



SAPIENTIA
ERDÉLYI MAGYAR
TUDOMÁNYEGYETEM
Marosvásárhelyi Kar

Time Management Android Application

Szoftverfejlesztés

**Hamza Balázs
Tóth Ádám
Infó 3. év**

Tartalomjegyzék

1.Bevezető.....	3
2.Célkitűzés(Target).....	4
3.Munkakör felosztás.....	6
4.Programozási eszközök.....	7
a. Segédeszközök.....	7
b.Firestore adatbázis.....	8
c.Programozási nyelvek.....	9
5.Diagrammok.....	10
6.Design.....	13
7.Továbbfejlesztési lehetőségek.....	16
8.Összefoglalás.....	17

Bevezető

Az aplikációnk az emberek időbeosztási problémájára szeretne egy megoldást ajánlani az által, hogy egyszerűen követhetővé teszi a napi teendőket bárki számára.

Mindazonáltal, hogy követni lehet a napi teendőket, az aplikáció segítséget nyújt a teendők többféle rendezésében is idő, prioritás stb. kritériumok szerint. Egyszerű és intuitív a használata, nem igényel semmilyen különleges ismeretet, egyszerűen a felhasználónak csak hozzá kell adnia a megfelelő dátumhoz a kívánt eseményt(event) és utána már meg is fog jelenni a felhasználó főoldalán az adott esemény.

Az aplikáció csak android eszközökre lett implementálva de későbbiekben importálható akár WEB vagy IOS applikációknak is.

Célkitűzések(Target)

Az applikáció tervezésekor próbáltunk minél tisztább és egyértelműbb célokat megfogalmazni annak érdekében, hogy reálisan előre tervezhető időn belül meg is tudjuk azokat valósítani.

Az első és egyben a legfontosabb célunk az volt, hogy minél hamarabb érjünk el egy működő regisztrációs felülethez melyet a bejelentkezési ablak kövessen ahhoz, hogy egy felhasználó ha nem is tud még funcionalitást használni az applikációba tudjon belépni problémák nélkül.

Az előző lépés megvalósításához kapcsolódik a második célunk is, mely egy adatbázis létrehozásában áll melyet össze kellett kötnünk a regisztrációval és bejelentkezéssel.

Ezt követte az adatbázis összekötése az aplikációnk 'backend'-jével további bővítésekhez.

Célkitűzések(Target)

A következő cél az egy esemény az adatbázishoz való sikeres hozzáadásban fogalmazódott meg a kódot használva. Amelyet az interfészen való esemény hozzáadás implementálása követett.

Ha eddig mindent sikerült megvalósítani akkor következnek a komplexebb feladatok, az első komplex feladat az adott dátumon lévő események kilistázásában fogalmazódott meg.

Az előző célunk bővítése állt az új feladat mögött, az események több módon való rendezése és kilistázásának implementálása.

Ezt követi az applikáció interfészének a 'design'-olása, illetve minimális funkcionálisokon való simítások.





Munkakör felosztás

Az első elképzelésünk az volt hogy Ádám fogja implementálni az applikáció ‘backend’-jét, illetve Balázs dolgozik a ‘frontend’-en. Az adatbázis használata, létrehozása stb. az közös munkaelosztásban részesült.

Nem sikerült teljes mértékben betartani, megvalósítani az első elképzelésünket, mert több olyan probléma is adódott amihez mindkettőnk tudására, segítségére szükség volt, de ezt inkább pozitívumnak említenénk meg, mert abban egyetértünk, hogy hasznos volt a problémák megbeszélése akkor is ha nem feltétlen a saját munkakörünkbe tartozott.

Programozási eszközök

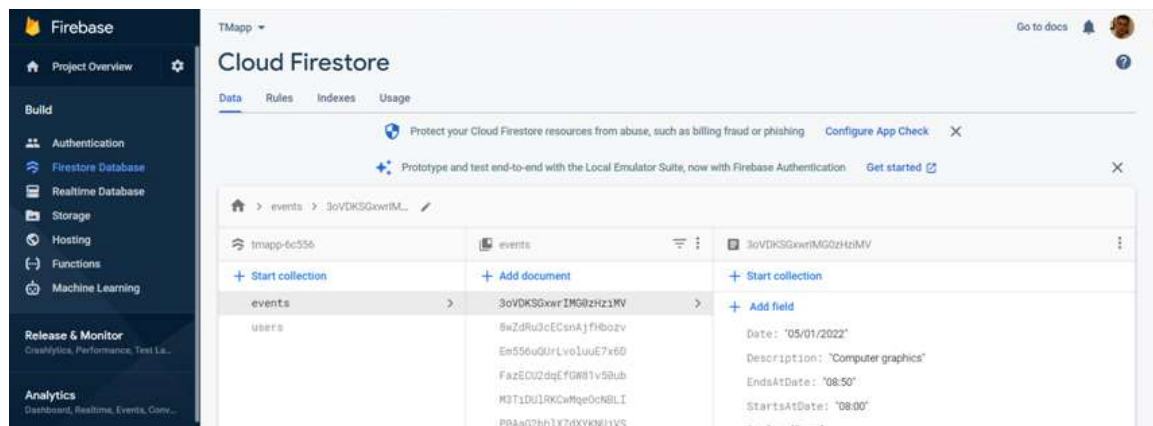
a. Segédeszközök:

- Jira-Software a projekt managementhez(kanban board, issue tracking, progress visualization)  **Jira Software**
- Github(git) a verziókezeléshez (branching, synchronized development, prevention for bug introduction into codebase)  **git**
- Android Studio az codebase implementálásához, kompiláláshoz, android készülék szimulálásához emulátor segítségével, debuggoláshoz, illetve az applikáció teszteléséhez 
- Firebase - Cloud Firestore webes felületen a felhőben eltárolt adatbázishoz, NoSQL adatbázis. 

Programozási eszközök

b. Firestore adatbázis:

- NoSQL adatbázis kezelő rendszer
- Könnyű integrálás az Android Studio-val, egy beépített modul letöltése elég az integráláshoz és működéshez.
- Egyszerű megoldást nyújt a regisztráció illetve bejelentkezéshez(Firebase Authentication).
- Google termék, tökéletes kisebb projektek megvalósításához.
- Hátránya, hogy nehezebb az adatok elérése a kódból egy SQL adatbázishoz képest.



Programozási eszközök

c. Programozási nyelvek:

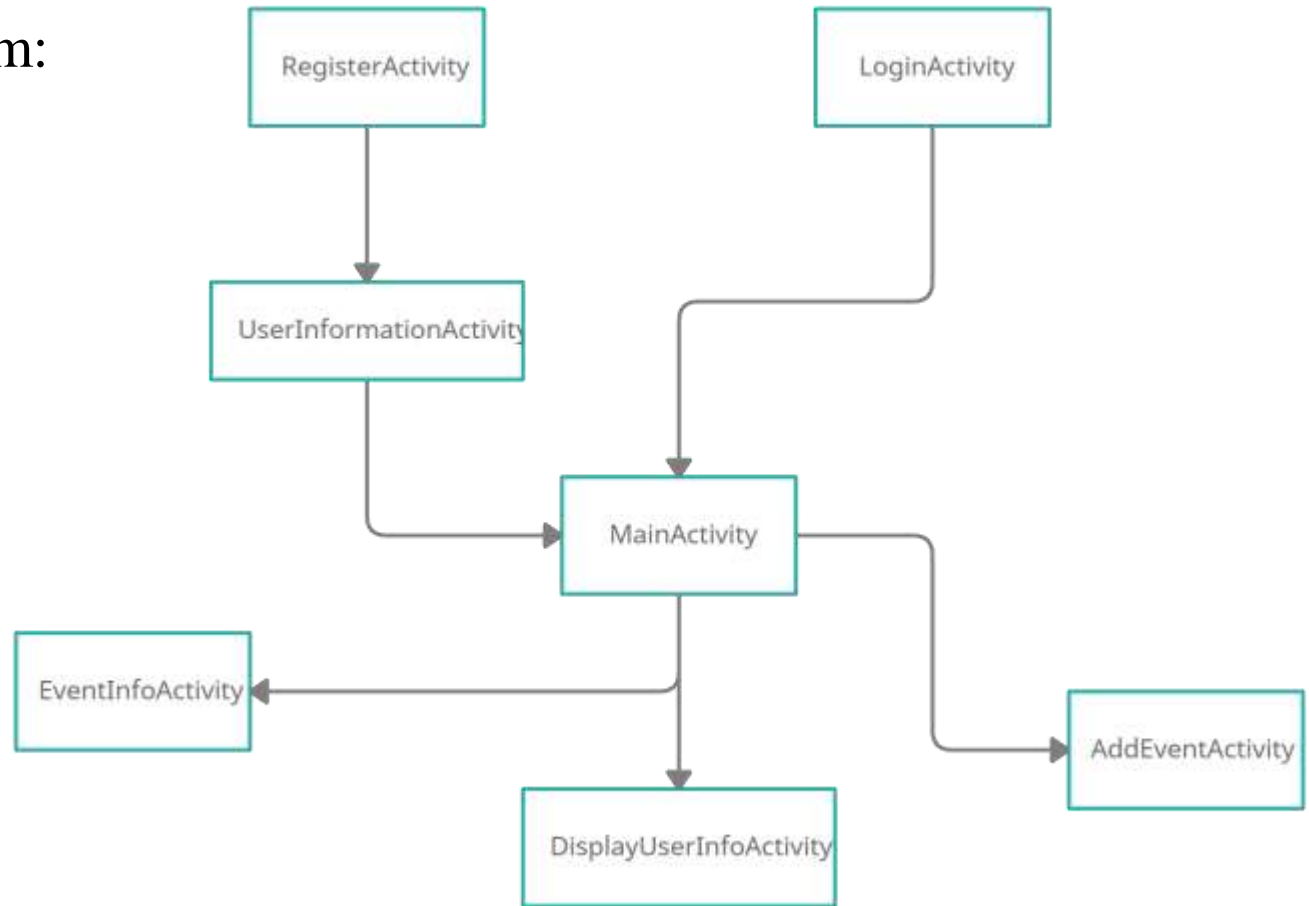
- Java backend
- XML frontend

```
MainActivity.java | Event.java | AddEventActivity.java | activity_event_info.xml | EventInfoActivity.java
eventsInZones.get(zone, SAIRI/MUROMBIE).add(zoneEvent);
break;
case MURESENI:
    eventsInZones.get(zone, MURESENI).add(zoneEvent);
    break;
case CENTAU:
    eventsInZones.get(zone, CENTAU).add(zoneEvent);
    break;
default:
    System.out.println("Invalid zone for an event");
}
}
//Create a new list of events with all of the zones already in sorted order by zones and by priority
List<Event> eventListSortedByZones = new LinkedList<>();
for (Zone key : eventsInZones.keySet()) {
    for (int i = 0; i < eventsInZones.get(key).size(); i++) {
        eventListSortedByZones.add(eventsInZones.get(key).get(i));
    }
}
```

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".EventInfoActivity">
8
9
10
11 <TextView
12     android:id="@+id/sortByTimeView"
13     android:layout_width="wrap_content"
14     android:layout_height="wrap_content"
15     android:onClick="sortByTime"
16     android:text="Sort By Time"
17     app:layout_constraintStart_toStartOf="parent"
18     app:layout_constraintTop_toBottomOf="@+id/sortByZoneTimeView" />
19
20 <TextView
21     android:id="@+id/sortByZoneTimeView"
22     android:layout_width="wrap_content"
23     android:layout_height="wrap_content"
24     android:onClick="sortByZoneTime"
25     app:layout_constraintStart_toStartOf="parent"
26     app:layout_constraintTop_toBottomOf="@+id/sortByTimeView" />
27 </androidx.constraintlayout.widget.ConstraintLayout>
```

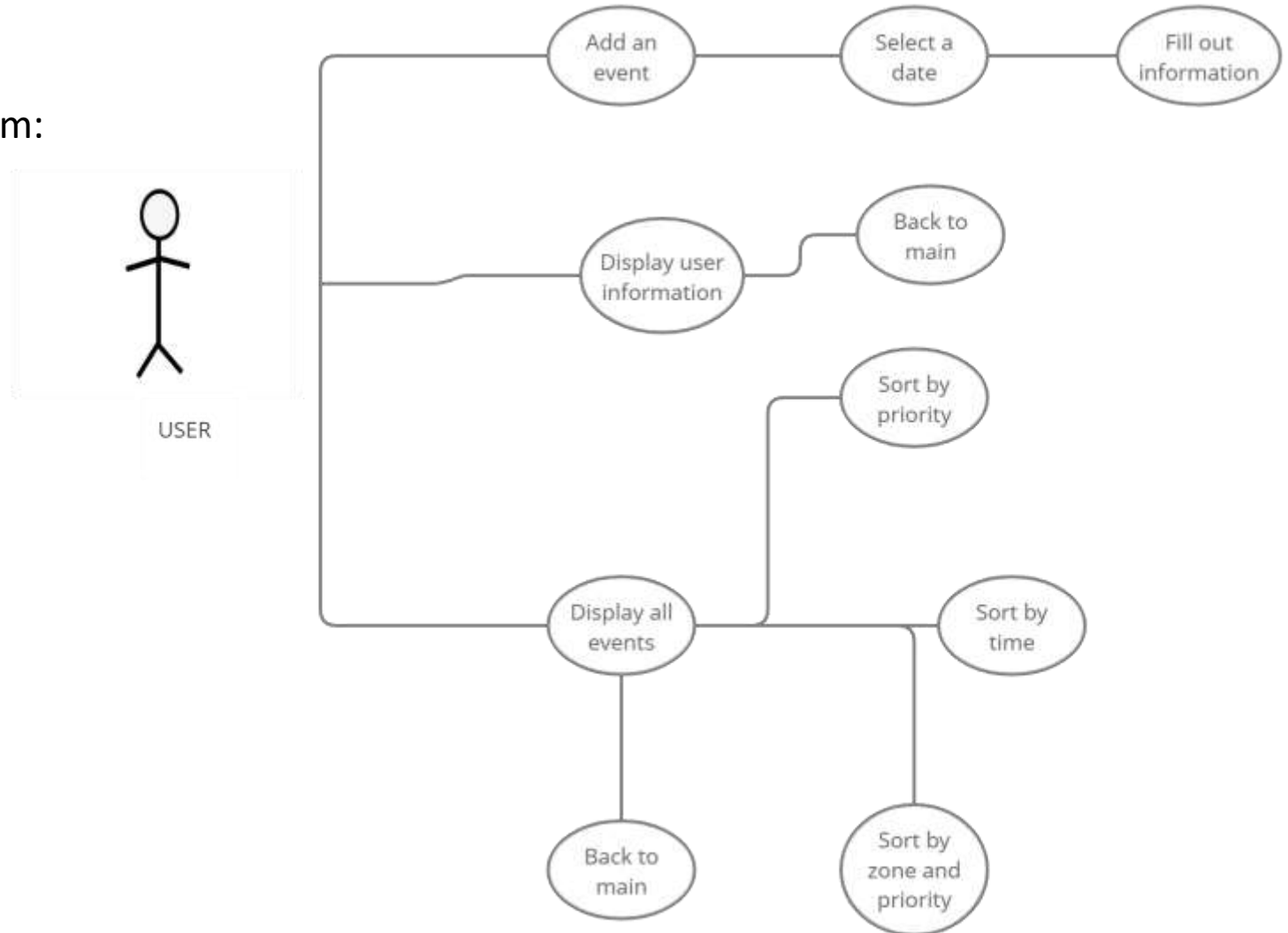
Diagrammok

Activity Diagram:



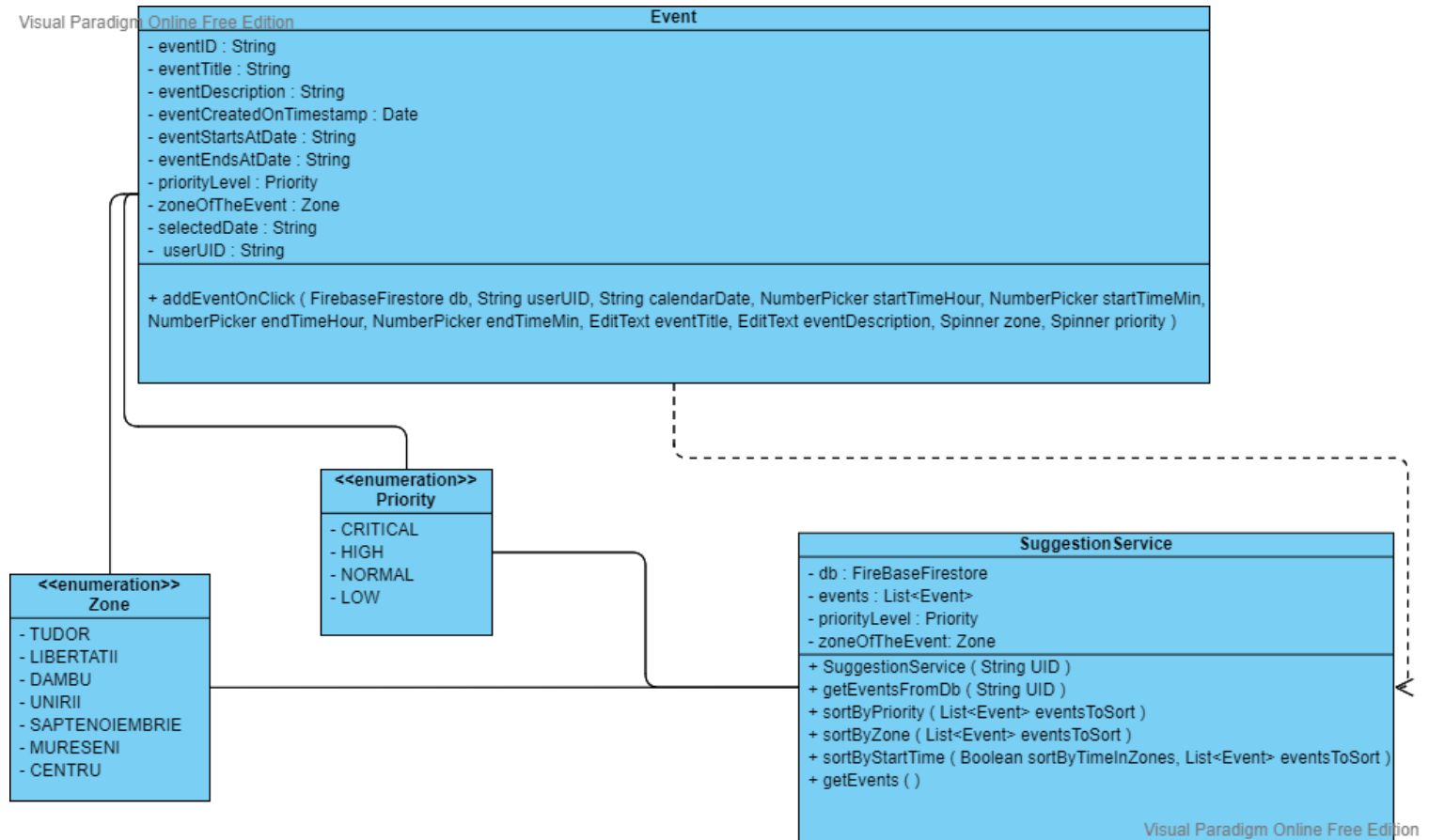
Diagrammok

Use-case diagram:



Diagrammok

Class diagram:



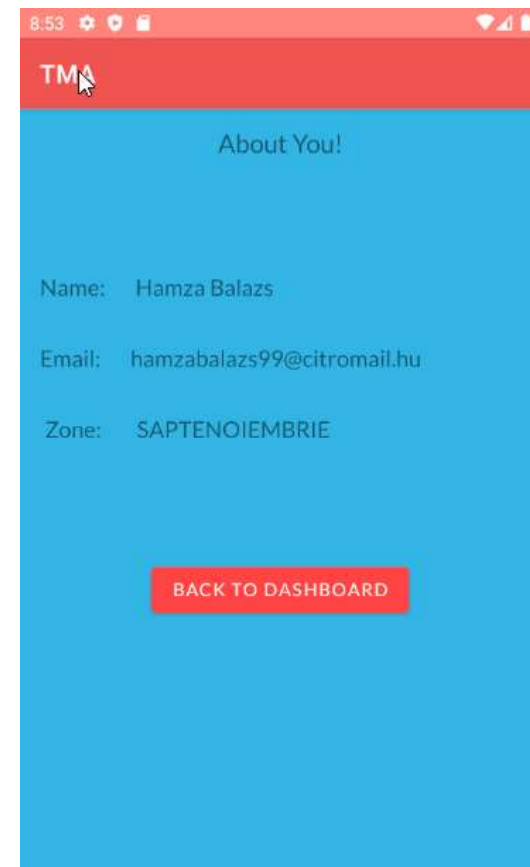
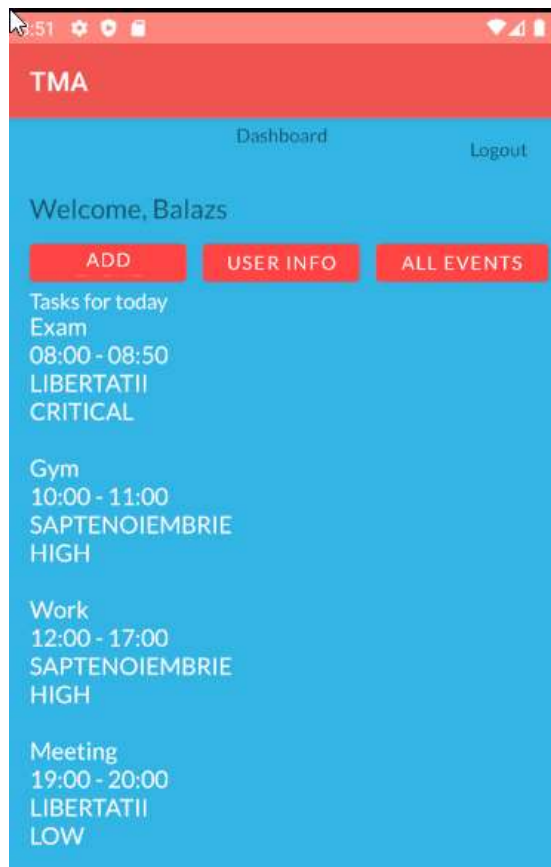
Design

Az applikáció 'design' -ja egyszerű, lényegre térő, az Activityk között gombokkal tudunk lépegetni.





Az applikációhoz használt színek alaposan lettek kiválasztva, hogy minél kecsesetűbbek legyenek az emberi szemnek.

Az egyszerű design lényege, hogy az a személy is tudja majd használni az applikációt, akinek nincs előző ismerete mobileszköz alkalmazásokról és hogy könnyű legyen az adatok olvasása.

Design



Design

- A gombokat piros színnel láttuk el, hogy kitűnjön a háttérből
- A fontos adatok fehér színűek, ezzel is kiemelve fontosságukat
- A többi szöveg fekete színű
- Ezeket a színeket a colors.xml fájlba adjuk meg, amit aztán a themes.xml fájllokba használunk majd fel
- Használt színek:
 - Elsődleges szín : #ef5350 
 - Elsődleges szín (világos) : #ff867c 
 - Másodlagos szín : #29b6f6 
 - Másodlagos szín (világos) : #73e8ff 

Továbbfejlesztési lehetőségek

- Alkalmazás implementálása iOS és webes felületre
- Értesítés kiküldés megadott idővel az esemény előtt
- Rutinos események hozzáadásának automatizálása (pld. edzés, munka)
- Több regisztrációs lehetőség (facebook, appleID, google account)
- GPS implementálása az alkalmazáson belül, ahol pontosan megtudjuk adni az esemény helyszínét.

Összefoglaló

Ez a projekt mindkettőnk számára hasznos volt, fejlesztettük tudásunkat, és még új dolgot is tanultunk a projekt elkészítése során. Sikerült az együttműködés is, nem voltak nézet eltérések.

A projekt kidolgozása során sikerült elsajátítanunk egy kevés NoSQL tudást, továbbá sikerült többet megtudnunk az Android eszközök működéséről, programozásáról.