

Szoftverfejlesztés

Egyetemi projekt

Vezető tanár: Szántó Zoltán

Diákok: Toducz Endre

Kezdés ideje: 2022.05.00
Befejezés dátuma: 2022.06.08

Tartalomjegyzék

1. Bevezető
2. Célok
3. Software követelmények
 - Felhasználói követelmények
 - Rendszerkövetelmények
 1. Funckionális
 2. Nem funckionális
4. Tervezés
 - Architektúra
 - Alkalmazás áttekintése
5. Mikroszervizek leírása
6. Kivitelezés
7. Összegzés

1. Bevezető:

Az ötlet onnan származik, hogy elkezdtem a magánéletemben jegyzeteket írni a mindennapjaimról. Hogy miért excelbe, hát ez egy jó kérdés. Kiolvastam Dr. Varga Péter

Spielhózní című könyvét, amely azt fejtegeti, hogy érzelmeim, és hatékonyságom, hogyan fejezzem ki számokkal. Majd azt fejtegeti, hogy ezeket, hogyan lehet és érdemes egy görbére feszíteni, hogy könnyedén monitorizálható legyen az életemben lévő érzelmek bizonyos területekhez kötve.

Egy kis idő után rájöttem, hogy ezeket a jegyzeteket, értékeket amelyet magamhoz fűzök szeretném mással is megosztani, vagy akár egymás irányába más személlyel együtt, költsönösen monitorizáljuk egymás érzelmeit. Erre is megfelelő volt az excel táblázat.

Mikor minden nap ezt megtettem egy év után arra vágytam, hogy írjak egy saját kis alkalmazást erre a feladatra.

A másik célom pedig az, hogy mások számára is meséljek és javasoljam ezt a módszert és ennek keretében tujak hozzá nyújtani egy segéd eszközt.

2. Célok:

Azt tűztem ki célul, hogy kezdetben megvalósítom az alkalmazás azon részét, ahol költsönösen visszajelzünk egymásnak. Napról napra jól látható módon, követhető formában.

3. Szoftver követelmények

a. Felhasználói követelmények:

A felhasználók szempontjából igyekeztem minél felhasználóbarátabb alkalmazást létrehozni, ami magába foglalja, hogy könnyedén kezelhető legyen még a kezdő felhasználók számára is. Emellett átlátható és vonzó designja legyen.

b. B.Rendszerkövetelmények

Funckionális: Az alkalmazas futásához mindenképp szükség van egy böngészőre és internet kapcsolatra.

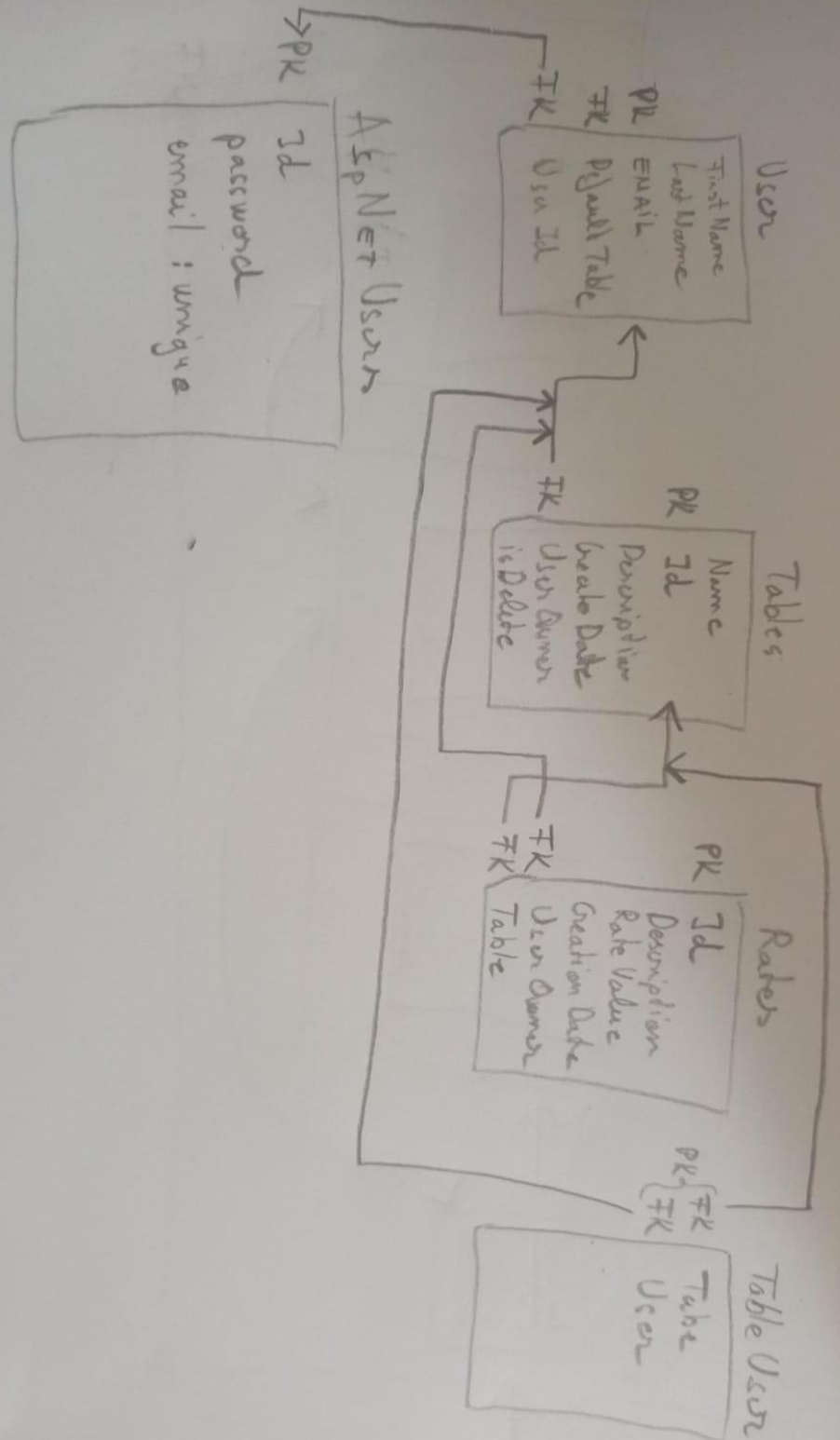
Nem funckionális: Az alkalmazás futtatásához szükség egy mindennapi személyi számítógéphez.

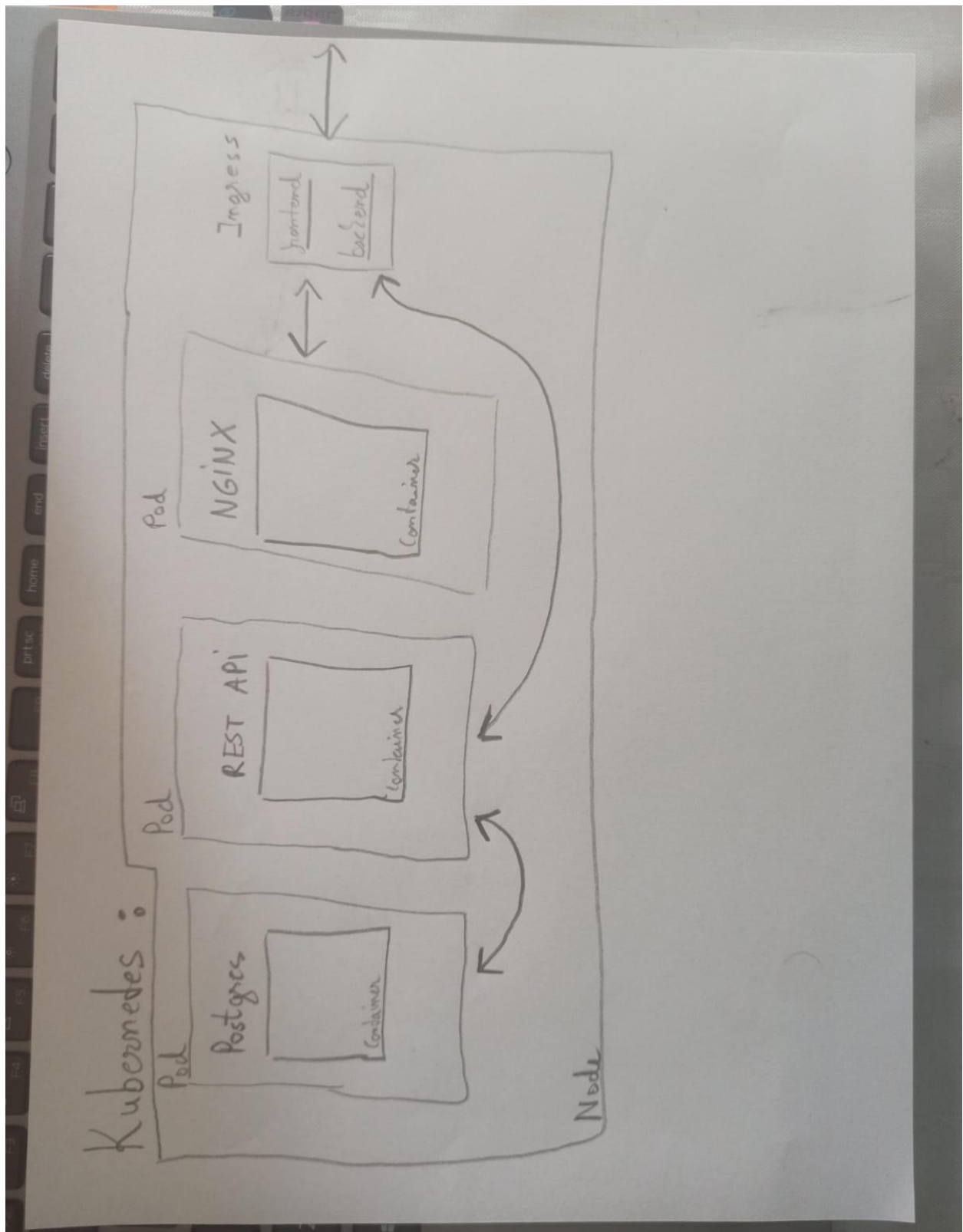
4. Tervezés

Az alkalmazás microservice architektúrára épül. Három részből tevődik össze. Az első egy postgres adatbázis, a második egy Rest Api, amely ellátja a backend feladatokat és az utolsó pedig egy webfelület, ahol a felhasználói környezetet valósítjuk meg.

Az alkalmazás kitelepítése Kubernetes környezetbe történt.

A felhasználói felület megírása angular keretrendszerben történt, mivel napjainkban az egyik leghasználtabb és legideálisabb környezet erre a célra. A backend részen dotnet 6.0-ra esett a választás, mivel volt már tapasztalatom benne és hatékonyan lehet benne felépíteni kisebb Rest Api alkalmazást.





2.ábra

5.Mikroszervizek leírása:

1.Postgres

A docker hubról töltöttem le a latest verziót, majd ezt telepítettem ki Kubernetesbe helm chart segítségével.

Az adatbázis megtervezése az első ábrán látható.

2.Rest Api:

A rest api tartalmaz egy központi kontrollert amely, kiszolgál minden kérést. Json formátumban küldözgetve az adatokat. A dotnet 6.0 a postgres adatbázissal kommunikál. Entity frameworkot használva az alkalmazás felépítéséhez és kezeléséhez.

3. Angular

Az angular keretrendszerben fejlesztettem le a felhasználói felületet. Generáltam belőle egy statikus fájlot, majd egy nginx szerverrel futtatva kiszolgálom a kéréseket.

A kubernetes segítségével helm chartokat használtam. A kubernetes architektúrát az 1.ábra szemlélteti.

6.Kivitelezés:

Verziókövetésre a githubot használtam, ahol létrehoztam két különálló projektet, egyet a frontend, még egyet a backend alkalmazásnak.

7.Összegzés:

A projekt hatalmas kaland volt számomra és egyben egy hobbi is, sok időt öltem bele, de úgy vélem, hogy sikerült egy olyan eszközt elkészíteni, amely az emberek javát szolgálja.

