

BPTracker



Készítette:
Balázs Tamás-Béla
Lukács Szabolcs
Automatizálás és alk. Informatika III.év

Vezető tanár: Szántó Zoltán

Tartalomjegyzék

1.Bevezető	2
2.Projekt célja	3
3. Követelmények	3
3.1 Felhasználói követelmények	3
3.2 Rendszerkövetelmények	3
3.3 Funkcionális követelmények	4
3.4 Nem funkcionális követelmények	4
4.Tervezés	5
5.Tesztelés	10
6. Jövőbeli tervek	11

1.Bevezető

A projekt témája onnan származott, hogy egyik este egy sör pong közepette támadt az ötlet, hogy kéne egy olyan applikáció amivel nyomon tudjuk követni a játékokat és erre nem nagyon találtunk megfelelő applikációt (legfeljebb olyat amit focira használnak). Ezután eldöntöttük, hogy jó ötlet lenne megvalósítani ezt projektnek. Az alkalmazás hasznos lehet akár nagyobb bajnokságok levezetésében is, ahol eddig ez általában papíron történt.

A játékról:

A sörpong (Beerpong) egy közkedvelt ivójáték, mely az 1950-es évekre vezethető vissza, amikor a játékot még ütővel játszották amerikai egyetemisták Beirut néven. A játék célja ugyanis az ellenfél poharaiba juttatni a labdát. A sörpongot két csapat játssza, csapatonként két taggal és 20 félliteres pohárral. A poharak háromszög alakban vannak felállítva egymással szemben, bennük poharanként 0,1-0,3l sör, ízlés szerint.

Az appot amit fejlesztettünk dart nyelvben, illetve Flutter keretrendszerben írtunk. Mivel nekünk ez egy teljesen idegen nyelv volt ezért nehezen haladtunk a dolgokkal. Fejlesztői felületnek választottuk az Android Studiot mert Flutterben dolgoztunk, ami egy úgynevezett cross-platform keretrendszer, ezért az appot könnyen el lehet készíteni androidra, IOS-re, illetve webre, ez is volt az egyik ok amiért ebben a környezetben dolgoztunk.

2. Projekt célja

A projekt célja egy sörpong bajnokság nyomonkövetése, ahol lehet számolni a meccsen dobott labdák számát, a különleges dobásokat, illetve ki mennyit dob be a poharakba, ezeket a rangsorolni a felhasználók bedobott labda pontszáma szerint. Ezeket az adatokat egy adatbázisban tároltuk ahonnan miután a felhasználó regisztrált ahányszor belép esélyt kap felhasználói adatai változtatásra. Mivel a mai világban szinte mindenki rendelkezik okostelefonnal, így akár egy meccs közben is könnyedén követheti a meccs állását, és vezetheti a saját statisztikáját. A felhasználó egyaránt lekérdezheti a jelenlevő bajnokságokat, illetve létre is hozhat bajnokságot.

3. Követelmények

3.1 Felhasználói követelmények

Az alkalmazás futtatására a felhasználónak kell legyen egy androidos telefonja(későbbiekben IOS es Webes felületre is lesz opció). A felhasználó regisztráció nélkül nem használhatja az appot illetve kell internet kapcsolata legyen, hogy csatlakozhasson az adatbázishoz. Az alkalmazás jelenleg csak egy lokális adatbázist használ, mivel nem volt lehetőségünk egy külső szervert használni, ami ki tudná szolgálni az összes felhasználót. A regisztrációhoz szüksége van egy email címre arra az esetre ha elfelejti a jelszavát tud kérni azon egy újat.

3.2 Rendszerkövetelmények

Az alkalmazás legalább a következő rendszerkövetelmények alatt futt el:

- Android 10 (API level 29) és ennél újabb operációs rendszerek
- Mobilinternet kapcsolat vagy Wifi kapcsolat
- Legalább 150 MB szabad tárhely
- 2 GB RAM
- Android SDK

3.3 Funkcionális követelmények

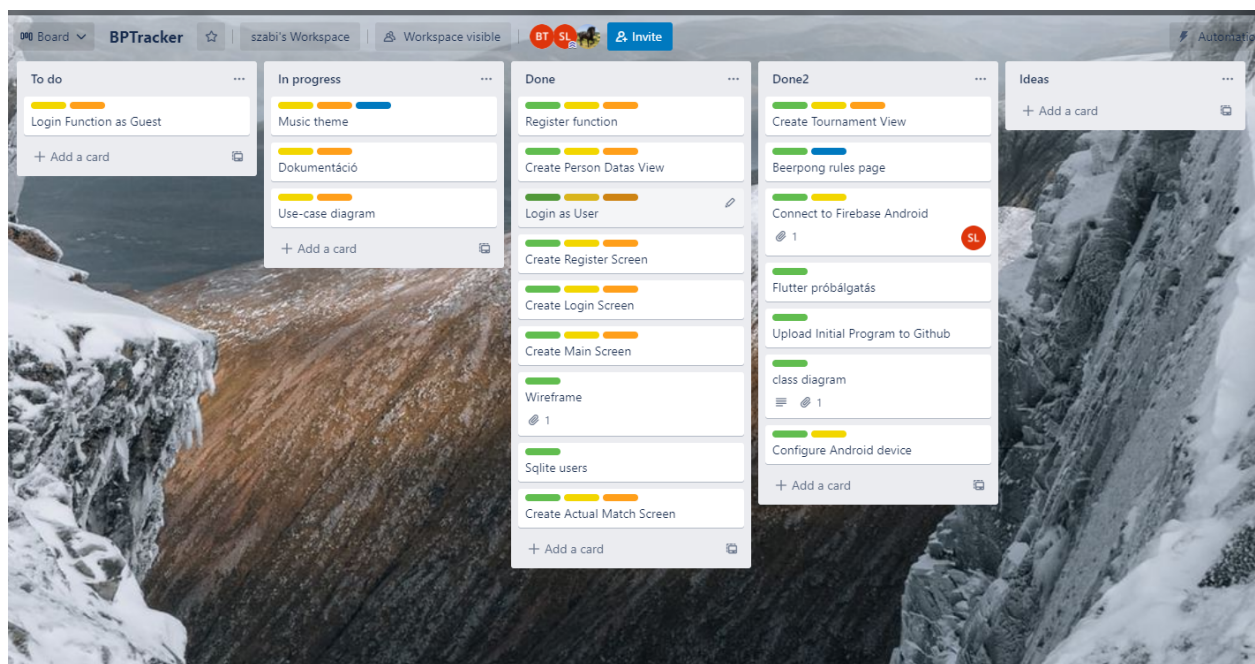
- Kezdőoldal: Ahonnan a felhasználó ha már egyszer regisztrált akkor újra bejelentkezhet, illetve van egy regisztráció gomb ahonnan átkerülnek egy regisztráció oldalra, illetve egy szabályzat gomb ahol eljutnak egy weboldalra ahol elolvashatják a szabályzatait a játék különböző fajtáinak.
- Regisztrációs felület: Ahol a nem regisztrált felhasználók tudnak maguknak profilt készíteni. Az adatok helyes kitöltése után és a 'Sign Up' gomb megnyomása után kezdhetik is a bejelentkezést.
- Saját profil: A bejelentkezés után ide kerülnek a felhasználók ahol tudnak változtatni az adatokon illetve lehetőségük van arra is, hogy törölni tudják a saját profiljukat és még egy gomb által átkerülhetnek a Bajnokság Csarnok menübe.
- Bajnokság Csarnok menü: Ahonnan a felhasználók kiválaszthatják a bajnokságot ahova be akarnak lépni illetve tudnak saját kezűleg létrehozni egyet.
- Bajnokság menü: Ahol a bajnokság játéakai megtalálhatóak és kiírja, hogy melyik csapat játszik melyik ellen.
- Meccs menü: Itt a játékvezető követi a játék pontszámait és növeli annak a játékosnak a pontszámát, illetve a csapat pontszámát, aki bedobott egy labdát a pohárba. Egy játéknak egy 12 perces időkorlát van állítva amit szükség esetén le lehet állítani, szüneteltetni és újraindítani.

3.4 Nem funkcionális követelmények

- Sqlite: 1.3.0
- Android sdk: $\geq 2.12.0 < 3.0.0$
- Url_launcher : 5.4.2
- Cupertino_icons: 1.0.1
- Path_provider: 1.6.5
- Countdown_timer: 4.1.0

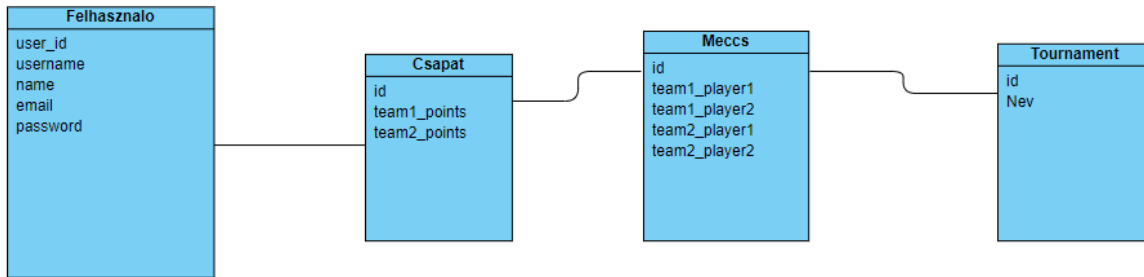
4. Tervezés

Miután megbeszéltük, hogy mi lesz a szoftvernek az alapja, nekikezdünk a Kanban board elkészítésének hogy felosszuk, ki milyen feladatot csinál, illetve elkészítettük a wireframe-t kezdetlegesen, hogy átlássuk a projekt főbb részeit. Ezután inicializáltuk a GitHub Repository-t és elkészítettük az adatbázis diagramot, melyet a későbbiekben még módosítottunk. Az első hetek az Android Studio inicializálásával teltek, egy flutter projekt létrehozásával és az android emulátor elindításával. Ezután kezdtük el tutorial videókat nézni a flutterről, próba projektet csinálni, abban kitapasztalni a keretrendszert, mielőtt nekivágunk a projektnek.



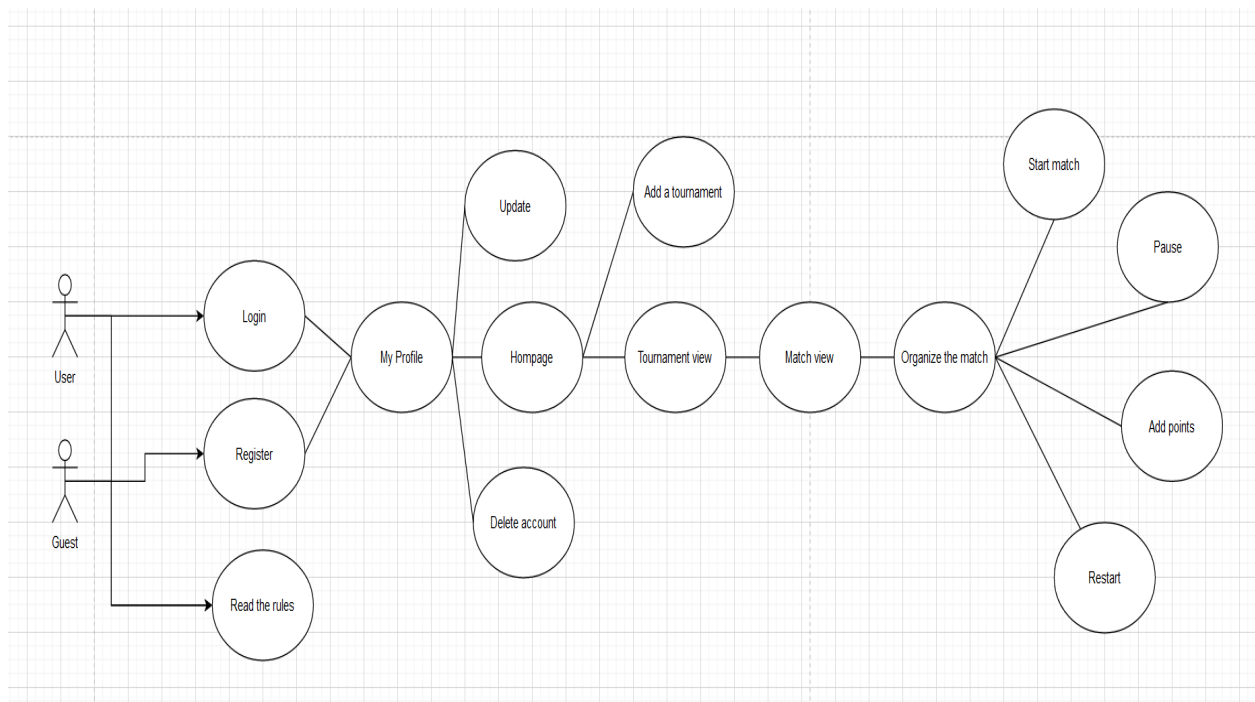
4.1 Kanban board Trello-ban

A Trelloban vezettük, hogy milyen főbb funkciókat, oldalakat kell megvalósítani, ezeknek a haladását, illetve a felosztását.



4.2 Adatbázis E-K diagramja

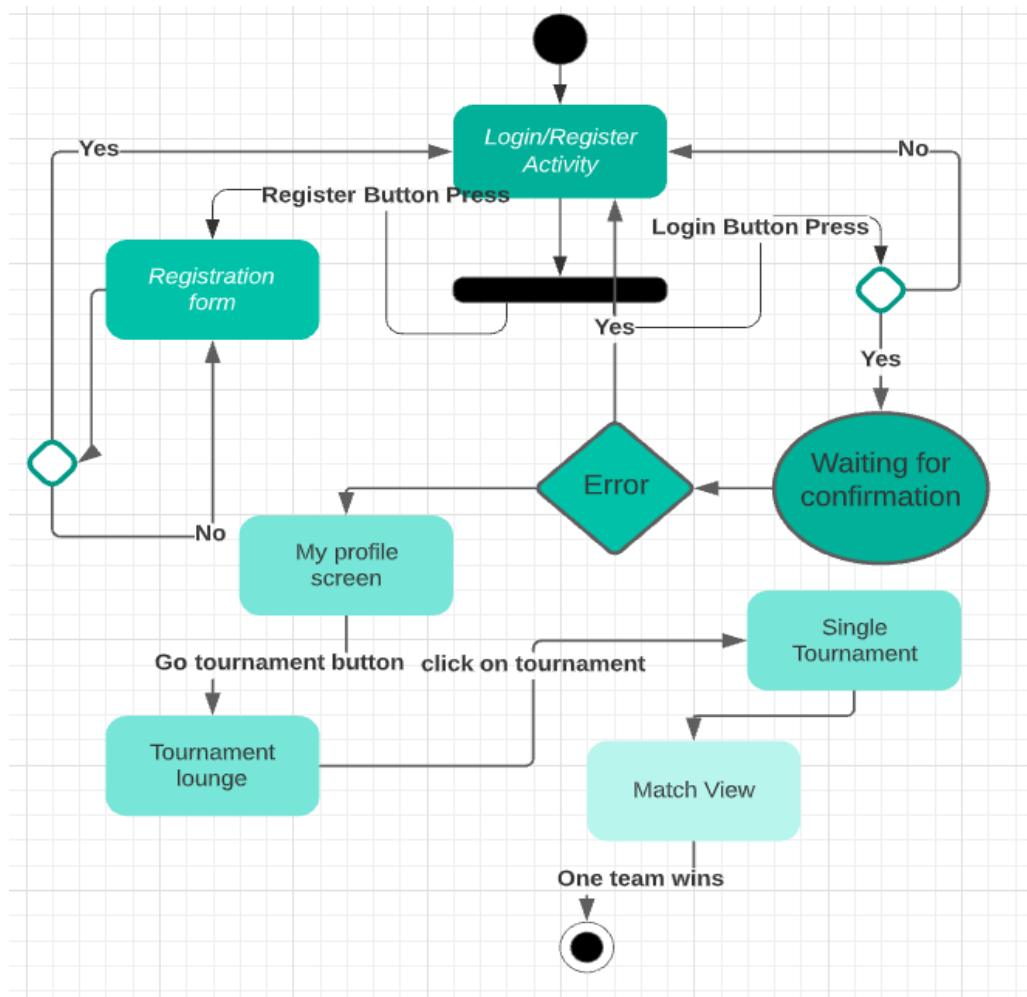
Az adatbázis tervezésénél először a firebase-t kezdtük el használni, de aztán idővel rájöttünk, hogy nem ez a legmegfelelőbb adatbáziskezelő, mivel a meccseket, csapatokat relációs sémákban képzeltük el megvalósítani, a firebase inkább a felhasználókra összpontosít és NoSQL alapú. Ezután tértünk át az SQLite -ra. Az SQLite-ban létre tudtuk hozni a modelleket, és ezek alapján használni az adatbázist.



4.4 Use-Case diagram

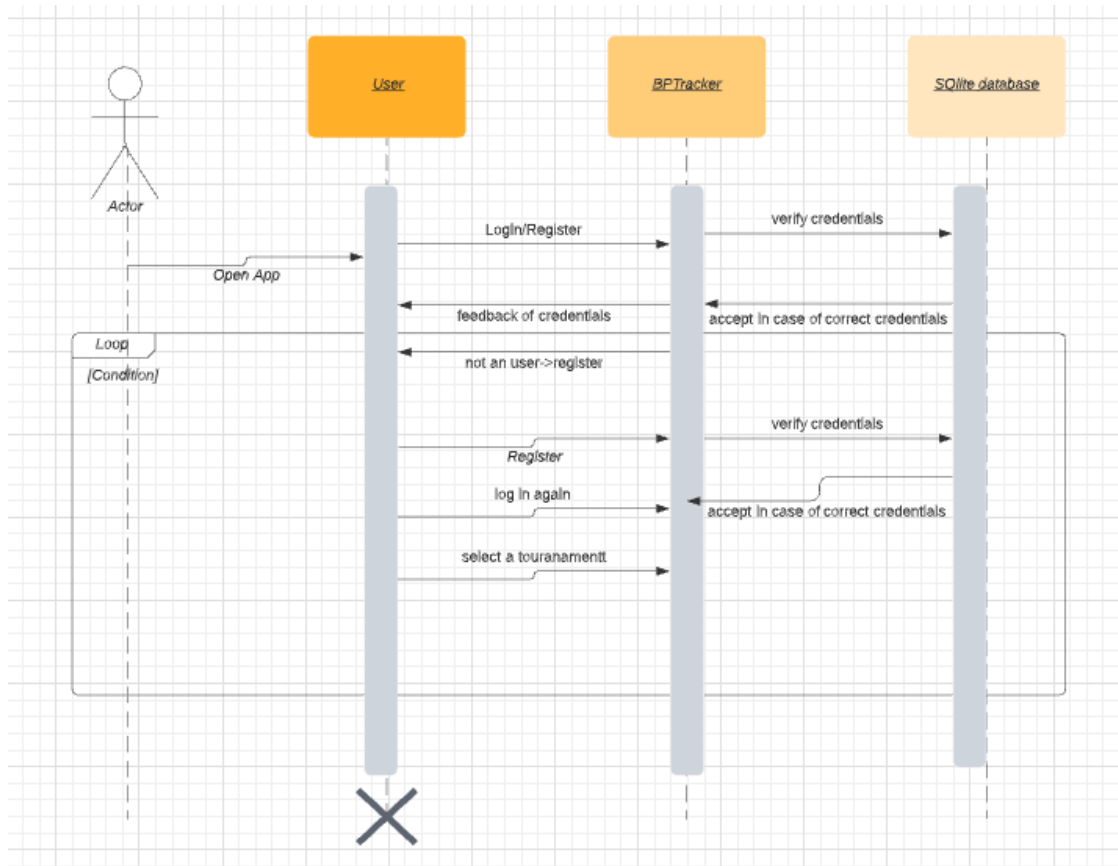
A 4.4-es ábrán az Use-Case diagram látható. Egyszerűen bemutatja egy felhasználó szemszögéből a lehetséges funkciókat. Aki elkezd használni az alkalmazást lehet még nem regisztrált felhasználó, így csak a regisztráció után tudja elérni az alkalmazás funkcióit, aki már regisztrált, az bejelentkezés után, illetve mindketten elolvashatják a játék szabályait. Belépés után a saját adatlapjára kerül, ahol láthatja a felhasználó a saját adatait, módosíthatja azokat.

Ezután a kezdőoldalra léphet, ahol megnézheti a jelenlegi bajnokságokat. Ott egyet kiválasztva kerül a meccsnézetbe, ahol látja a meccs folyását, lehet követni a pontokat, attól függően hogy melyik csapat melyik játékosa szerzett pontot. A meccset lehet szüneteltetni, esetleg újraindítani.

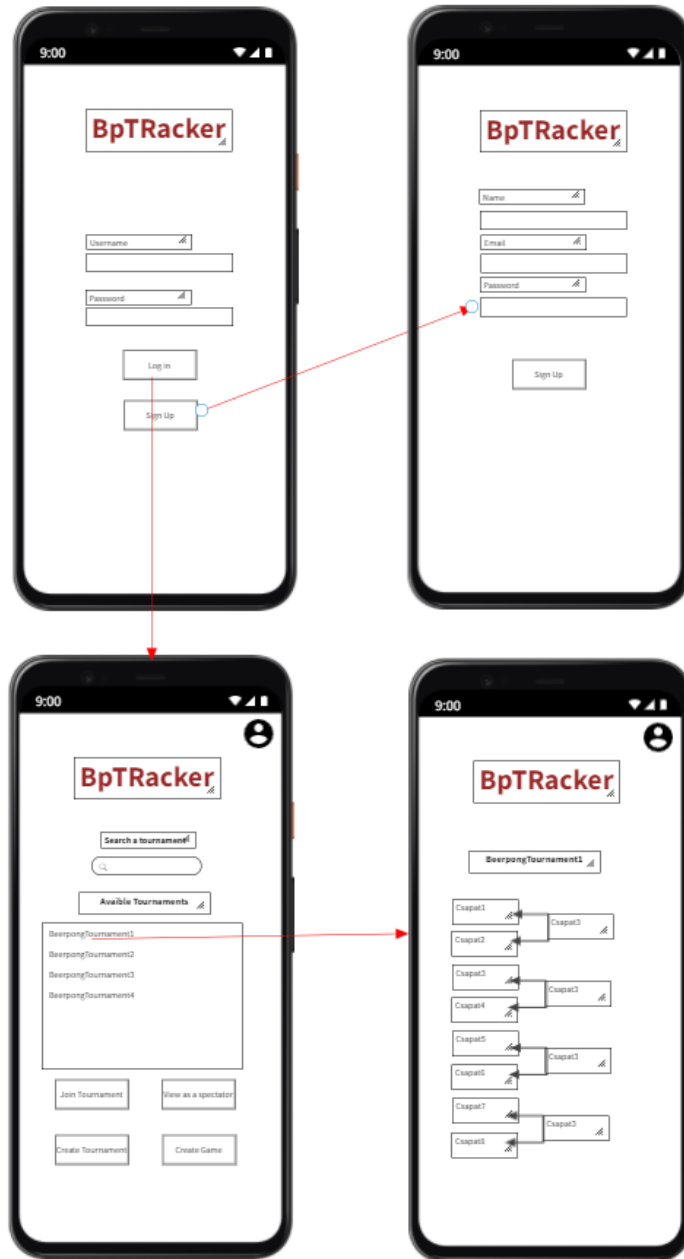


4.5 Activity Diagram

A 4.5 ábrán látható ahogy a felhasználó requesteket küld az aplikációnak, amely az adatbázisnak továbbítja azokat, hogy onnan legyen ellenőrizve például a felhasználói adatok. Helytelen adatok után (például helytelen email cím, már használatban lévő username esetén) egy hibaüzenettel visszatér a login képernyőre. Ha sikerült a bejelentkezés, akkor továbbenged a már felsorolt funkciókhoz.



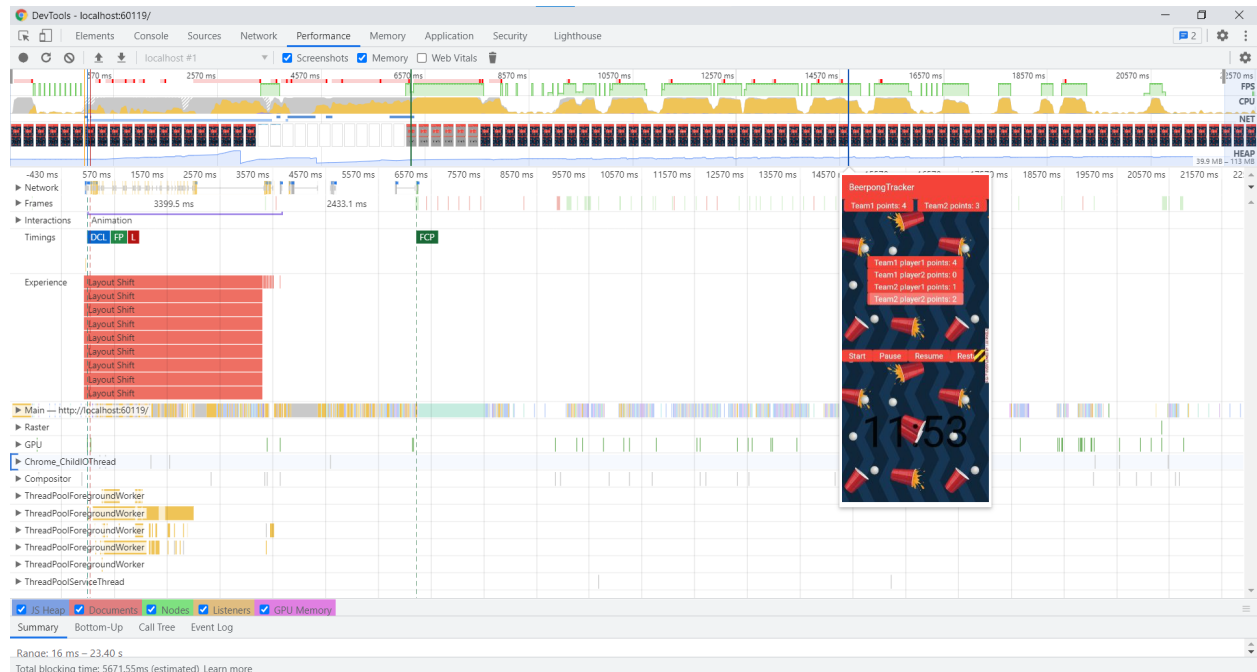
4.6 Szekvencia diagram



4.7 Wireframe

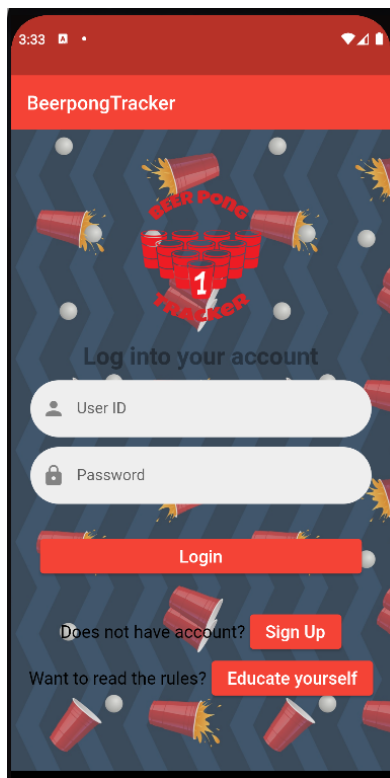
Az alkalmazás kezdeti terve, a főbb képernyők, a főbb funkciókkal, navigációs lehetőségekkel a képernyők között.

5. Tesztelés

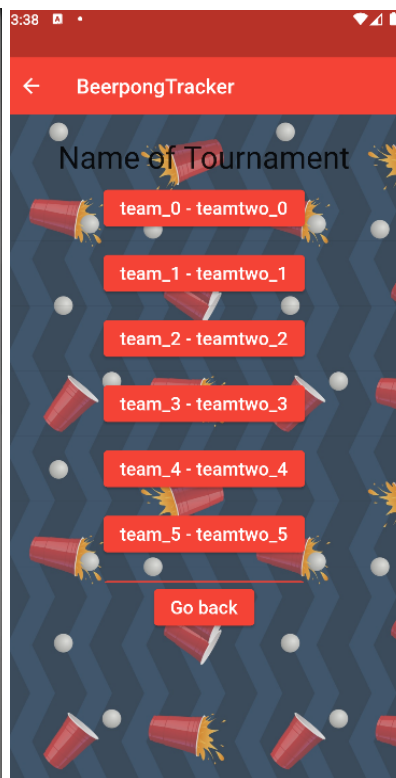


5.1 Alkalmazás teljesítmény webes környezetben

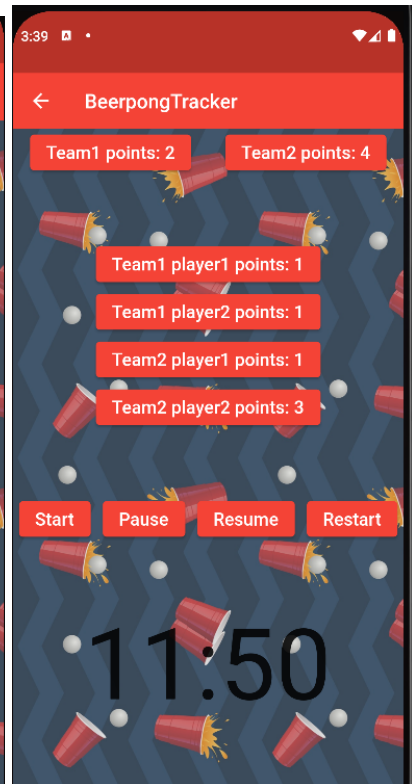
A Google Chrome segítségével letudtuk mérni, nyomon tudtuk követni az alkalmazás viselkedését. A 4.3 - as ábrán az alkalmazás a meccs nézetben való teljesítményét nézte. Az FCP (first contentful paint) kicsivel több, mint 6 másodperc volt, ez az a pillanat amikor a felhasználó először lát valamit a képernyőn. A különböző műveleteknél, például amikor egy játékos pontszámát növeltük, látható hogy megnövekedett a memóriahasználat. A teljes időtartam alatt a memóriahasználat 40 és 113 MByte között volt a 23 másodpercig tartó tesztelés alatt.



4.8



4.9



4.10

Néhány kép az applikáció kinézetéről, illetve különböző képernyőkről mint például “login” (4.8 ábra) “tournament” (4.9 ábra) és a “match” (4.10 ábra) képernyő.

6. Jövőbeli tervek

- Adatbázis átírása firebase-re
- Bővített játékbeli opciók, mint például dupla labda, pattanós labda, háttérzene, stb.
- Animációs felület elkészítése a valós játék nézethez
- Admin mód bevezetése