Miskolci Egyetem

Gépészmérnöki és Informatikai Kar

Általános Informatikai Intézeti Tanszék

****

**Csomaglogisztikát támogató**

**rendszer fejlesztése**

**Diplomaterv**

**Készítette:**

**Név:** Éliás Dániel

**Neptunkód:** W8YSQV

**Szak:** Mérnök Informatikus BSc

Korszerű web technológiák szakirány

Tartalomjegyzék

[1. Csomag Logisztika 3](#_Toc103221530)

[2, Feladat ismertetése: 5](#_Toc103221531)

[3.Felhasznált keretrendszerek 7](#_Toc103221532)

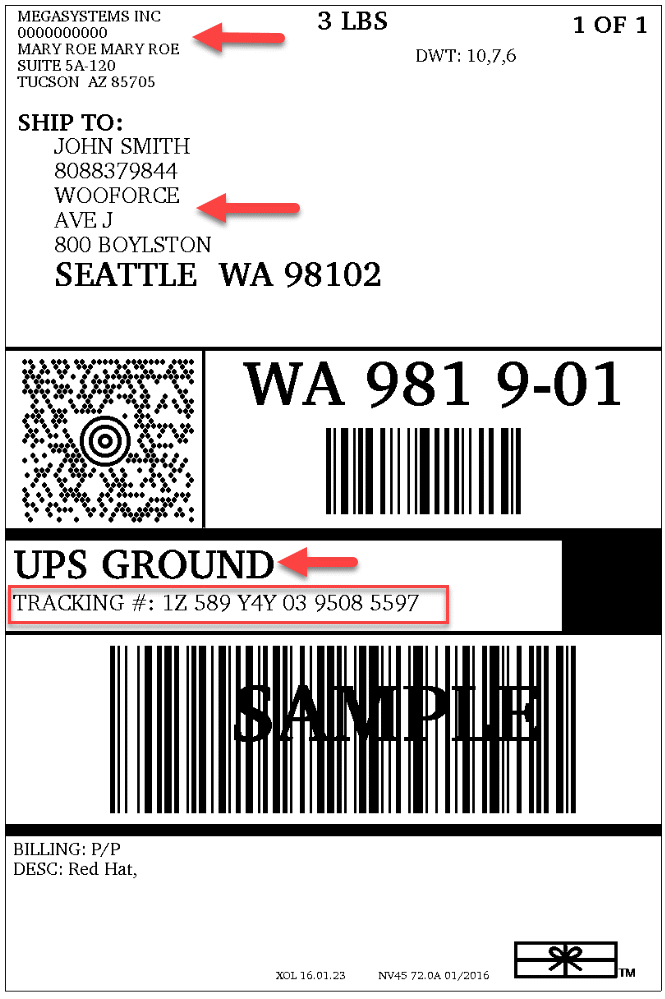
[Java csomagok: 9](#_Toc103221533)

[Források: 11](#_Toc103221534)

# Csomag Logisztika

A logisztika nagyon egyszerűen megfogalmazva egy dolog eljutása a küldő féltől a fogadó félig, vagyis a csomag eljut A-ból B-be. Ez azért fontos, mert az emberek általában lusták és egyszerűbb megrendelni valamit egy boltból például minthogy elkészülni és elmenni egy boltba meg venni magadnak. De nem feltétlenül lustaság az oka annak, hogy az emberek rendelgetnek az interneten ez csak a kis része a nagy képnek. A nagy kép viszont elég tág tud lenni a csomag logisztikát mert, akár, ha kinézünk az útra akkor biztosan lehet látni kamionokat, amik viszik a napi élelmet a boltokba vagy gyárakba visznek különböző termékeket vagy alkatrészeket. A logisztika nagyon fontos szerepet játszik az életünkbe és ha leállna akkor nagyon sok minden problémás lenne a világba.

A csomag logisztika több részből áll össze lehet belföldi logisztika ez azt jelenti, hogy egy országban áramlik a csomag csak. Ez például azt jelenti, ha én Miskolcról feladok egy csomagot Budapestre egy országon belüli szállítóval. A következő rész az maga az import ez ugye azt jelenti, hogy a csomagot külföldről feladják és te saját országodban felveszed a csomagot ugye ekkor a csomag bejön egy másik országból. Például amikor te Kínából berendelsz egy csomagot és hazai szállító hozza be az árut például import részhez tartozik. A következő rész az export itt ugye te küldesz egy csomagot külföldre és hazai szállító viszi ki a csomagot. A negyedik és ezzel az utolsó része az egész csomag logisztikának a harmadik ország által intézett csomag küldés itt ez azt jelenti, hogy te például Magyarországról akarsz küldeni egy csomagot Németországba mondjuk egy Ausztriai szállítóval.

A csomagokat ilyenkor először egy felületen létre kell hozni, amihez a szállító maga le generál egy címkét, ami alapján létrejön egy csomag azonosító és megtalálható rajta az olyan adatok, ami a szállítás során elengedhetetlenek, mint például a küldő neve, címe ez országgal irányító számmal várossal utcával és házszámot jelent akkor ide tartozik még a küldő telefon száma is és lehetnek ezen külön opciók még, mint például a küldő email címe a küldő üzente a futárnak. A másik fontos dolog még a csomagoknak a súlya és a méretek, vagyis a szélesség, hosszúság és a magasság. Emellett természetesen rajta van a címkén a fogadó adatai is ugyan úgy, mint a küldő félnek. Ezek mellett természetesen rajta vannak a futár szolgálatnak egy-egy specifikus szállítási szolgáltatása vagy akár másféle szolgáltatás is ilyen például az elektronikus vámkezelés, ahol, ami azt jelenti, hogy minden gépen intéznek nem nyomtatnak külön papírt mindennek mert egy idő után mondjuk egy sok csomagos küldeménynél sok papír fogyhat el feleslegesen mert meg lehet oldani elektronikusan is. De van még olyan megoldás is, ami megjelenhet egy címkén, hogy Carbon semleges küldemény ez azt jelenti, hogy te plusz díjat fizetsz és ezzel támogatod a szállított azzal, hogy csökkentse a CO2 kibocsátást azzal, hogy vagy gázzal hajtott autóval menjenek ki és vigyék el a csomagot, de ugye ebből a pénzből segíti a különböző szervezeteket, akik a CO2 kibocsátás csökkentésére törekednek. De ezek mellett megtalálható még a számítóknak fontos vonal kódok, amik a rendszereikbe való beolvasást segíti meg.

2. ábra GLS retúr címke

Ezek után a felhasználónak még egy csomag felvételt is el kell indítania mert anélkül nem mennek ki a szállítók a csomagért hogyha meg történt a csomag felvétel foglalás akkor a lefoglalt idő intervallumon belül megérkezik a csomag szállító cég aki fel veszi a csomagodat és utána viszi tovább a szállító raktárába és onnan meg a fogadó félhez a csomag és ez a szállítási folyamat eltarthat nagyon sokáig mert ha külföldről jön be akkor a vám megfoghatja pár napig vagy különböző szolgáltatásokat biztosít egy-egy szállító mint például az aznapi csomag felvétel vagy a szombati csomag felvétel és ezek is gyorsíthatják a csomagok szállítását. Természetesen ezek mind plusz szolgáltatások, vagyis ezek plusz költséggel járnak.

1. ábra UPS Címke

Minden szállítónak meg vannak a saját szolgáltatásai és ezek egyre bővülnek az idők során, mint például most már egyre népszerűbb csomagpontos szállítás ez ugye azt jelenti, hogy a szállító nem az fogadó félhez viszi a csomagot, hanem az ő általa kiválasztott csomagpontra és nem kell attól félnie a fogadó félnek, hogy olyan időpontba hozzák ki a csomagját amikor ő nincs otthon ezzel csomó idegességet meg lehet spórolni. Ezeknek a szolgáltatásoknak meg van egy saját díjazási módja, amit minden futár cég biztosít. Általában ezek a díjazások úgy működnek, hogy az országon belüli szállításnál egy az ár, de mondjuk, ha az Európai szállítást nézzük akkor ott már Európát zónákra osztották és úgy van minden ország egy zónába besorolva és úgy van beárazva.



3. ábra GLS HU ártáblája az Európai II.zónához

# 2, Feladat ismertetése:

Az én feladatom egy olyan applikáció létrehozása, amely egy felhasználó létre tudja hozni a csomagját és kezelni is tudja ezeket a csomagokat és ehhez az oldalhoz létrehozni egy bejelentkező rendszert, hogy lehessen a csomagokat emberekhez kötni és a felhasználók kezelése érdekében egy új szerep bevezetése, ami az admin szerepkör, hogy egy felhasználót tudjunk módosítani vagy létrehozni.

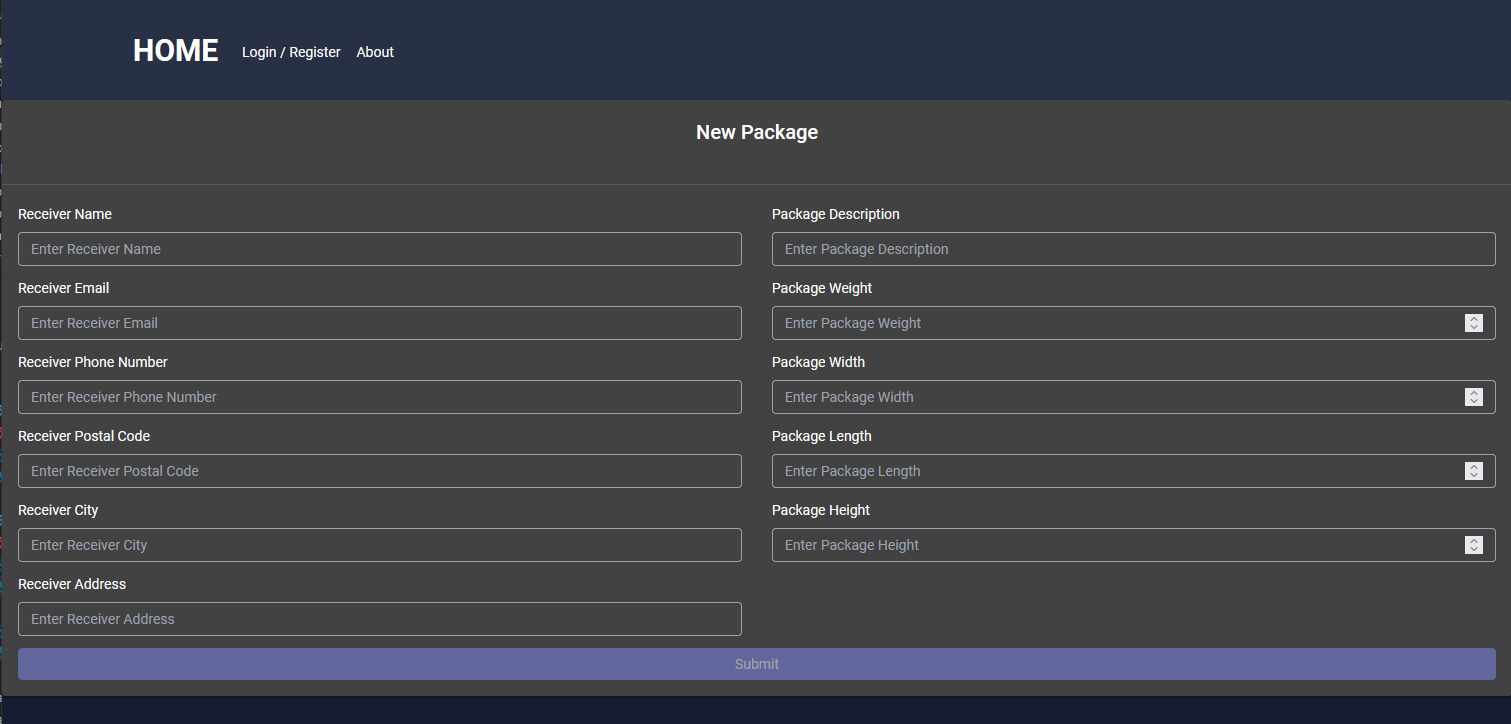
Az applikációhoz az ötlet onnan jött, hogy jelenpillanatban egy ilyen cégnél dolgozom, aki csomag logisztikával foglalkozik és itt kiderült nekem, hogy mennyire fontos maga a logisztika és ha jobban bele gondol az ember, ha a szállítás megállna akkor nagyon nagy gondok lennének. Szerintem mondhatnám az is, hogy megállna az élet. Egy ilyen applikációban nagyon nagy potenciált látok e miatt mert nagyon sok ilyen oldal van ami csomag feladással foglalkozik és mindegyiknek vannak előnyei és hátrányai ezért kell megtenni mindent hogy egy olyan applikációt kell létrehozni ami felhasználó barát vagyis egyszerű és csak is a lényeges dolgok jelenjenek meg a felhasználó előtt és természetesen követni kell a felhasználói igényeket, mert egy fejlesztés szerintem akkor lehet jó és sikeres ha vannak vissza jelzések hogy egy bizonyos fejlesztés megérte-e vagy sem.

Gondolkoztam azon, hogy egy webapplikációt készítsek vagy asztali alkalmazást, mind a kettőnek meg van az előnye és a hátránya, de én jobban preferálom a webes alkalmazásokat. Az ok az pedig az, hogy szerintem ebben vannak lehetőségek és ez fog folyamatosan fejlődni, mert egyszerűbb ezt arra értem míg egy asztali alkalmazást fel kell telepíteni míg a webes alkalmazásokat nem kell csak egy webes címre és rögtön láttod is az applikáció tény és való, hogy van pár dolog, amit még nem lehet megoldani webes alkalmazásokkal, de biztos vagyok benne, hogy már valaki dolgozik egy megoldáson. Egy ilyen dolog, amiről például én is tudok, hogy nem tud a webes alkalmazás közvetlen egy nyomtatóra csatlakozni minden platformon, de biztos vagyok benne, hogy valaki már dolgozik rajta vagy már lehet kész is van. A másik nagy előnye a webes alkalmazásnak, hogy platform független mert csak egy böngésző kell hozzá, de nagyon ügyelni kell arra, hogy csak olyan dolgokat használjunk fejlesztés során, ami minden platform böngészőjével kompatibilis mert néhány új fejlesztési opciók nem érhetők el.

Az alkalmazásom lényege csomagfeladásból áll egy felhasználó bejelentkezik és elkészíti a csomagját majd arra elkészül egy címke és azt viszi majd tovább a futárszolgálat. Ezt az alkalmazást bárki használhatja bármely korosztály mert

A funkciók, amiket meg kellett oldanom az regisztrálás, bejelentkezés, felhasználó létrehozása, módosítása, törlés a csomagok létrehozása és törlése a törlés az úgy kell működnie, hogy ne tényleges törlés legyen mert ha egy csomag eltűnik, vagyis ő ezt állítja és közben kiderül, hogy csak kitörölte a csomagját. A csomag módosítása azért nem ajánlott mert ha egy csomagot tudnánk módosítani mondjuk miután elkészült hozzá egy címke akkor gond lehet, ha emiatt rossz helyre megy a futár. Ezért is mentem le egy csomag létrehozásánál mind a küldő mind a fogadó felhasználó adatait, amit megjelenítek a címkén és ezzel megtudtam azt oldani, hogy a felhasználó simán megtudja változtatni hogyha esetleg cím változás történt a cégnél, vagy csak egy új raktárt telepít. Emellett még két funkció maradt még a felsorolásból az egyik a küldemény azonosító létrehozása ezt a legutoljára elkészült csomag küldemény azonosítójából számolom ki. Az utolsó ami maradt az maga a címke legenerálása amihez egy template fájlt használok amire rárakom az adatbázisból lekért adatokat.

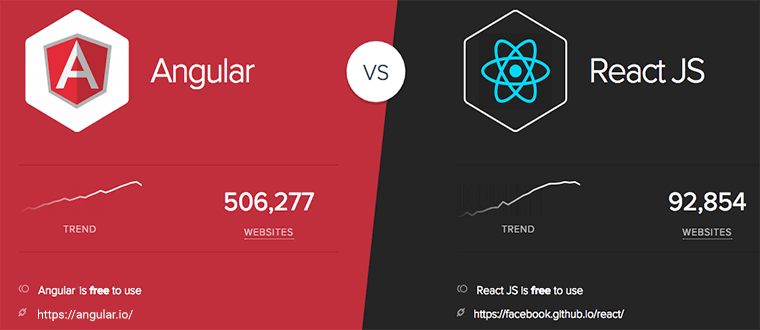
A felületen arra törekedtem, hogy egyszerű legyen minden jól látható legyen és egyszerű, mert általában az emberek nem szeretik a bonyolult dolgokat mert biztos, hogy máshol megtalálják egyszerűbben.



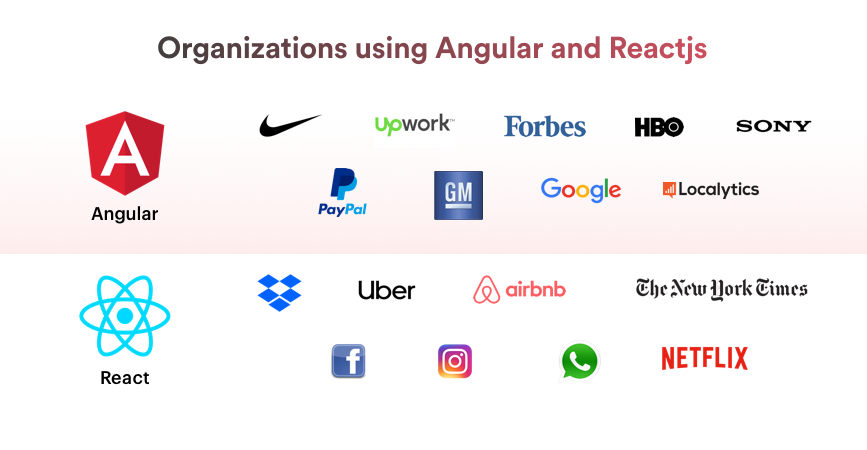
A felületen figyeltem arra, hogy egy előzetes ellenőrzés meg legyen ezzel is könnyebbítve a backend feladatait.

# 3.Felhasznált keretrendszerek

Egy webalkalmazás általában 3 komponensből épül fel van egy frontend egy backend és egy adatbázis. A frontend az maga a megjelenítés hogyan néz ki a felület, vagyis ide rakjuk az olyan dolgokat és logikai megoldásokat, amiket egy felhasználó láthat itt nagyon kell ügyelni, mert ha bent marad egy pár dolog, mint például egy adatrögzítés struktúra akkor azt egy támadó simán feltudja használni és szét tudja spammelni az oldalunkat ez által vagy hamis adatokkal tölti fel az adatbázisunkat. A következő komponens a backend, ami a frontendről beérkező adatokat dolgozza fel és továbbítja az adatot vagy az adatbázis felé vagy a frontend felé. A harmadik és utolsó az pedig az adatbázis ide kerülnek azok az adatok, amiket tárolni szeretnék az adatbázis azért jó mert egyszerűbb tárolni az adatokat nem kell különböző fájlokat kezelni beolvasni, hanem egyszerűen el lehet ide tárolni az adatokat.



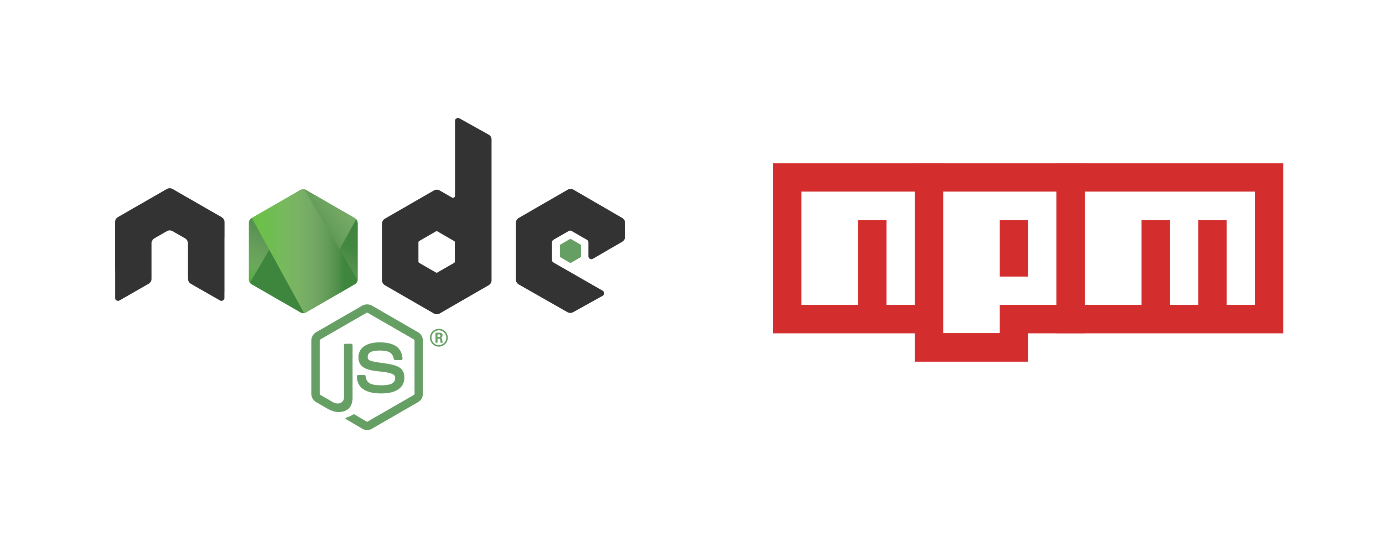
5. ábra Angular és ReactJS használata

Amikor elkezdtem keresni Frontend keretrendszert akkor két lehetőséget találtam, amit használnak az egyik a Google által létrehozott Angular vagy a facebook által létre hozott ReactJS mind a kettő ingyenes és teljesen nyitott forrás kódú az angular egy teljes értékű keretrendszer a ReactJS pedig egy Javascript könyvtár. Az angular igazi DOM-ot használ, ami azt jelenti, hogy template alapján generálja le az adatokat míg a ReactJS virtuális DOM-ot használ, ami ugye azt jelenti, hogy le készít egy oldalt azzal, hogy hozzá adja folyamatosan a tag-eket. A ReactJS JSX-et használ és ES6-ot míg az angular HTML és TypeScriptet. Az angular adatkötés két oldalú mert ha változik a model akkor változik a kinézet is míg a ReactJS-nél egy oldalú adatkötés van ez szülő gyerek kapcsolatot valósít meg vagyis a szülő komponensből csak a tulajdonság alapján tudunk eljutni a gyerek komponensig míg ugye ebből már vissza nem tudunk menni mert csak egyirányú a kapcsolat és a komponensek státuszát lehet lekérni ez azért jó mert könnyen meg lehet találni a hibát mert könnyen lehet az adatokon ellenőrzést végezni. Az Angular MVC Modellt használja, vagyis a Model View Controller-t használja míg a ReactJS a fent már említett Virtuális DOM-ot.

6. ábra Angular és ReactJS-t használó cégek

Ami miatt én az Angular-t választottam mert tény és való, hogy mind a kettő kiváló keretrendszer a ReactJS-t mostanában kapták fel nagyon és még nem lehet annyira sok minden megoldani, mint az Angularban és az is egy szempont volt számomra, hogy amiatt, hogy a ReactJS most lett felkapott ezért sokkal több anyagot lehet találni az interneten az Angular keretrendszerről. Régebb óta elérhető is az Angular.

A backend választásom a Node.js. A Node.js egy nyílt forráskódú JavaScript futtató környezet. Vagyis Javascript alkalmazásokat lehet létrehozni futtatni böngészőn kívül, mint például egy kiszolgálón. A Node.js az a V8 nevű JavaScript motor burkolója. A Node.js az azért jó még mert egy csomag kezelőt is feltelepít vagyis az npm parancs használatával egyszerűen tudunk feltelepíteni JavaScript könyvtárakat. Ez azért jó mert nagyon egyszerűen konzolon keresztül tudunk alkalmazásainkat fejleszteni különböző csomagok feltelepítésével.



7. ábra Node.js

Az utolsó komponensem az nagyon nehéz döntés volt mert kettő adatbázison gondolkoztam a mysql és a mongoDB között végül a mongoDB-t választottam. Munkám során Mysql-t használok, de az egyetem alatt sokat használtam mongoDB-t is. Ami azt jelenti, hogy mindkét adatbázissal van már tapasztalatom, de úgy éreztem, hogy a mongoDB az egyszerűbb lesz nekem mert ez gyüjteményekből áll ez megfeleltethető egy Mysql-ben ez a tábla. A mongoDB kulcs-érték párok halmaza. A Mysql egy relációs adatbázis- rendszer, amely nyílt forráskódú a mongoDB pedig NoSQL adatbázis, amely egyedi azonosítóval operál.



8. ábra Mysql vs mongoDB

## Java csomagok:

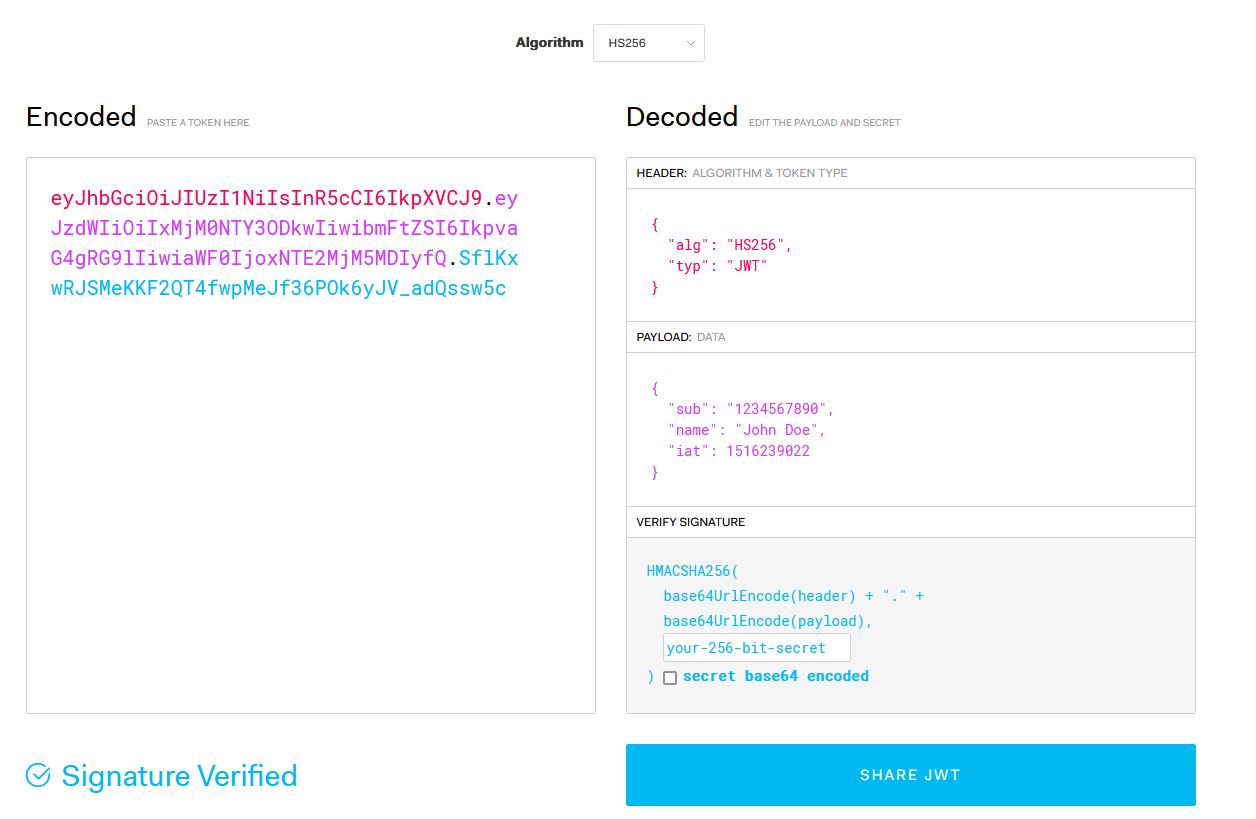
**Express.js:**

Express js fogom használni ezt használom munkám során is ezért ez lett a választásom ezzel a nagyon egyszerűen lehet különböző út választási módokat lehet megadni lehet különböző közvetítőket beállítani, ami ugye jól jön mikor egy bejelentkeztetést végzel. Ezzel nagyon egyszerűen lehet egy template fájlnak át adni adatokat és ezeket nagyon könnyen le lehet generálni. Út választási térképeket is létre lehet vele hozni és MVC alapján építhető fel

**Bcrypt:**

Ezzel a csomaggal nagyon egyszerűen lehet létrehozni jelszó hash-eléseket ugye ezzel el tudod rejteni a kényes adatokat és elmenteni egy adatbázisba, de igazából ezt csak jelszónál érdemes használni azért, mert nagyon lassan ellenőrizni le, hogy a hash-elés megegyezik és hogyha ezt egy más adat, amit sokszor kellene végig ellenőrizni akkor az nagyon sokáig tölthetne egy oldal. Úgy müködik hogy először létrehozok egy Só nevezetű titkosító jegyet amivel majd a Só generálása után azzal a szóval hash-el. Ezzel nagyon egyszerű az ellenőrzés mert csak meg kell adni egy metódusába az alap adatot és ő le ellenőrzi, hogy meg egyezik e a hash-elt adattal.

**JSONWebtoken:**

****

**9**. ábra JSONWEBTOKEN kódolása

Ez a csomag arra jó, hogy legenerál egy tokent egy bizonyos ideig és segít ez a tokent be regisztálni és ha meg egyezik a különböző útválasztásoknál az azonosítás akkor beléptetve tudja tartani az embert.

**Cors:**

A CORS vagyis Cross-Origin Resource Sharing azért kell mert a külső forrásokat ez emeli be az alkalmazásunkba. Ilyen külső forrás például kép vagy szkript.

**Angular/material:**

A materialt azért használtam, mert egyszerű lehetőséget nyújt különböző komponenseket lehet behúzni és nem kell különböző css megoldásokkal szórakozni azért, hogy jól nézzen ki

**Ngx-mask:**

Ezt azért használom mert ezzel nagyon könnyen meg lehet adni, hogy egy űrlap milyen formátumú adatokat fogadjon be, mint például egy telefonszámot csak egy bizonyos formátumban tudod meg adni, de ezt ugyan úgy el lehet játszani idő formátummal is például.

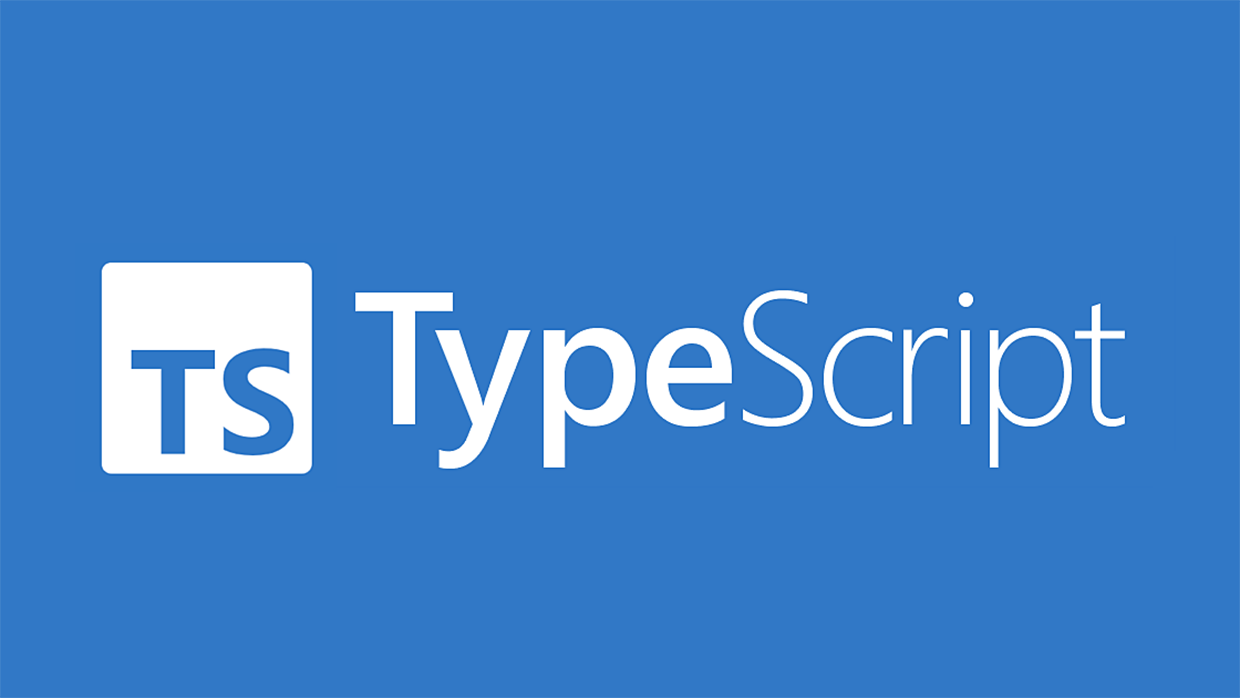
**Tailwind CSS:**

A tailwind CSS az egy CSS keretrendszer, ami ingyenes és nyílt forráskódú. A tailwind azért jó mert a CSS class neveket nem kell kitalálni. Az osztályok újra felhasználhatók emiatt a CSS nem fog nőni. Minden lokális stílus.

A másik előnye, hogy tartalmazza a PurgeCSS-t, ami azt csinálja, hogy a nem használt CSS fájlban lévő osztályokat kitörli, ami ugye azért jó mert lehet ezzel a felesleges kódot.

**TypeScript:**

A TypeScript egy objektum orientált script nyelv a Microsoft készítette. Ezt lehet mondani egy JavaScript kiegészítésnek mert típusos nyelvről beszélünk. Nem lehet ezt futtatni magában ehhez kell a TypeScriptet feltelepíteni npm csomag kezelő által. A nyelv maga nyílt forrás kódú és nincs operációs rendszerhez kötve. Ez azért van így mert mikor egy TypeScriptes fájl lefut akkor sima JavaScriptet generál le ezáltal a böngészők egyszerűen érteni tudják.



10. ábra TypeScript

# Források:

<https://cdn.elextensions.com/wp-content/uploads/2018/12/UPS-Label.png>

<https://sw9762.sfstatic.io/upload_dir/pics/GLS-returlabel-ny.jpg>

<https://csomag.hu/hu/csomagfeladas>

[https://miro.medium.com/max/1400/1\*CQhxGxTV6TinxBKApUcN0Q.png](https://miro.medium.com/max/1400/1*CQhxGxTV6TinxBKApUcN0Q.png)

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

<https://docs.microsoft.com/hu-hu/learn/modules/intro-to-nodejs/2-what>

<https://hu.education-wiki.com/7788121-mysql-vs-mongodb>

<https://jwt.io/>

<https://cdn.thenewstack.io/media/2022/01/10b88c68-typescript-logo.png>