Kölcsey Ferenc Főgimnázium

Egyszerű mobil alkalmazások készítése

Készítette: Bauer Kristóf

XII.A

Felkészítő tanár: Kui Rita

Tartalom

1. Bevezető
2. Az applikáció ismertetése
3. Fejlesztői környezet
4. A kód részletes magyarázata
5. Rendszerigény
6. Hivatkozások

Bevezető

Dolgozatom céljául, egy olyan alkalmazást szerettem volna megírni, amely egyszerre edukatív és interaktív így a felhasználók könnyedén és hamar tanulhatnak általa.

Fontos volt számomra, hogy a mobil platformot válasszam, ugyanis manapság mindig nálunk vannak telefonjaink, így bármikor használhatjuk az adott applikációt.

Az alkalmazásom célközönsége főleg az iskolások és egyetemisták.Nekik akartam lehetőséget biztosítani arra, hogy szabadidejükben is könnyen ismételhessék a képleteket tankönyvek nélkül.

Az applikáció ismertetése

Sokak számára fejfájást okoz a matek képletek „magolása”. Az általam megírt alkalmazás amelynek neve „Matek Trivia” épp ezt szeretné egy barátságosabb módon begyakoroltatni.

Az alkalmazás megjelenít egy trigonometriai képletet, majd megkéri a felhasználót, hogy adja meg a helyes választ. Miután a felhasználó leírta válaszát, ellenőrzi hogy egyeznek-e a képletek. Ha helyes, a felhasználó kap egy pontot, ha helytelen akkor megjeleníti a teljes képletet és új kérdést kap. A kérdések véletlenszerűen jönnek, így akár többször is ismétlődhet egy.

Ahogy dolgozatom címéből is látható célom volt, hogy az applikáció egyszerű és átlátható legyen úgy a grafikus felületében mint a kódban. Bárki aki egy alap tudással rendelkezik könnyedén megérti és módosíthatja azt.

Fejlesztői környezet

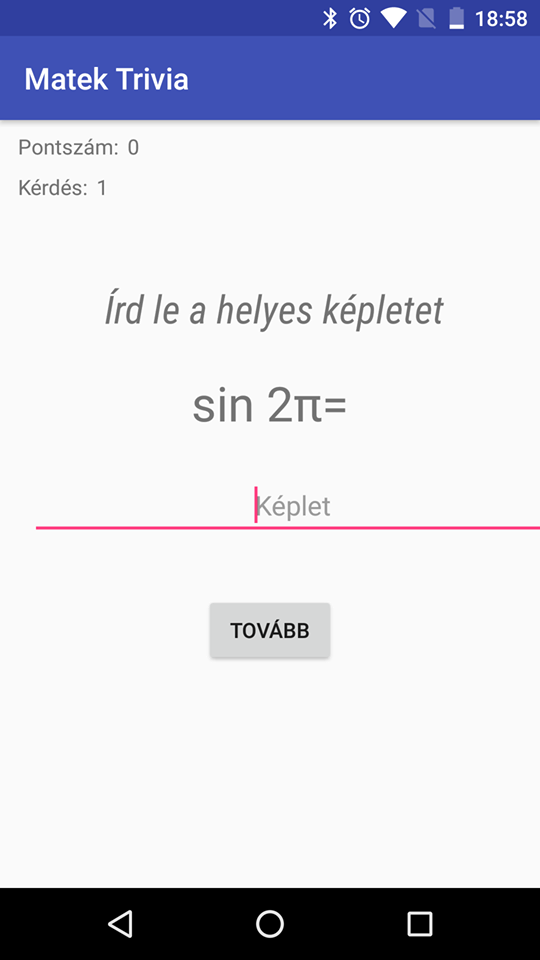
Az androidos alkalmazások mind java nyelven íródnak, de mi is az a java? A java egy olyan objektum orientált programozási nyelv amelynek nagy előnye, hogy platformfüggetlen vagyis a kocsik, TV-k, okosórák, mobilok és egyéb platformokon is megtalálható.

Ahhoz, hogy a java nyelven írt kódunk fusson először le kell töltsük az Oracle oldaláról Java Development Kit-et (JDK), ez tartalmazza a Java Virtual Machinet(JVM) ami egy virtuális gép és feladata a platformfüggetlen kód futtatása. A másik amit tartalmaz a JDK az a Java Runtime Enviorment(JRE) amely a JVM implementációja és ez már lehetővé teszi magán a fizikai gépen vagyis az adott platformon a kódunk futtatását.

A Google ahhoz, hogy segítse az androidos fejlesztők munkáját létrehozták az Android Studiot. Ennek segítségével könnyedén megtervezhetjük a grafikus felületet amelyet XML kódban írunk és a studion belül megírhatjuk Java kódjainkat. Az Android Studio javasol virtuális gépeket is amiken futtathatjuk a kódjainkat és ugyanakkor lehetővé teszi azt is, hogy telefonunkat a gépünkhöz hozzákötve USB-n keresztül valós időben használjuk és ellenőrizzük kódunkat, így az Android Studio egy kötelező program az androidos mobil alkalmazáso fejlesztésében.

A kód részletes magyarázata

Az alkalmazás létrehozásakor először megterveztem a grafikus felületét. Papírra lerajzoltam a kinézetet, majd különböző TextView (szövegek), ButtonView(gombok) és elrendezések (linear layout) kombinálásával megírtam az XML kódot.



Amiután megterveztem a kinézetet áttértem a java kód megírására.

Mivel az Android Studio alapból létrehoz egy main activity classt (aminek a neve a projectünk neve vagyis MatekTrivia) ahol a program összes class-je lefutásra kerül. Mivel egyszerű mobilalkalmazást írtam ezért nem alkalmaztam külön classseket, hanem a MatekTriviába írtam az összes kódom.

Az kód elején lévő „import” szavak azokat a packageket jelöli amiket az Android Studio autómatikusan implementált mivel, különböző objektumokat használtunk amikhez tartoznak előre definiált metódusok ezeket pedig azok a packagek tartalmazzák.

**int kerdes**=1;  
**int pont**=0;  
**int i**=0;

Ezek után létrehoztunk három globális változót. A kerdes számolja az eddigi feltett kérdések számát, a pont az eddigi eltalált kérdéseket az i pedig későbbiekben random számot generál, hogy a kérdések véletlenszerűen tevődjenek fel.

String[] **foKerdes** = **new** String[] és String[] **foValasz** = **new** String[]

Ez a két tömb stringeket tartalmaz. Az elsőben a kérdések vannak a másodikban pedig a válaszok. Mivel tömböket használunk bármikor könnyedén több képletet is hozzáadhatunk az appunkhoz.

TextView kerdesTextview = (TextView) findViewById(R.id.***keplet***);

Mivel a pontszám (a helyes válaszok) valamint maga a kérdés és a válaszok száma folyamatosan változni fog ID-kat rendelünk hozzá, hogy amikor azok változnak az XML kódban is tudjuk változtatni szövegeiket.

Az onCreat metódus alatt inicializáljuk a többi változót is és megjelenítjük az első kérdést.

Random rand = **new** Random();  
**i** = rand.nextInt(57) + 1;

kerdesTextview.setText(**""**+**foKerdes**[**i**]);

Mivel 57 kérdésünk van 57 közül véletlenszerűen választunk egy kérdést és megjelenítjük azt.

**public** String GetMessage() {

A GetMessage metódusunk egy alapból definiált metódus, bekéri a felhasználó által beírt szöveget.

Mikor a felhasználó beírta a szövegét és rákattint a tovább gombra egy újabb kérdésért meghívódik az általunk megírt

**public void** tovabbiKerdes(View view) {

metódus amely meghívja az

**public void** ellenoriz()

metódust. Ez ellenőrzi, hogy helyes-e a kérdés. Úgy nézi meg, hogy a kérdés helyes-e, hogy a főválasz vektor i. pozicióján lévő string, ami a helyes válasz megegyezik-e a felhasználó által beírttal. Ha a válasz helyes, az ellenőriz metódus kiíratja a sikeres szöveget, ha pedig nem akkor kiíratja a helyes megoldást.

Ezek után a kódunk visszatér a továbbikérdésre, randomolódik egy szám és a főkérdés vektorból a főkerdes[i]. pozíción lévő string kiiratódik.

Rendszerigény

Mivel a kód igen egyszerű és kevés memóriát használ el **bármilyen** androidos rendszeren futtatható, így nincs hozzákötve hogy mennyi a minimum Androidos rendszerigény amin futtatható, ezt a fejlesztő állíthatja be fakultatívan.

Hivatkozások

* <https://developer.android.com/>
* https://www.javatpoint.com/