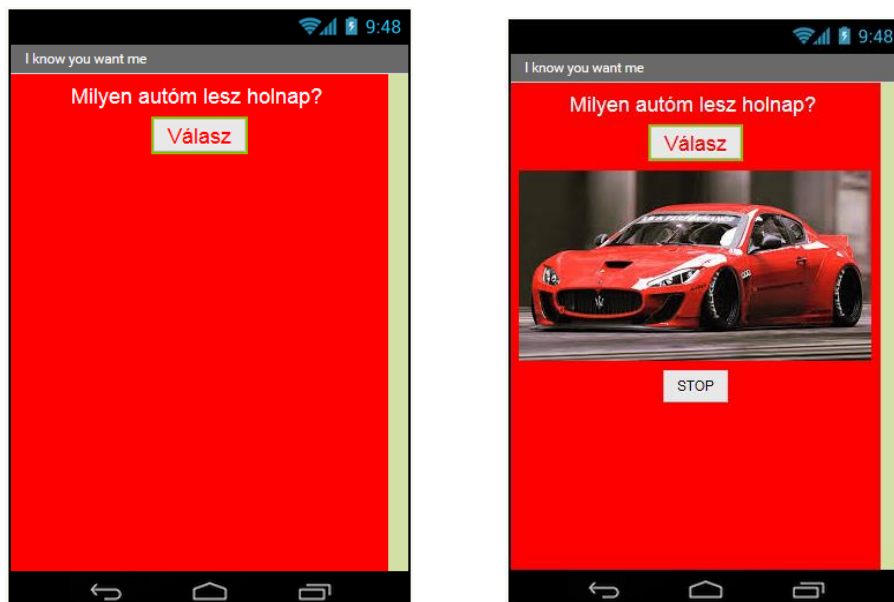
Egy könnyű, gyors (vagy akár bonyolult és lassú) applikáció fejlesztésének 3 fontos mozzanata van.

- DESIGNER - design kialakítása: itt alakítjuk ki, hogyan nézzen ki az applikációnk (ez csak eszközök behúzogatásából és tulajdonságok megadásából áll)
- BLOCKS - program blokkok készítése: itt programozzuk az applikációt (a korábban a design részben már megadott eszközökhöz különböző programblokk lehetőségek vannak, melyekből a kívánt programrészeket összehúzogadjuk)
- BUILD – Applikációnk okos eszközre történő telepítése

Egy egyszerű (10 perc alatt összerakható) applikáción keresztül mutatom be a fontosabb elemeket és használatukat.

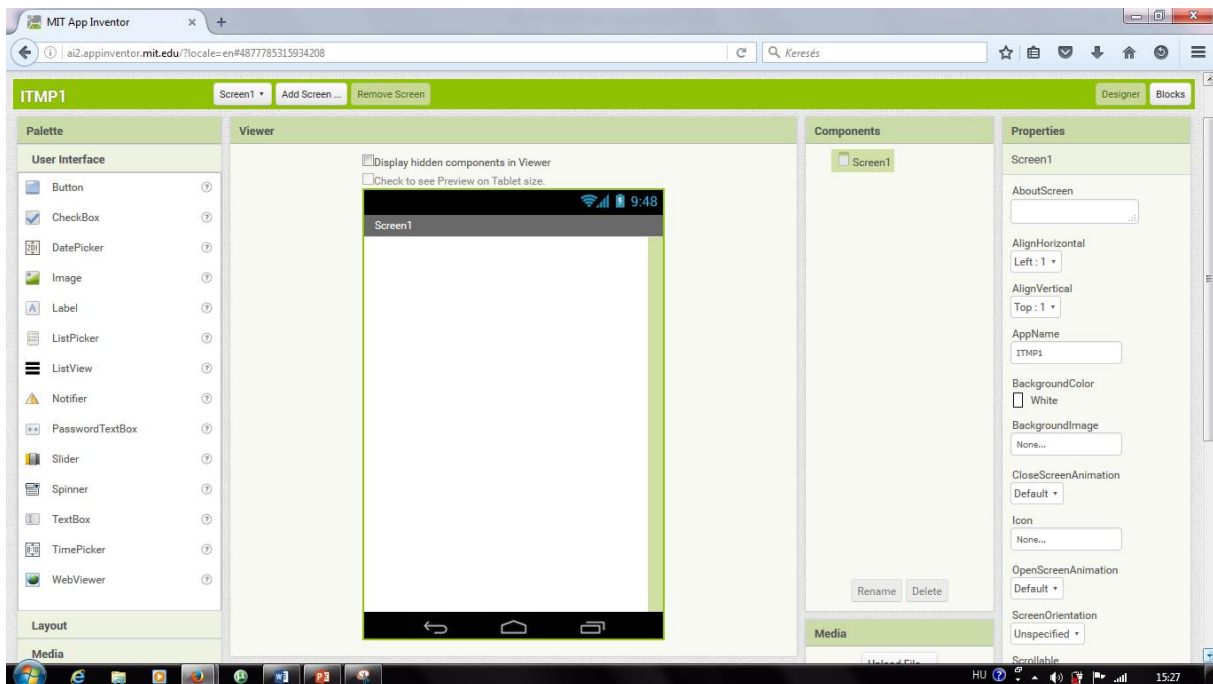
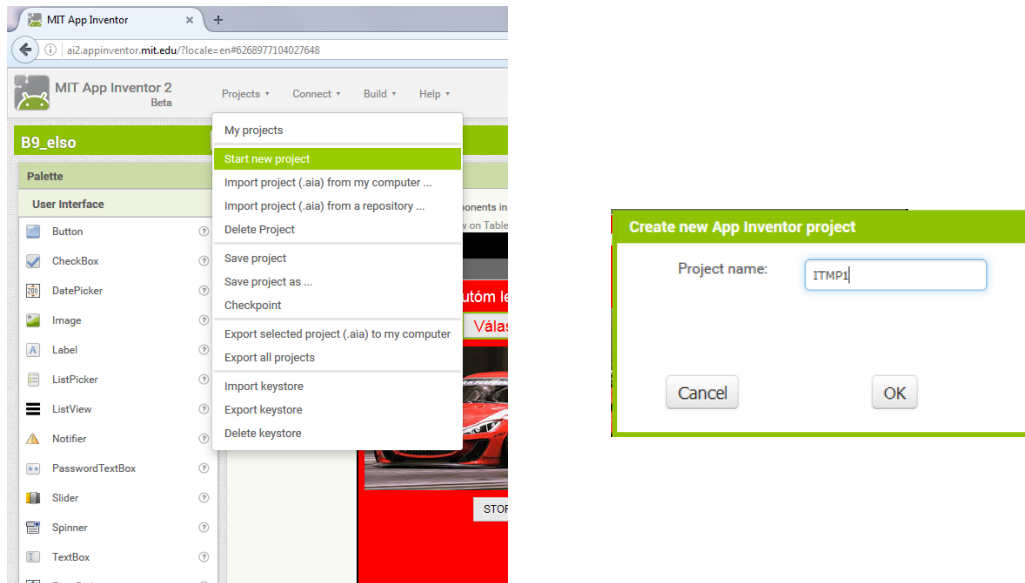
Az applikáció egy kérdést tesz fel a felhasználónak: Milyen autóm lesz holnap?, majd a Válasz gomb megnyomására megjelenik egy Ferrari (vagy Porsche, nem tudom, képtelen vagyok az autókat megkülönböztetni) képe, egy igazán vérpezsdítő zene lejátszásával egybekötve (Pitbull – I know you want me). Az ilyen feladatokat nagyon szökták szeretni a diákok ☺

Ezután a STOP gomb megnyomásával lehet eltüntetni a képet és leállítani a zenét.



3.1.) DESIGNER használata – nézet kialakítása

Először is nyitunk egy új projektet a legfelső sorban található Projects menüben (Projects / Start New project), majd adunk a projektünknek egy nevet. Ezután megjelenik egy felület, középen a szerkesztendő mobiltelefon képével, amire behúzhatjuk a megjeleníteni kívánt elemeket, illetve azok tulajdonságait szerkeszthetjük.



Baloldaltól indulva:

Palette – Ez tartalmazza az eszközöket, melyeket meg kívánunk jeleníteni a képernyőn. Ezen kívül számos olyan elemet is tartalmaz, amely nem jelenik meg a képernyőn, de a háttérben hozzájárul az applikáció működéséhez. (Pl. jelen applikációnkban a zene lejátszásához nem kell semmilyen látható eszköz a képernyőre, viszont mégis ide kell betenni a media/player eszközt, amin keresztül lejátszunk a zenét). A Palette-n található eszközök száma elég nagy, talán egy későbbi anyagban lesz idő ezek részletesebb ismertetésére is. (User Interface eszközöktől az adatbázis, illetve media tartalmak kezelésén át a LEGO Mindstorms robotok irányításáig sokféle eszközt tudunk beépíteni).

Középen a **Viewer**-ben szerkesztjük a képernyőt.

Jobb oldalon a **Components** ablak tartalmazza a felhasznált komponenseket, illetve a **Properties** ablakban állíthatjuk be a komponensek főbb tulajdonságait.

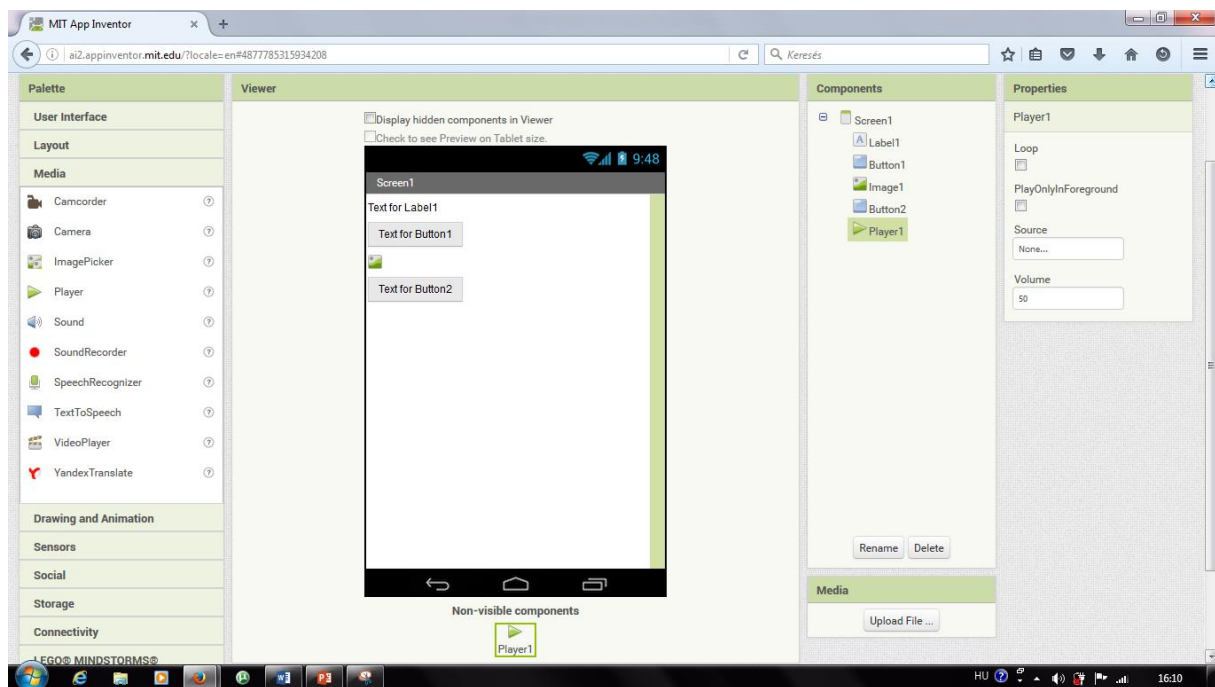
Applikációnkhoz az alábbi komponensek kellenek: (csak behúzzogatjuk őket a mobilképernyőre)

Palette/User Interface eszköztár

- **Label** – ez kiírás a felhasználónak, ezen szerepel majd a kérdésünk: Milyen autót szeretnél?
- **Button** – kell egy Válasz és egy Stop gomb (a Válasz és Stop elnevezést később adjuk meg)
- **Image** – ez fogja majd tartalmazni, ide tallózzuk be az autó képét (a képet már elmentettük a gépünkre)

Palette/Media eszköztár

- **Player** – ide linkeljük be a lejátszandó zenét (a zenét már letöltöttük gépünkre), ez nem a képernyőn fog megjelenni, hanem a képernyő alatti NON VISIBLE részen lesz látható (a kész applikációban pedig nem is látszik)



Mint látjuk, az eszközök már fent vannak a mobilon, de sem tartalom, sem design nincs még hozzá szerkesztve, ráadásul minden balra van rendezve.

Itt kezdődik a különböző komponensek szerkesztése.

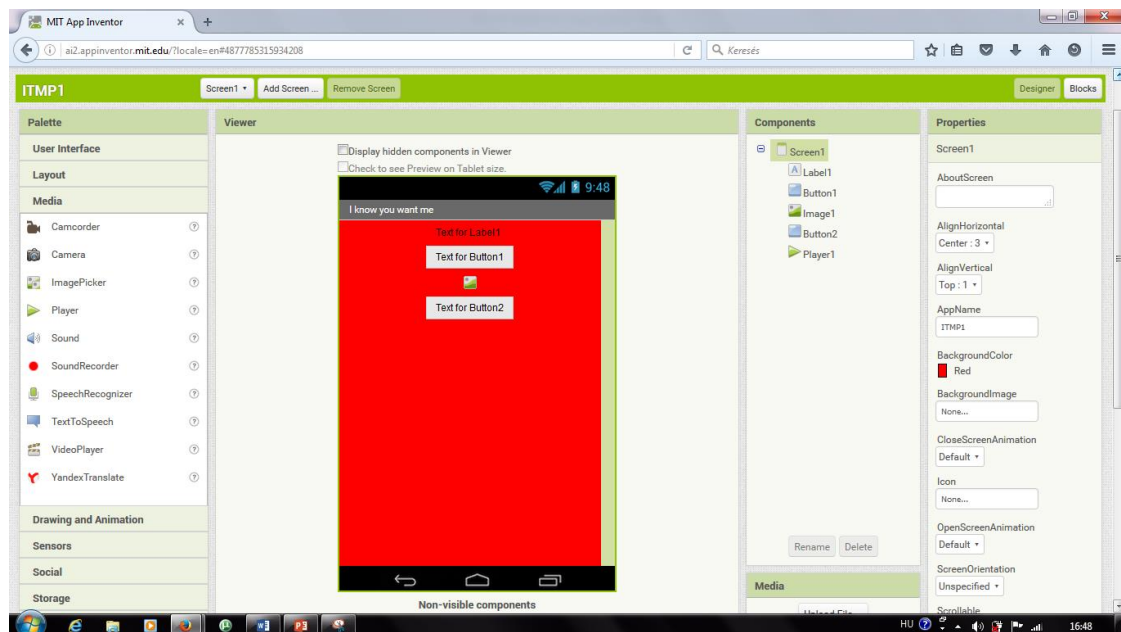


Figyelem, gyakori hiba, hogy nem a megfelelő komponensen állunk és úgy próbálunk szerkeszteni. Fenti mintánkon a Components ablakban látszik, hogy a Player1 eszközön vagyunk (annak zöld a háttere) és a mellette lévő Properties ablakban csak a Player1-hez tartozó tulajdonságokat tudjuk szerkeszteni. Ha más eszköz tulajdonságait szeretnénk beállítani, egyszerűen rákattintunk a kívánt eszközre.

Hát akkor szerkesszünk! 😊

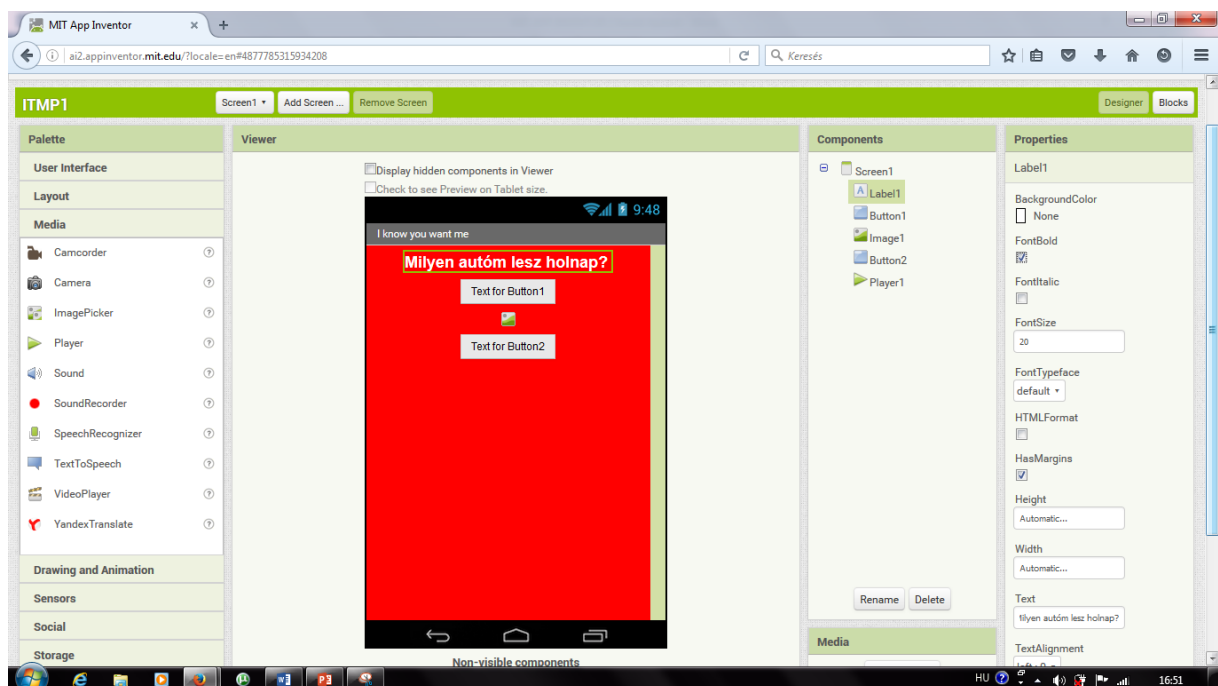
A **Screen1** eszközön, a **Properties** ablakban található tulajdonságok közül az alábbiakat állítottam át:

- AlignHorizontal: Center 3 (ezzel középre igazítottam a balra igazított alap beállításához képest)
- BackgroundColor: Red (piros lett a háttér, sokféle színt be lehet állítani a színpalettáról)
- Text: I know you want me (ez a felirat látható most a képernyő szürke sávjában)



A **Label1** eszközön az alábbi tulajdonságokat állítottam át:

- FontBold: pipa (félkövér betű),
- FontSize: 20 (ekkora a betűméret),
- Text: Milyen autóm lesz holnap?
- TextColor: White



Figyelem, nálunk gyakori hiba, hogy projektorral történő kivetítéskor nem látszik pl. a TextColor-nál a teljes színskála, csak az első néhány szín. Ilyenkor bátran menjünk lefelé a kurzornyílal, mert ott vannak azok a színek, csak nem látjuk őket rendesen. Persze lehet, hogy senki másnál nem fog ez előfordulni, csak nálunk 😊.

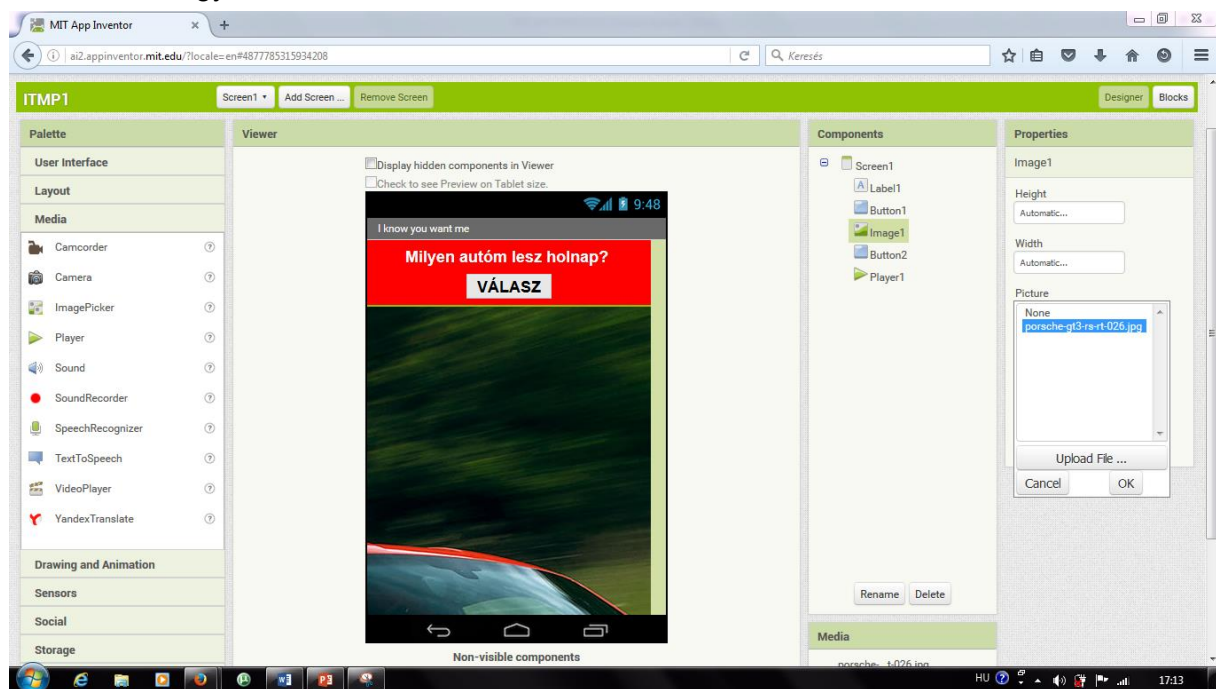
Button1 eszköz **Properties** ablakban történt átállításai az alapértelmezettől:

- FontBold: pipa (félkövér),
- FontSize: 20,
- Text: VÁLASZ

Amikor erre a Button1 (VÁLASZ) gombra kattint majd a felhasználó, akkor fog megjelenni az autó képe és ezzel egyidőben elindul a zene is

Image eszköz beállításai:

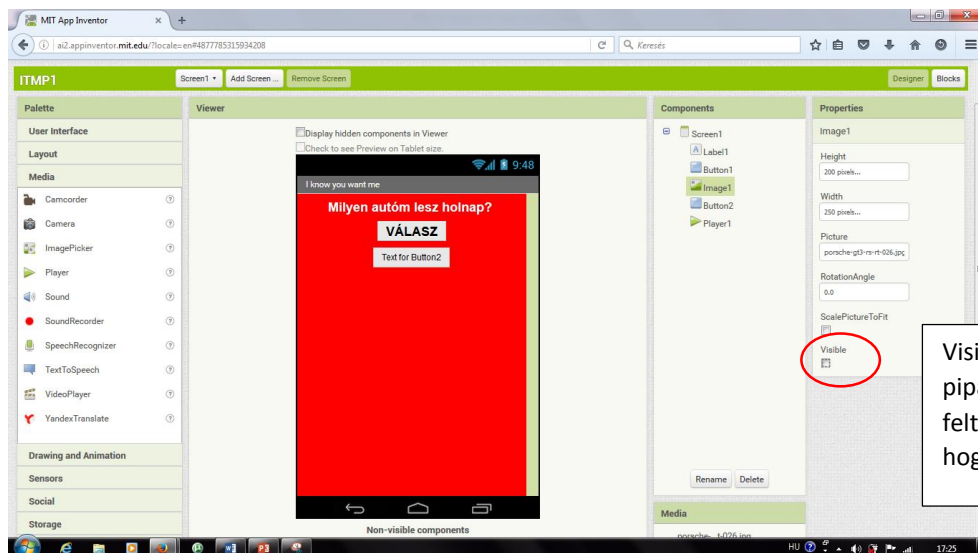
Ez az eszköz tartalmazza majd az autó képét. A képet a számítógépről tudjuk majd betallózni. Az itt látható PrintScreen-en látszik, hogy a kép sokkal nagyobb, mint amekkora elférne a mobilunk képernyőjén, direkt egy nagyobb méretű képet választottam, hogy a beállítások aztán érthetőbbek legyenek.



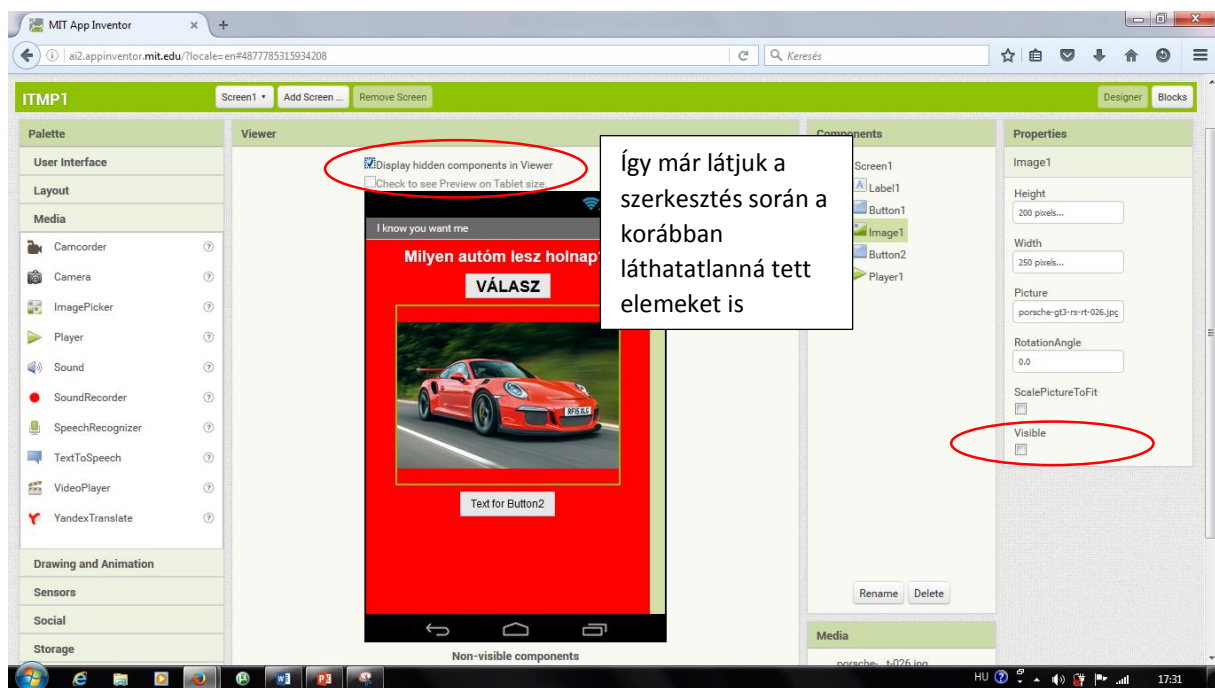
Mint láthatjuk, a képet a *Picture* tulajdonság beállításával tudjuk feltölteni, Upload File-lal egyszerűen betallózzuk a könyvtárunkból.

Ezután állítjuk át a kép magasságát és szélességét a *Height* és *Width* tulajdonságokon keresztül, én egyszerűen átállítottam a magasságot (*Height*) 200 pixelre, a szélességet (*Width*) pedig 250 pixelre. Mindkét tulajdonságnál van FillParent lehetőség, mely az eszközön rendelkezésre álló helyhez igazítja a kép méretét.

Fontos! Most állítom át az *Image1 Visible* (láthatósági) tulajdonságát is (kiveszem a Visible tulajdonság melletti pipát), hogy az applikáció első indítása előtt még ne látszódjon maga a kép. Majd később a programban fogom szabályozni, hogy mikor kell látszódnia a képnek (akkor, amikor a Válasz gombra kattintok).



Mint látjuk, egyelőre még szerkesztés közben sem látjuk a láthatatlanná tett elemeket. Nyilván ez így nem túl hatékony, így nem kell mást tennünk, mint bekapcsolni a láthatatlan elemek mutatóját a szerkesztés során (Display hidden component in Viewer), a Viewer ablakban.

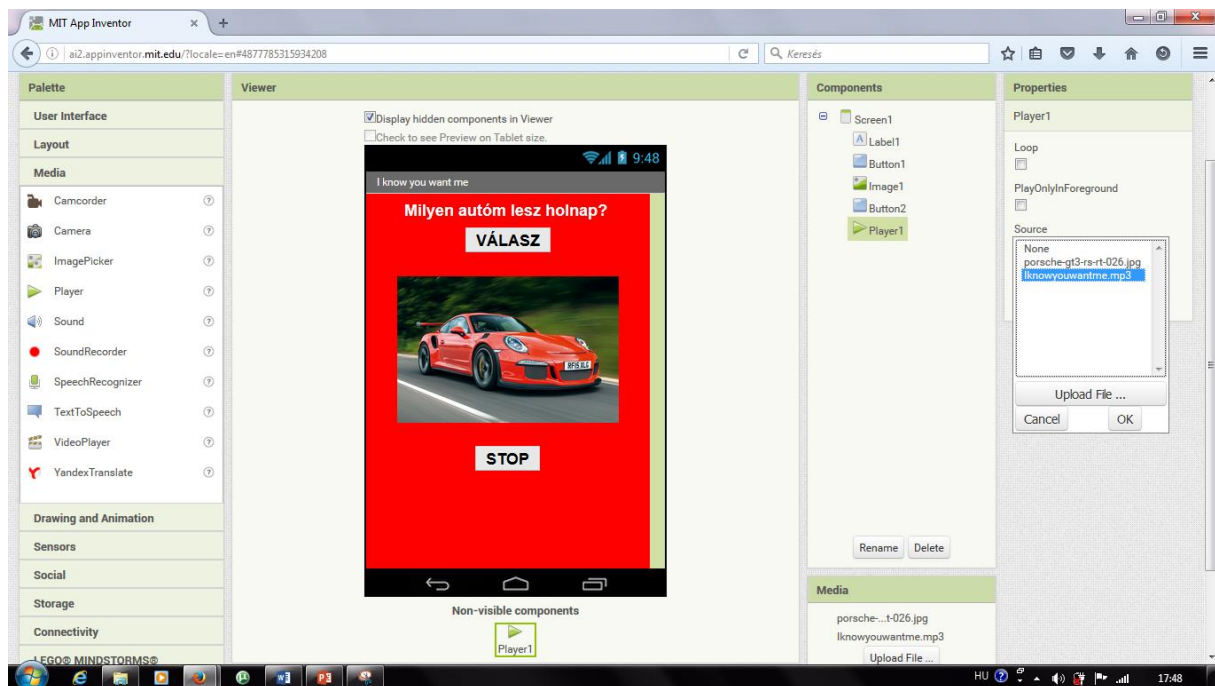


Utolsó előtti lépésként a **Button2** tulajdonságait állítjuk be.

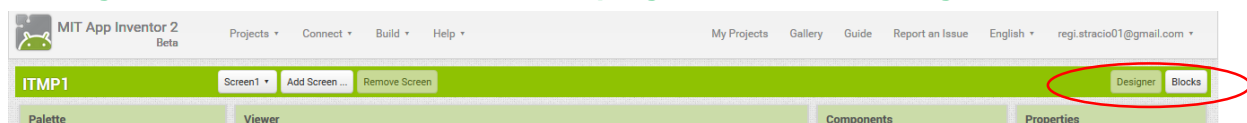
- FontBold: pipa (félkörvér),
- FontSize: 20,
- Text: STOP,
- **Visible: kieszedjük a pipát** (ez is láthatatlan az első pillanatban, majd akkor lesz látható, ha már látszik a kép és szól a zene is)

Végül a **Player1** eszköz

- Source tulajdonságánál betallózzuk a gépről a zenét (Upload file).



A Design megszerkesztése után jön a programozási rész és az is pont ilyen könnyű lesz. A programozást nem a Designer részben folytatjuk, hanem a jobb felső sarokban átállunk a Design-ról a Blocks-ra. Ez tartalmazza a programozáshoz szükséges blokkokat.



3.2.) BLOCKS használata - programozás, blokkok használata

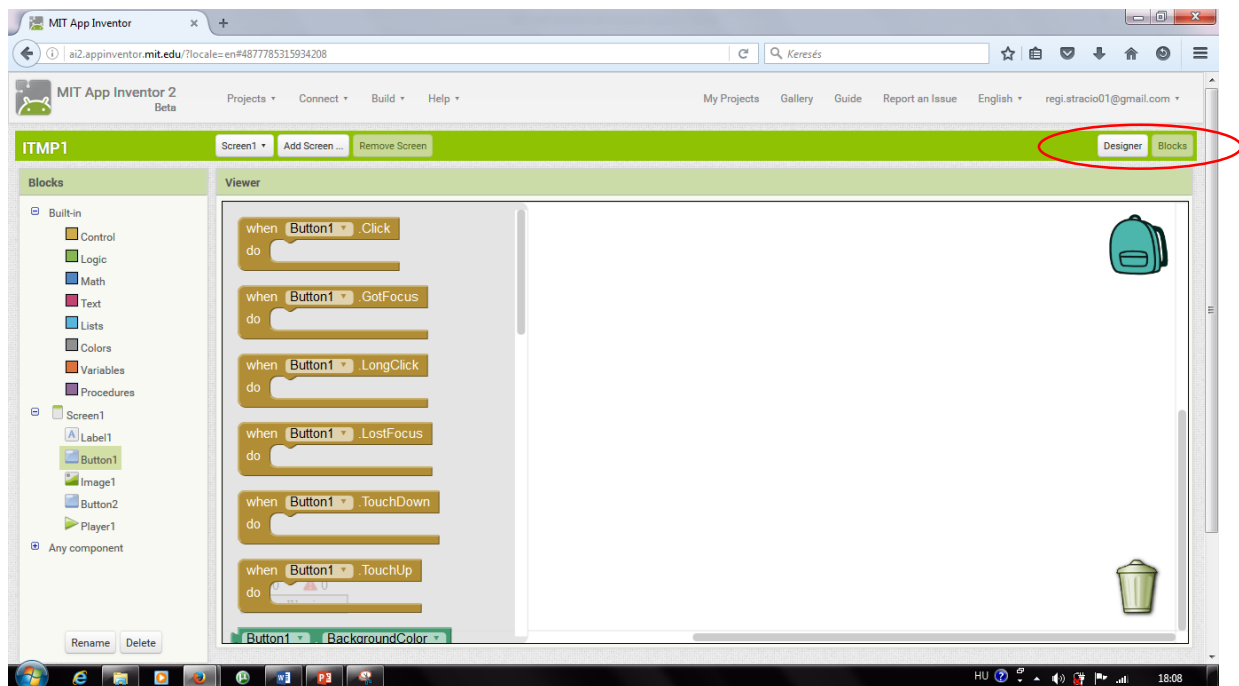
Jelen applikációnkban az alábbi feladatot kell leprogramozni:

Van egy kérdésünk (Milyen autóm lesz holnap?) és van egy VÁLASZ gomb. Amikor a VÁLASZ (Button1) gombra kattintunk, akkor kell megjelennie az autós képnek (Image1) és megszólalnia a jó hangulatú zenének (Player1). Ezzel egyidőben feltűnik a STOP gomb is.

Amikor a STOP (Button2) gombra kattintunk, akkor újra láthatatlanná kell válnia az autós képnek (Image1) és el kell hallgatnia a zenének (Player1), valamint el kell tűnnie magának a gombnak is.

Itt is a már korábban megszokott metodikát követjük, először kiválasztjuk a programozni kívánt eszközt, de itt már nem a tulajdonságait állítjuk át, hanem a hozzá tartozó programozási lehetőségek (blokkok) közül választjuk ki a megfelelőt.

Átmegegyünk a Blocks részre, majd rákattintunk a Button1 eszközre és ezt fogjuk látni:



Itt az adott eszközökhöz tartozó eseményeket, tulajdonságokat blokkok formájában találjuk meg, melyeket egyszerűen össze kell húzogatni egymással.

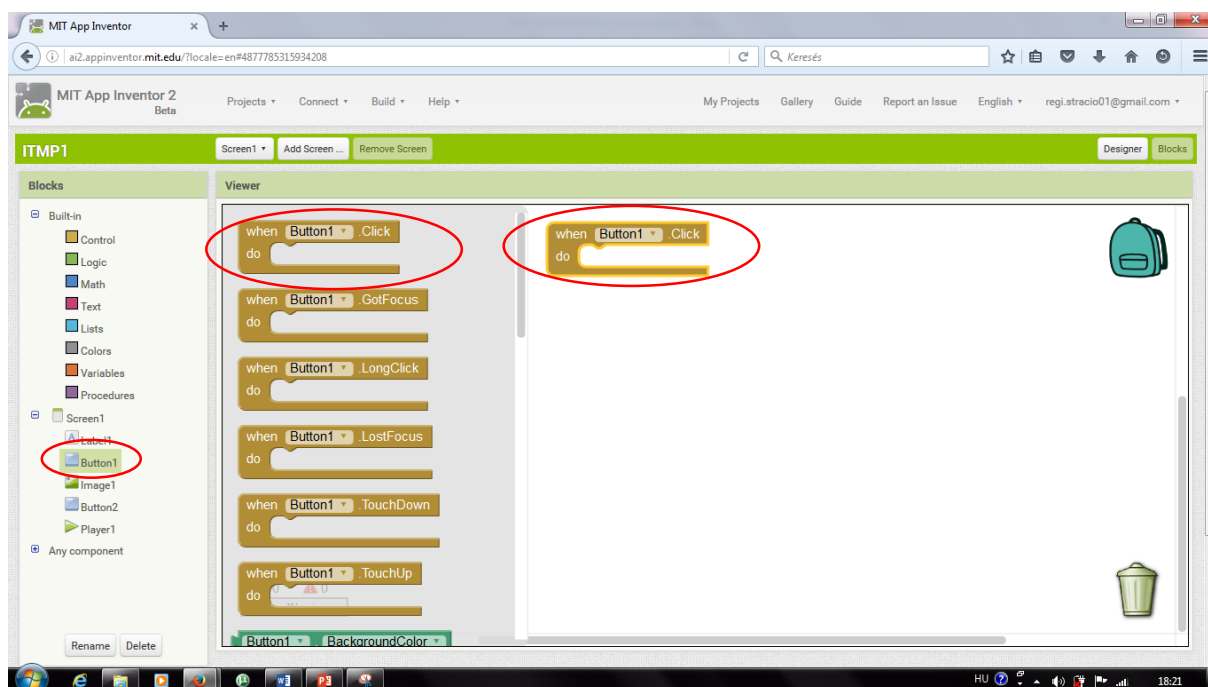
A **Blocks/Built-in** részben csoportosítva vannak a különböző lehetőségek:

- **Control** (barna színű) csoportban vannak az adatvezérlési eszközök, lehetőségek (if, if-else stb.).
- **Logic** (zöld színű) csoportban vannak a logikai kifejezések blokkjai.
- **Math** (kék) csoporttal matematikai műveleteket lehet végezni.
- **Text** csoportban a szövegekhez kapcsolódó műveletek vannak.
- **Lists** – listákhoz kapcsolódó lehetőségek.
- **Colors** – színbeállítási blokkokat tartalmaz.
- **Variables** – változók megadásánál használhatjuk.

Applikációnkhoz visszatérve most az alábbi fogjuk leprogramozni:

Amikor a **Button1**-re (VÁLASZ) klikkelünk, váljon láthatóvá az **Image1** (autós kép) és szólaljon meg a **Player1** (induljon el a zene), majd jelenjen meg a **Button2**(STOP gomb).

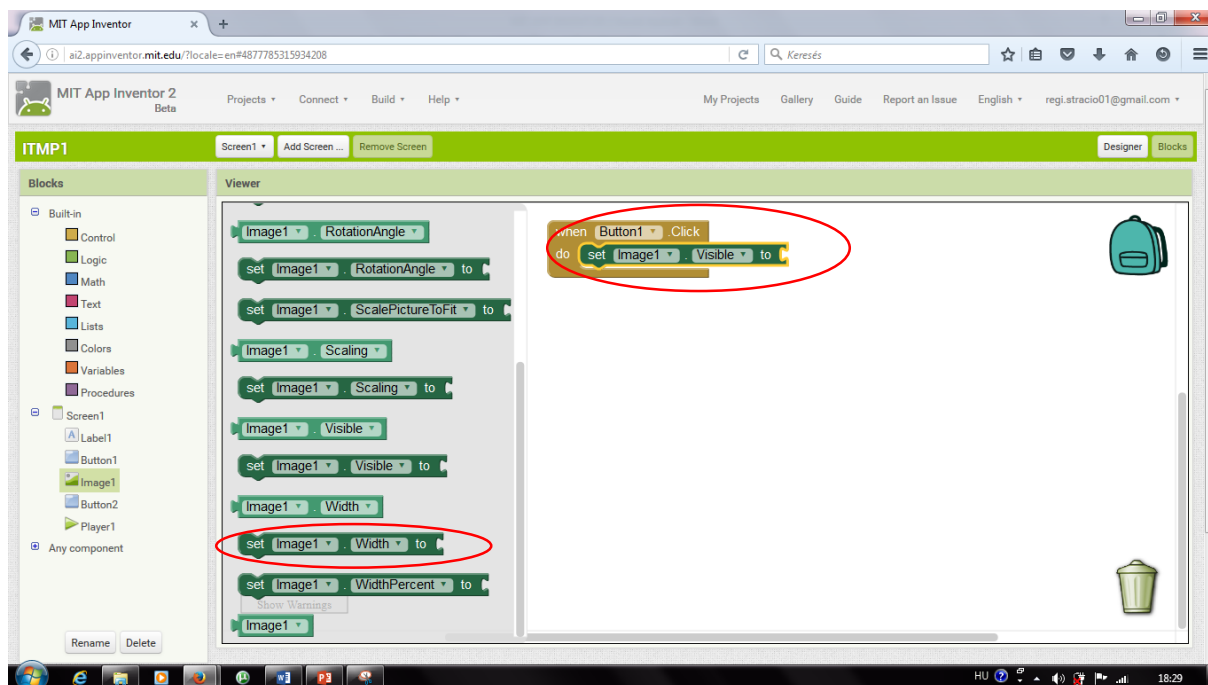
Ehhez először kiválasztjuk a **Button1** mellett szereplő lehetőségek közül a „When Button1.Click-do” blokkot és egyszerűen behúzzuk a fehér szerkesztő felületre (ezzel szabályozzuk le, hogy amikor a Button1-re klikkel a felhasználó, akkor mi történjen):



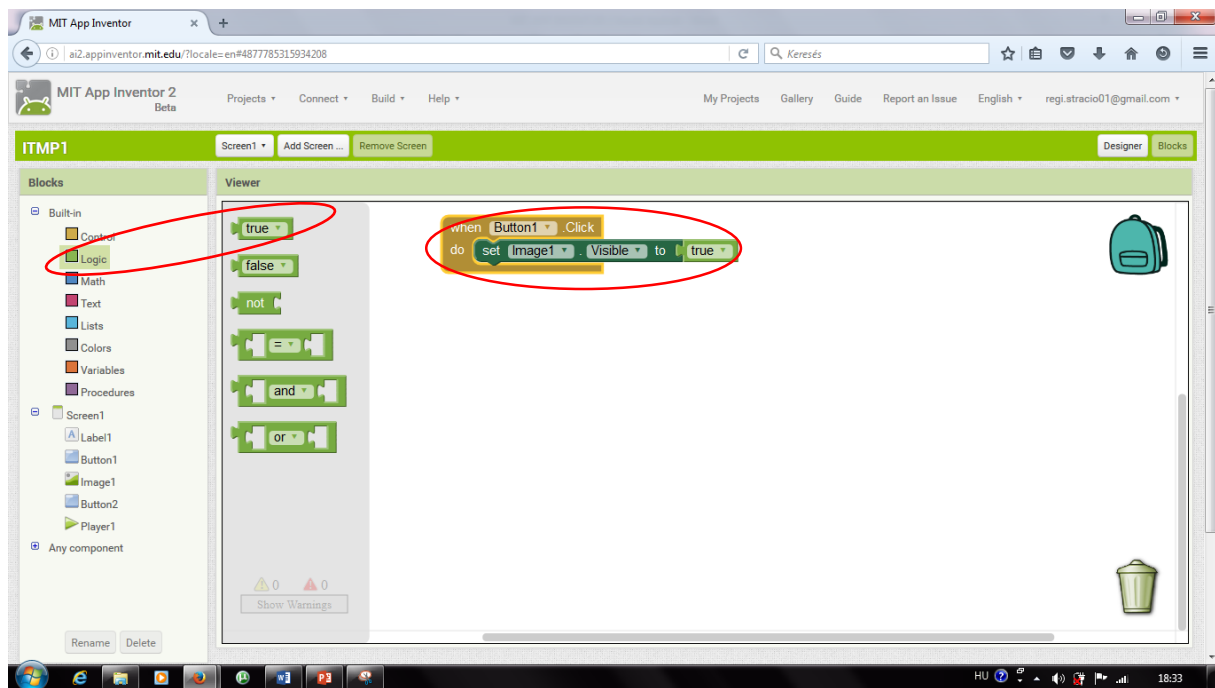
Ezek után már csak bele kell húzni ebbe az egységbe 3 programozási feladatot:

1. Az **Image1** *Visible* tulajdonságát állítsuk true-ra (hiszen eddig láthatatlan volt, most állítjuk vissza láthatóra),
2. A **Player1** indítsa el a lejátszást (hívjuk meg a Player1-et és Start)
3. Jelenjen meg a STOP (**Button2**) gomb, hiszen eddig láthatatlan volt

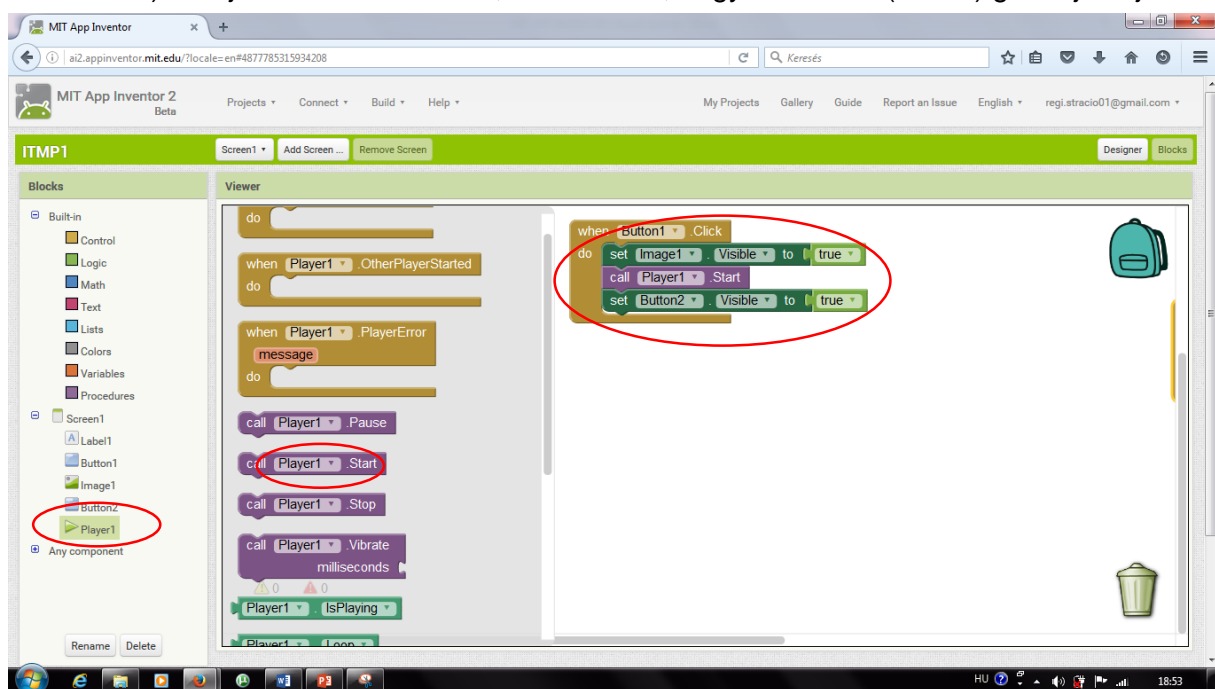
Menjünk rá az **Image1**-re és a feljövő rengeteg beállítási lehetőség közül fogjuk meg a „set Image1.Visible to” beállítást és húzzuk be a **Button1** klikk eseményébe:



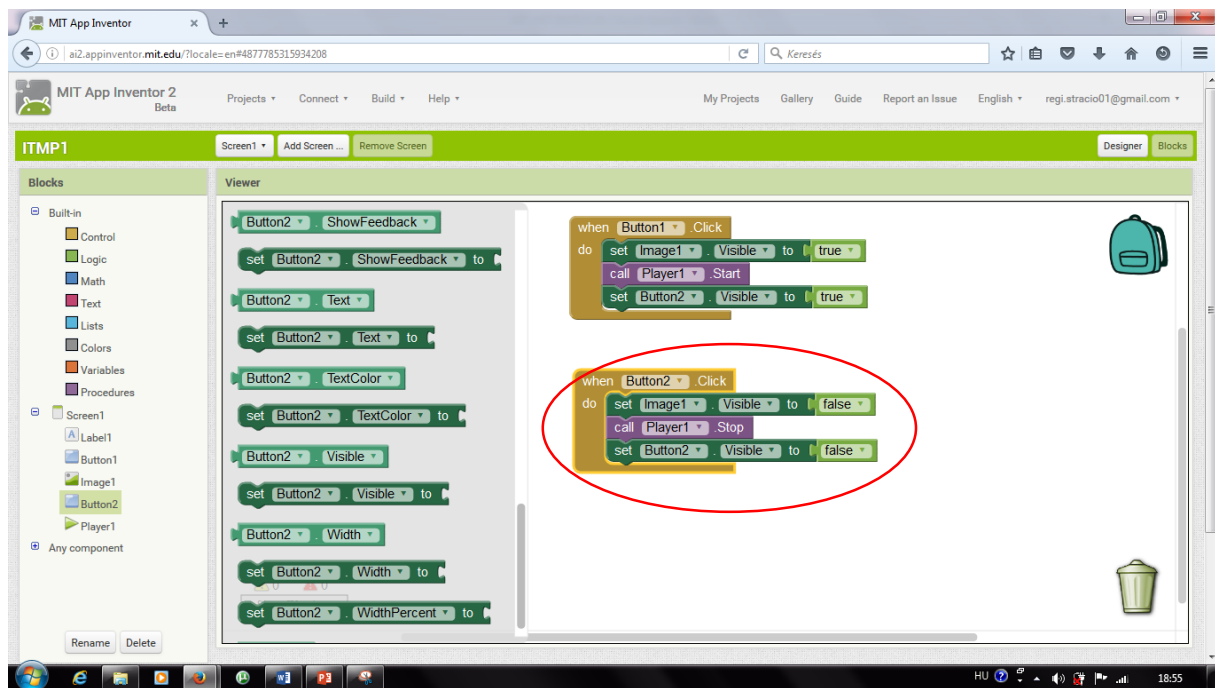
Innen is látszik, hogy a láthatóságot (*set Image1.Visible to*) még be kell állítani, itt maradt 1 üres kapcsolódási lehetőség, ide kell behúznunk a „true” logikai kifejezést, amit a **Logic** (zöld) komponenseknél találunk meg.



Ezután beállítjuk, hogy a **Player1** (ami már tartalmazza az odatallozott zenét, a Designer részénél töltöttük fel) le is játssza a számunkat, illetve azt is, hogy a **Button2** (STOP) gomb jelenjen meg.

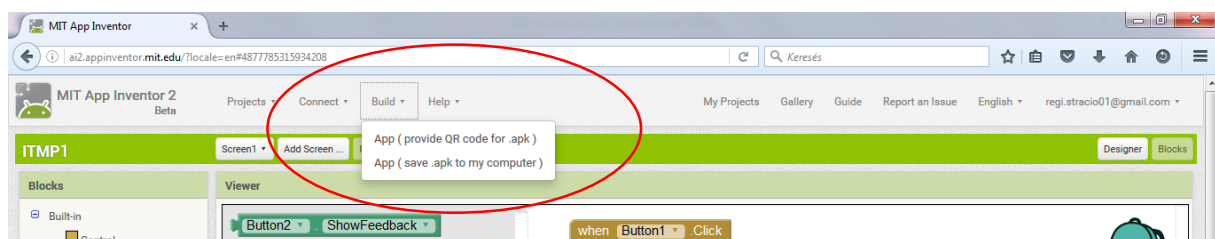


Programozásunk utolsó lépéseként a **Button2**-re (STOP) klikkelve újra láthatatlanná tesszük az autó képét (*set Image1.Visible to false*), elhallgattatjuk a **Player1**-et (*call Player1.Stop*) és láthatatlanná tesszük magát a STOP gombot is. Ezt az alábbi blokkokkal oldjuk meg:

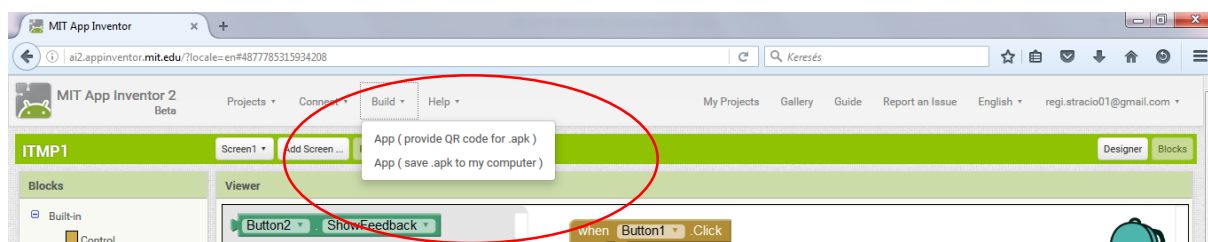


Most, hogy megvagyunk a Designer és a Blocks részekkel (kinézet és programozás), már csak applikációt kell építenünk a kis programunkból.

Az applikáció-építés a Build menü segítségével fog történni.



5.) Applikáció-építés



A programunkból 2 féleképpen lehet applikáció.

A gyorsabb út az, ha van a telefonokon QR Code olvasó applikáció. Amennyiben van, úgy egyszerűen válasszuk ki az „App (provide QR code for .apk)” fenti menüpontot, a program generál egy QR kódot, amit a telefonnal beolvasva, az applikáció ott „terem” a telefonon.

Tapasztalataink szerint kevés gyereknek van ilyen applikációja és nem is annyira szívesen töltik le, így mi többnyire a második menüpontot használjuk (App (save .apk to my Computer)).

A rákattintás után egy apk (applikáció) fájl töltődik le a gépünkre (nem kevés idő alatt!!), beállításától függően többnyire a Letöltések könyvtárba. Innen adatkábel segítségével egyszerűen átmásolhatjuk a gyerekek telefonjára az applikációt, majd telepítés után elindíthatjuk a telefonról az applikációt. (Érdemes a telefonunkon egy Android_app mappát létrehozni, ahova a saját készítésű applikációinkat tehetjük fel.)

Nos, itt azonban elsőre egy csomó hibába futhatunk bele ☺ :



A gyerekek telefonján sokszor nincs Fájlkészítő alkalmazás, pedig az fontos lenne nem csak ehhez, hanem a későbbiek során is. Vegyük rá őket arra, hogy letöltsenek a Google play-ről egy fájlkészítő alkalmazást, mert csak így tudjuk az .apk fájlokat állományként kezelni (másolni, törölni, áthelyezni stb).



A telefonok többsége először nem engedélyezi a saját készítésű applikációk telepítését, hiszen a helyes beállítású telefonok alapban csak a Google Playről engednek ellenőrzött applikációkat telepíteni. Így miután átmásoltuk az applikációkat a telefonra a számítógépről és el akarjuk indítani az applikációt, megjelenik egy olyan üzenet, hogy idegen applikációkat nem telepíthetünk, ehhez engedélyezni kell a nem Google Play-ről származó applikációk telepítését. Itt követnünk kell a lépéseket és a végén, amikor már mindezt engedélyeztük, hát akár el is indíthatják a gyerekek a saját applikációjukat.

Egyébként az a személyes tapasztalatunk, hogy a diákok sokszor jobban ismerik a telefonokat és az ahhoz tartozó engedélyezési lehetőségeket, mint mi tanárok, így simán előfordulhat, hogy az egyik gyerek hamarabb fog segíteni a másinak, mint ahogy mi odaérünk ☺.

És végül egy utolsó személyes tapasztalat...



Sok diáknak nem Androidos telefonja van, hanem Iphone-ja (és néhánynak Windows phone-ja). Az első gondolatuk az, hogy nekik ezt miért is kéne csinálni, hiszen ők Iphone-osak. Ilyenkor megbeszéljük, hogy nagyon szerencsések az Iphone-nal, de azért gondolkozniuk és programozniuk nekik is kell ☺. Ezen kívül van Emulátor is az MIT App Inventor2-höz (<http://appinventor.mit.edu/explore/ai2/setup-emulator.html>), melynek telepítésével a gépen létrehozhatunk egy pszeudo applikációt az Iphone-osok (vagy bárki) részére, hogy lássák ők is, hogyan halad az applikációjuk. Nálunk sajnos a gépek nem bírják az emulátort, nagyon lassan, nagyon dögögve megy és sokszor meg is hal tőle a további munkalehetőség (és a gép is).

De a szerencsésebb gépparkkal rendelkező iskolákban ez is használható módja az applikáció folyamatos tesztelésének, hogy a gyerekek lássák, miként alakul és működik az applikáció, amit megcsináltunk.