

JegyKalk

KGHSZ

Kocsis Kamilla, Cseri Gábor

Felkészítő tanár: Brósch Éva

*Berettyóújfalui Szakképzési Centrum Közgazdasági Szakgimnáziuma,
4200 Hajdúszoboszló, Gönczy Pál u. 17.*

1. Bevezetés

Szorgalmas, mindenre felkészült; csendes vagy éppen hangos; sosem csinál házi feladatot, vagy majd óra előtt, éppenséggel a buszon ülve; aki egy ötös alát is ki szeretne javítani, de olyan is akad, aki még az egyest is alig, a diák az diák. És minden tanuló életében megtörténik az, mikor a jegyeire kell néznie.

Valaki úgy szereti, ha napra pontosan tudja átlagát, de akadnak olyanok is, akik féltve, évvége utolsó hetén akar csak bosszankodni ezen. Ez mind nem számít, hisz mi mindenkinek segíteni próbálunk, akiknek tudását jegyekben mérjük.

Bevalljuk őszintén még a legelején: bármennyire is egyszerű, nem szeretünk átlagot számítani. Ami nem annak köszönhető, mert a jegyeink csak nagy ritkán akarnak az ötösre hasonlítani, hanem mert unalmas és hosszadalmas. Mennyire egyszerű lenne, ha esténként nem számológéppel a kezünkben kéne számolgatnunk az érdemjegyeink átlagát, kiderítve így, hogy mennyit kell még letenni az asztalra, hogy elérjük az elvárt szinteket. Helyette inkább egy forró teával a kezünkben beülnénk a kényelmes fotelba, egy izgalmas kalandregénnyel és nem hagynánk hátra a telefonunkat sem. Ugyanis ezt a borzalmasan unalmas és fárasztó munkát elvégzi a telefonunkon található alkalmazás is.

2. Probléma megoldásának menete

A kitűzött cél egy könnyen kezelhető, praktikus és főként számunkra soha nem látott program elkészítése volt. Kezdetben két fő feladat megvalósítását terveztük el, ezek a számolások megírása, tesztelése, valamint az adatbázis felépítése volt. Ezt a két feladatot felosztottuk egymás közt.

2.1. Számolások

A matematikai számolásokat először Netbeansben írtuk meg, ekkor még konzolon keresztül írtattuk ki és kértük be a szükséges adatokat.

2.2. Adatbázis

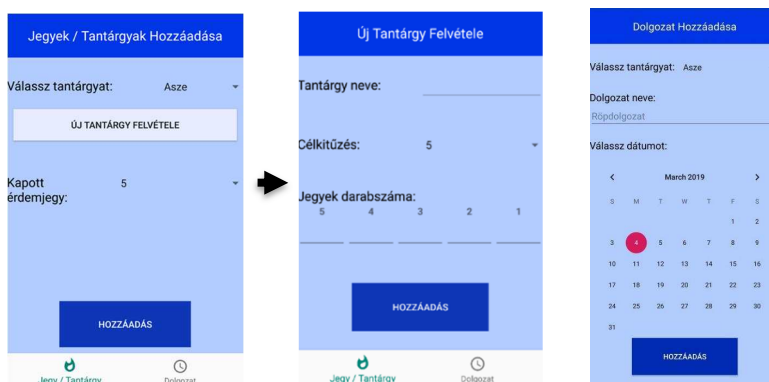
A programunk adatbázisát SQLite-ban írtuk. Az adatbázis megírása során törekedtünk arra, hogy kevés tárhelyet foglaljon. Igyekeztünk elkerülni benne a redundanciát és a vele járó anomáliákat.

2.3. A program működése

A JegyKalk főmenüjében 4 opció közül választhatunk. Mindegyik lehetőség különböző feladatok ellátására szolgál.

2.3.1. Hozzáadás menüpont

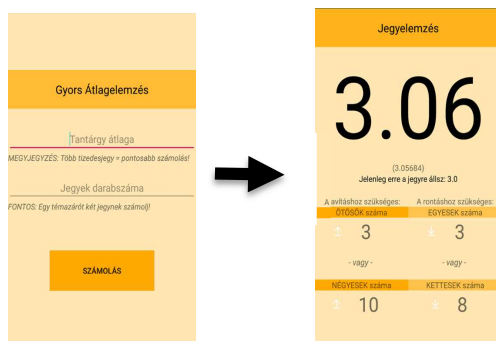
A program működésének megkezdése előtt szükséges felvenni azokat a tantárgyakat és a hozzá fűződő jegyeket, dolgozatokat, amelyeket szeretnénk, ha tartalmazna a program adatbázisa. Azonban a JegyKalkban nem szükséges azonnal megadnunk ezeket az adatokat, ahhoz, hogy elmentésre kerüljön a tantárgy.



1. ábra: Hozzáadás kinézete

2.3.2. Gyors Számlálás menüpont

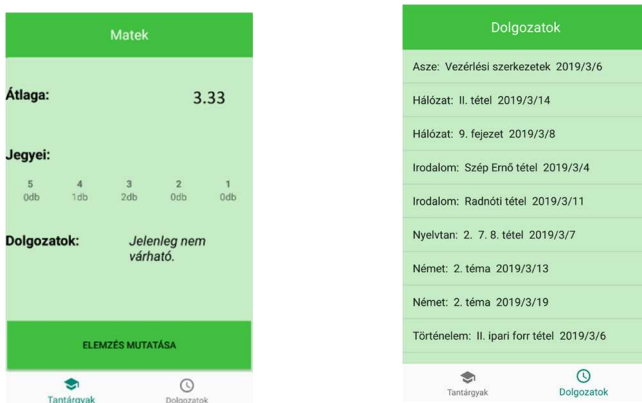
Mi történik akkor, ha valaki nem akarja napról napra végigvezetni tanulmányait, viszont szeretné tudni, hogy milyen közel vagy éppen távol áll ahhoz, hogy átlaga megváltozzon? Erre a problémára is megoldást találtunk ebben a menüpontban. Ha a felhasználó megadja a számolásokhoz szükséges adatokat, egy kattintással megtudhatja a választ égető kérdésére.



2. ábra: Gyors számlázás kinézete

2.3.3. Jegyek és Dolgozatok menüpont

Ebben a menüpontban tekinthetőek meg a már felvett tantárgyak jegyei és a hozzá kapcsolódó dolgozatok listája. Ha kiválasszuk a tantárgy nevét a listában, megjeleníthető az adott tantárgyhoz fűződő adatok, ha pedig az Elemzés gombra kattintunk, azokat az adatokat láthatjuk, amelyen az egész program fő célja és haszna nyugszik. Ha a Dolgozatok almenüt válasszuk, kilistázva tekinthető meg a ránk váró dolgozatok, valamint ahhoz a tantárgyhoz tartozó célkitűzés és átlag.



3. ábra: Jegyek és Dolgozatok kinézete

2.3.4. Törlés menüpont

A felhasználó a Törlés menüpontban ismét kilistázva a felvett adatokat a tantárgy fontosságának csökkenése vagy éppen a tantárgy megszűnése miatt könnyen és egyszerűen, egy kattintással kitörölheti.

2.4. Nehézségek

Számunkra az Android Studio teljesen ismeretlen volt, így nem tagadás, hogy számos nehézséggel kellett megküzdenünk, valamint ennyire összetett adatbázist se állítottunk még össze. Annak ellenére, hogy a program megírásához nélkülözhetetlen tudásnak csak az egyik részével: az objektum-orientált programozással voltunk tisztában, úgy gondoljuk sikerült az elvárásoknak megfelelően teljesíteni.

Az első nehézségeket az adatbázis és a felület összekötése jelentette, majd az, hogy a Fragmentek megfelelően kommunikáljanak egymással. Ehhez a Bundle-t használtuk.

A következő nehézséget a felület kialakításában találtuk, hisz fontos, hogy jól érthető és könnyen kezelhető legyen.

3. Elért eredmények

Az első hetekben autodidakta módon igyekeztük megtanítani magunkat az Android Studio használatára, ennek köszönhetően rengeteg egyéb programot megírtunk a könnyebb megérthetőség végett. Úgy gondoljuk megérte minden fáradságot, hisz fontos szempontjaink közé tartozott a könnyebb elérhetőség, így jó ötlet volt a mobiltelefonokat megcélozni. Mindazok mellett, hogy fontosnak tartjuk a kinézetét és az egyszerű kialakítását, a programot igyekeztünk a legegyszerűbben megoldani. Noha a program alapja az átlagszámítás, annál sokkal több mindent képes kiszámolni és erre is szerettük volna a figyelmet irányítani. A tantárgy jegyeinek elemzésével olyan fontos dolgokra figyelhetünk, mint az átlag rontásának veszélye, a javítás közelsége, valamint a dolgozatok időpontjának lementésével a felkészülés észben tartása is könnyebbé válik.

Mind a hibák észrevételét és a felület egyszerűbb felépítését diákoknak köszönhetjük, akik mai napig rendszeresen használják programunkat, és mindig igyekszünk az ők véleményük alapján módosítani azt.

Visszatekintve az elmúlt hónapokra, számos fontos dolgot tanultunk, mind az objektum-orientált programozásban, mind a megbeszélések, ötletcserék szervezésében és tovább erősíthettük a csapatmunkában való képességeinket. Úgy gondoljuk, programunk többet adott, mint lexikális tudás, hisz jó érzés olyat készíteni, amire környezetünkben nagy szükség van.