

## Rövid összefoglaló a 2019-es nyári szakmai gyakorlatról

A szakmai gyakorlat keretein belül az SAP HANA keretrendszerét használtuk alkalmazásfejlesztéshez.

Az első két héten ennek birtokbavétele volt a feladatunk. Megismerkedtünk az olyan alapvető funkciókkal, mint például az alkalmazás létrehozása, modulok hozzáadása, kalkulációs nézet készítése, az alkalmazáshoz adatbázis hozzáadása és annak kezelése és az adatbázisba adatok importálása, melyeket nagyrészt online tutorialokon keresztül tettünk, de szakmai gyakorlatot vezető tanárok is nagymértékben elősegítették fejlődésünket.

A gyakorlat során két fontosabb feladattal kellett kisebb csoportokban megbirkóznunk.

Az első fejlesztési feladatunk egy olyan alkalmazás volt, amelyben a CSV formátumban megkapott társasági adatokat importálni kellett a HANA rendszerbe, az importált táblát normalizálni kellett, az adatokat némiképp meg kellett tisztítani. Az adatbázishoz kalkulációs nézetet kellett készíteni. További lehetőség volt a kapott adatokat egy térképes felületen megjeleníteni, azaz geokordinátákat rendelni az egyes címekhez melyek a cégekhez tartoztak, azon kívül ezt valamilyen felületen előhozni. Az én feladatom itt a Bing map API segítségével való adatkinyerés volt a címekből, egy java modulon keresztül, mely az API kismértékű támogatottsága miatt sajnos nem egy használható alternatíva más hasonló modulokkal szemben, így később elkerülendő, Ugyanis a várakozási idő a szerverre elfogadhatatlan, azon kívül jelenlegi tudásom szerint nem lehet specifikusan csak koordinátákat lekérdezni a szerverről, mivel az a beírt címre válaszul egy teljes file-t ad vissza, melyből az adatkinyerés értékes számolási kapacitást vesz el. Ez a probléma később is látható volt, ellenben a kliensoldali adatkinyerés ideje eltörpült a szerver válaszüzeje mellett.

Második feladatként egy streamen érkező szenzoradattömeget kellett feldolgoznunk, tárolnunk és megjelenítenünk. Mi már csak a raw Data-val "találkoztunk", mely sajnos a későbbi adattárolásra nem felhasználható volt, emiatt saját parser programot kellett írunk, melyet megegyezés alapján legtöbbször megtettünk, lehetőleg más platformokon, hogy végül legyen egy minél gyorsabb adatfeldolgozású parserünk. Én ezt a feladatot Java környezetben oldottam meg, de úgy gondolom a felhasznált algoritmusok sokkal nagyobb szerepet jelentettek ilyen szinten, mint a programozási nyelv/környezet. Büszkén jelenthetem ki, hogy a saját parserem nagyságrendekkel gyorsabban dolgozott mint más megoldások, ellenben azon nagy hátránnyal, miszerint ha a kapott adat formátuma később megváltozna, a parser átírása jóval több időt venne igénybe mint más parserek esetében, mivel túlságosan is erre a feladatra specifikált.

Ezután a kész/feldolgozott adatot kellett feltöltenünk az OData állományba, ez egy kiváló lehetőség volt arra hogy különböző hibákat javítsunk a parser-en, mivel a

WEB IDE rögtön jelzett bármiféle hibát a táblában. Az adatok innen kerültek át egy Angular felületre, mely a megjelenítést végezte.