

High Level Design

Megoldandó probléma	2
Méretezési és tervezési alapelvek	2
Infrastruktúra felépítés	3
Logikai ábra	3
Adatbázis	3
Megvalósítás	4
Funkciók	4
REST API	4
Megvalósítás	4
Funkciók	4
BI	4
Megvalósítás	4
Funkciók	4
Kliensek	5
Megvalósítás	5
Funkciók	5
Alkalmazott technológiák összefoglaló	5
Tanulmányok, amik alapján kiválasztottam a programnyelveket	5

Megoldandó probléma

A program egy keresztplatformos mobilalkalmazást fog megvalósítani. Alapvetően egy közösségi média-szerű applikáció lenne, amelynek elsődleges célja újdonsült üzleti kapcsolatok teremtése. Az applikáció megvalósítaná a közösségi média applikációk általános funkcióit, azaz a regisztrációt, a bejelentkezést, kijelentkezést, posztok készítését és olvasását, kontaktok felvételét és az azokkal történő kommunikációt. (Utóbbit leginkább írásos üzenetek formájában). Emellett egy algoritmus biztosítaná, hogy a megfelelő személyekkel tudja a felhasználó felvenni a kapcsolatot.

Méretezési és tervezési alapelvek

Az alkalmazásnak maximum 1000 egyidejű felhasználót kell tudnia kiszolgálni.

Tegyük fel, hogy a felhasználók 80%-a aktív. Az aktív felhasználók 20%-a posztol egy nap. Egy felhasználó várhatóan napi 2 posztot készít, posztonként átlagosan 5 MB adatot generál. Ez 1000 - aktív és passzív - felhasználóra és 1 évre vonatkoztatva körülbelül 584 GB.

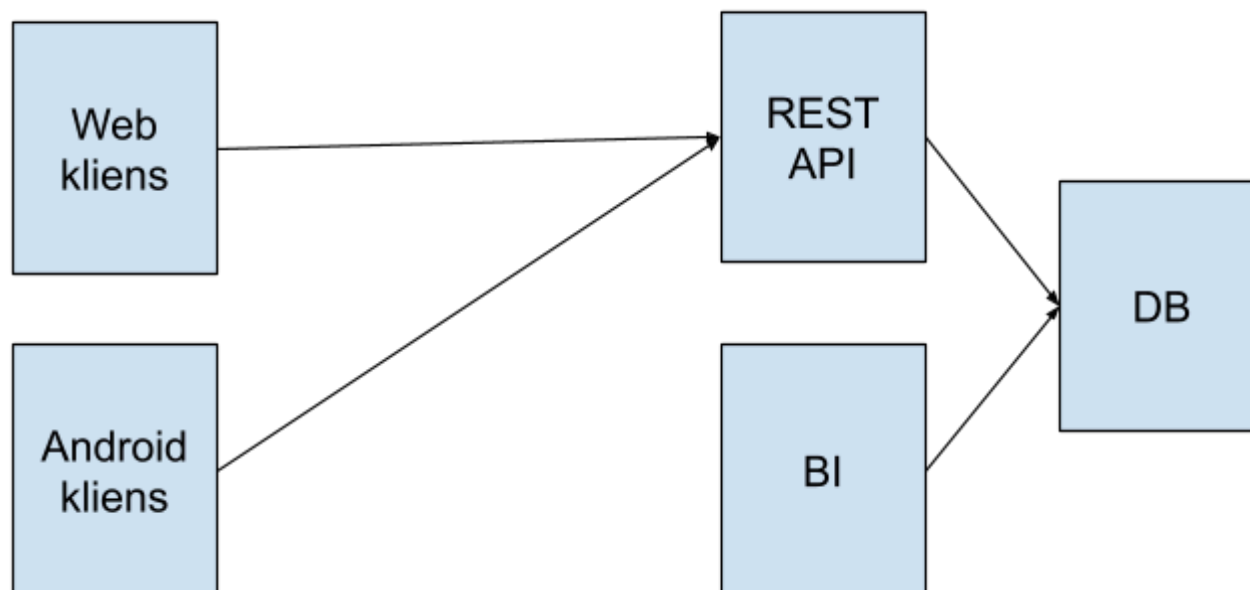
A felhasználó modern böngészőt használ. (Vagy Android telefonja van, minimum Android 9-es operációs rendszerrel.)

Infrastruktúra felépítés

Logikai ábra

Frontend

Backend



A különböző komponensek közötti kommunikációnak titkosítottnak kell lennie.

Adatbázis

A kötöttségek elkerülése céljából logikát nem tartalmaz.

Megvalósítás

SQL vagy NonSQL.

A választás SQL.

Funkciók

- Felhasználói profilok tárolása (uuid4 szabvány használata az egyedi azonosítókhoz, email mint felhasználói azonosító)
- Egyéb információk (pl. posztok) tárolása

REST API

Esetleg RPC is használható lehetne helyette, de az API a választás.

Megvalósítás

A backend PHP nyelven lesz megvalósítva.

Funkciók

- A kliens kéréseinek kiszolgálása (CRUD)
- Felhasználói autentikáció és autorizáció. (JWT, Bearer Token)
- Email küldés külső email szolgáltató API-ját használva

BI

Üzleti intelligencia. Az input paraméterek megadásához komplex logikai megfogalmazás szükséges, ennek formáját meg kell határozni. Célszerű nem egyedül használni, hanem a jól bevált megoldásokhoz fordulni. Ezzel kapcsolatban szükség lehet kutatásra.

Megvalósítás

A backend Python nyelven lesz megvalósítva.

- [A Python platformfüggetlen.](#)
- [Linux disztribúciók tartalmazzák.](#)
- [Szkriptnyelv.](#)

Funkciók

- A megadott paraméterek alapján létrehozni a “match” táblát

Kliensek

Felhasználói interface

Megvalósítás

A választott programnyelv az Angular.

Funkciók

- Regisztráció
- Bejelentkezés
- Saját profil karbantartása
- BI funkcionális eredményének publikálása felhasználó felé
- Egyéb

Alkalmazott technológiák összefoglaló

Frontend: Angular

Backend: PHP

Adatbázis: MySQL

Tanulmányok, amik alapján kiválasztottam a programnyelveket

<https://www.integrate.io/blog/the-sql-vs-nosql-difference/>

<https://www.ibm.com/blog/sql-vs-nosql/>

<https://www.geeksforgeeks.org/difference-between-sql-and-nosql/>

<https://www.simform.com/blog/angular-vs-react/>

<https://radixweb.com/blog/react-vs-angular>

A programnyelvek kiválasztásánál figyelembe vettem azt is, hogy ez a projekt az én tudásomat hivatott tükrözni. El akartam mélyülni a fejlesztés során több nyelvben is, hogy minél több programnyelvvvel kapcsolatos tudásomat mutathassam be.