# OSA Oktatás Segítő Alkalmazás

Tartalom

[OSA Oktatás Segítő Alkalmazás 1](#_Toc153181123)

[Mi az az OSA? 3](#_Toc153181124)

[Funkciói 3](#_Toc153181125)

[Használt technikák 4](#_Toc153181126)

[Frontenden használtak: 4](#_Toc153181127)

[Backenden használtak 4](#_Toc153181128)

[Adatbázishoz használtak 4](#_Toc153181129)

[Adatbázis 5](#_Toc153181130)

[Táblák 5](#_Toc153181131)

[Kapcsolatok 6](#_Toc153181132)

[Json szerepe az adatbázisban 7](#_Toc153181133)

[Felhasználói felület 8](#_Toc153181134)

[Backend server 9](#_Toc153181135)

# Mi az az OSA?

Napjaink erősen digitalizált világában az oktatást sem hathatjuk ki a korszerűsítésből ezzel a céllal állt neki eme feladatnak a csapatunk hogy alkosson erre a feladatra alkalmas szoftvert. Így született meg az OSA egy alkalmazás amely webes, asztali és telefonos alkalmazásként futtatva egy olyan felületet biztosít a diákoknak és tanároknak ami megkönnyíti a mindennapjaikat az oktatásban mint például: gyorsan elérhetik a számukra fontos információkat vagy könnyedén eltudják végezni az adminisztrációs feladataikat.

# Funkciói

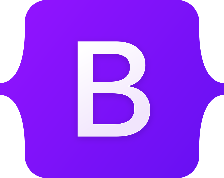
* „jelenlegi óra”
  + Egy értesítő hogy a felhasználónak milyen órája van jelenleg és lesz a következő feltéve ha van neki
* Osztályok nyilvántartása
  + Vizuálisan felsorakoztatja a diákokat ahol a szaktanár és az osztályfőnök a szerepétől függően tud interakciót folytatni a diákkal (adatok lekérése, üzenet küldése) ki jelzi a hiányzó és a hetes diákokat is egyaránt
* Teremek adatainak nyilvántartása
  + Az adott teremhez további adatokat fűzve az órarend készítésénél optimálisabban összelehet állítani az órarendet ne forduljon elő olyan baleset ahol az elméleti terembe kerül a gyakorlati óra illetve az újabb diákok eseténél kiírva a terem teljes nevét könnyíteni a terem megtalálását
* Tanmenet összeállítás
  + egy felületet biztosítunk a tanároknak ahol össze állíthatják a tanmenetjüket
* Mulasztások kezelése
  + A mulasztások feljegyzése ehhez ha van akkor indok vagy igazolás hozzáfűzése (pl.: iskolai indokú hiányzás, betegség, családi indokból való távollét vagy ismeretlen). Plusz funkció hogy a diák küldhet az OSA felületén keresztül egy igazolást képpel, szöveggel.
* Tanóra dokumentálása
  + Tanóra megtartásáról egy dokumentálás ami tartalmazza a tantárgyat, az óra témáját, hiányzókat. Ha az órát nem dokumentálja a tanár akkor jelzést küld ha ennek ellenére sem dokumentálja le egy alapértelmezett dokumentálást kap az óra.
* Értékelések
  + jegyek beírása, azok nyilvántartása, átlag és súlyozott átlag számítása
* Kapcsolattartás lehetősége
  + Faliújság közérdekű vagy osztályszintű hírekhez, személyenként üzenetküldés lehetősége
* Tanított tanulók
  + amint a tanuló elvégezte a tanulmányait megadjuk a lehetőséget hogy bejegyezzük a tanulót mint “tanított diák” ezzel megfelelően elmentve az adatait
* Órarend
  + tanáronként és diákokként megnézhető órarend

# Használt technikák

## Frontenden használtak:

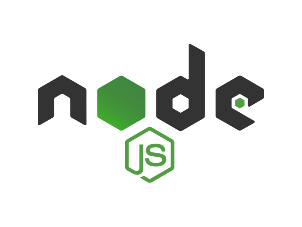
**HTML 5** : A egy leíró nyelv, melyet weboldalak készítéséhez fejlesztettek ki. A weboldalak létrehozásához használtuk.

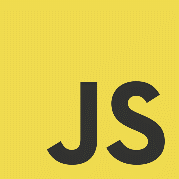
**CSS**: A számítástechnikában egy stílusleíró nyelv, mely a HTML vagy XHTML típusú strukturált dokumentumok megjelenését írja le. a weboldalaink stíluslapjait ebben írtuk meg.

**Bootstrap**: A Bootstrap egy nyílt forráskódú keretrendszer, mely HTML, CSS, JavaScript technológiákat használ. Ezt a technológiát a reszponzivitás megoldására használtuk

**Angular**: Az Angular egy olyan JavaScript alapú rendszer, amellyel dinamikus webes alkalmazásokat készíthetünk ezt a frondend kiszolgálásához használtuk.

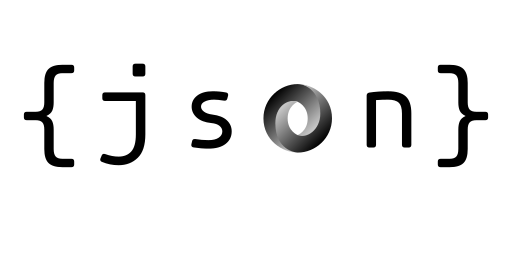
## Backenden használtak

**NodeJS**: A Node. js egy szoftverrendszer, melyet skálázható internetes alkalmazások, mégpedig webszerverek készítésére hoztak létre. A backend szerverünk ebben a nyelvben íródott.

**JavaScript:** A JavaScript (röviden JS) egy programozási nyelv, amelyet kifejezetten az internetre fejlesztettek ki. A weboldalon található bizonyos funkciókhoz használtuk.

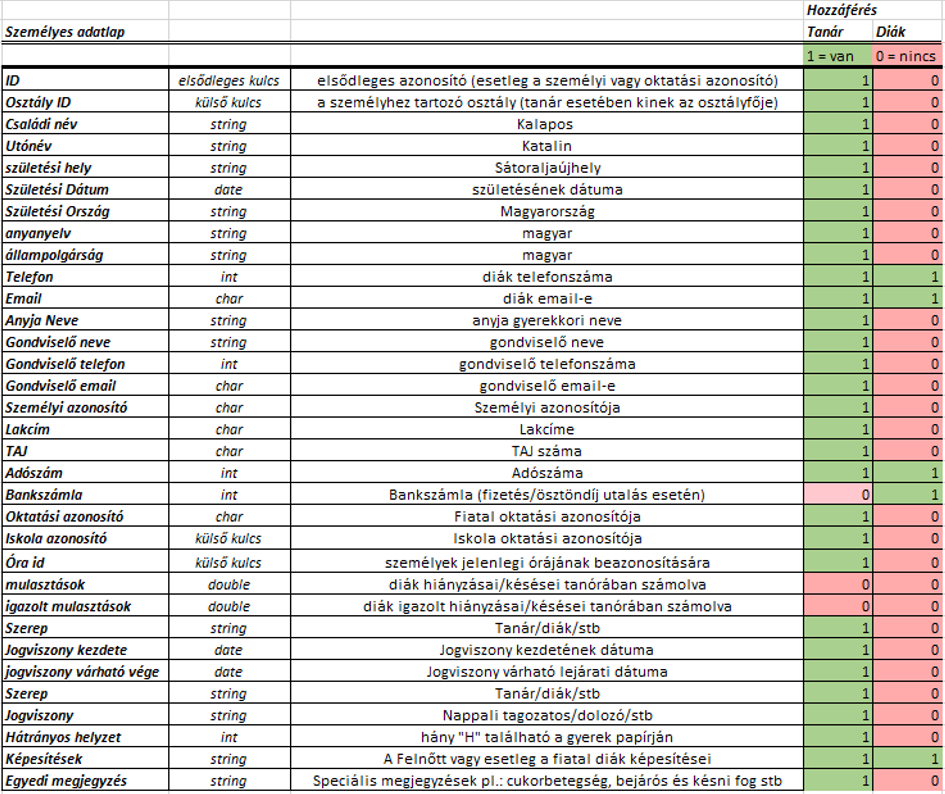
## Adatbázishoz használtak

**MySQL:** A MySQL, ahogy a neve is mutatja, egy SQL-alapú adatbázis-kezelő rendszer vagy DBMS. Az adatbázisunk első sorban ezt használja.

**JSON:** JavaScriptobjektumjelölés egy egyszerű, ember által is olvasható szöveg alapú szabvány számítógépek közötti adatcserére.

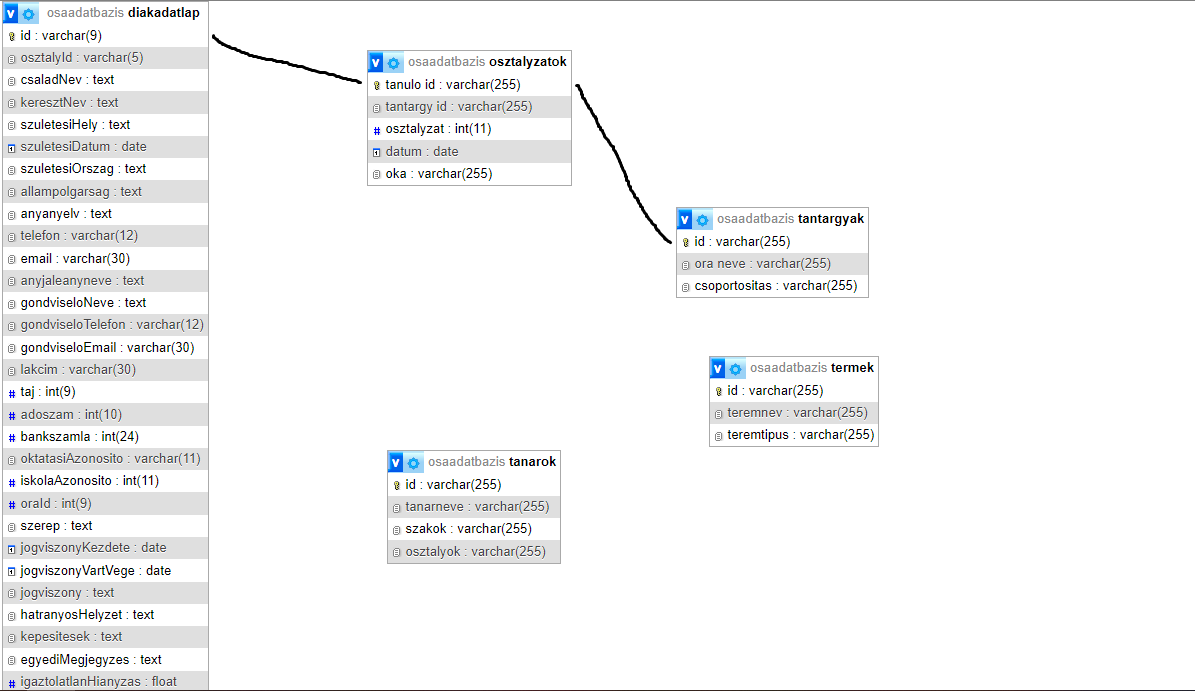
# Adatbázis

## Táblák

Személyes adatlap tábla itt a felhasználó adatait tároljuk jobb oldal látható hogy melyik adathoz van hozzáférése milyen típusú felhasználónak

Egyébb tábláink ahol a tantárgyakat, termeket és a tanárokat tároljuk le

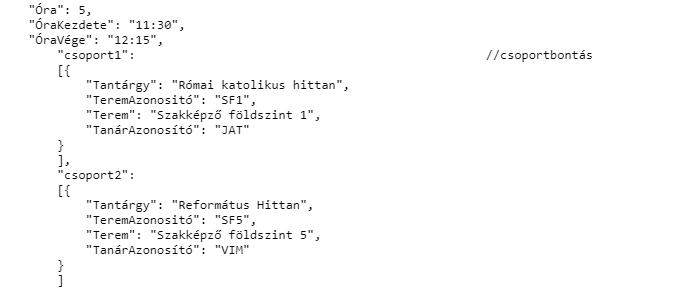
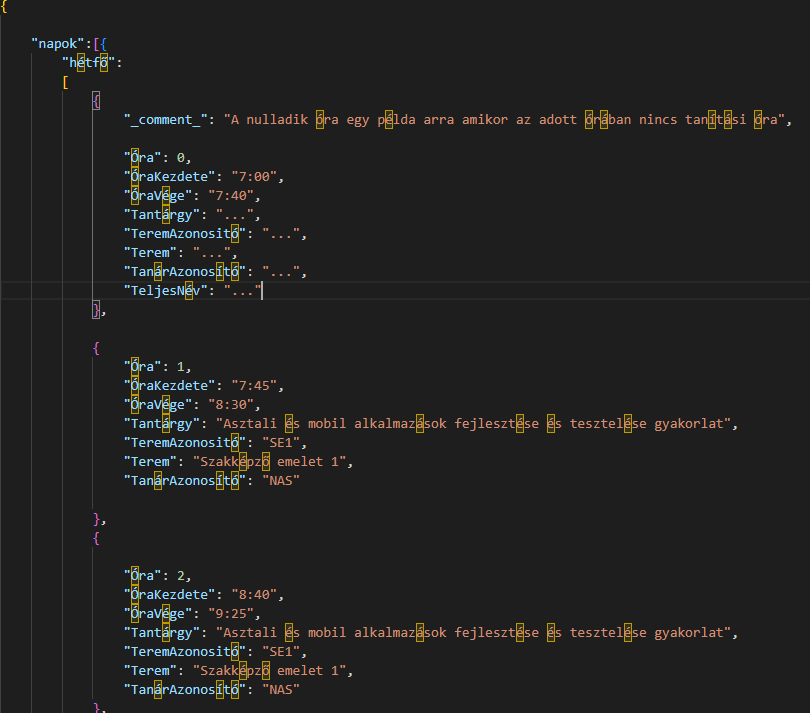
## Kapcsolatok



Fontos megjegyezni hogy az adatbázis nincs teljesen kész fázisban és fejlesztésre és simításra vár a jelen időpontban(2023.11.28)

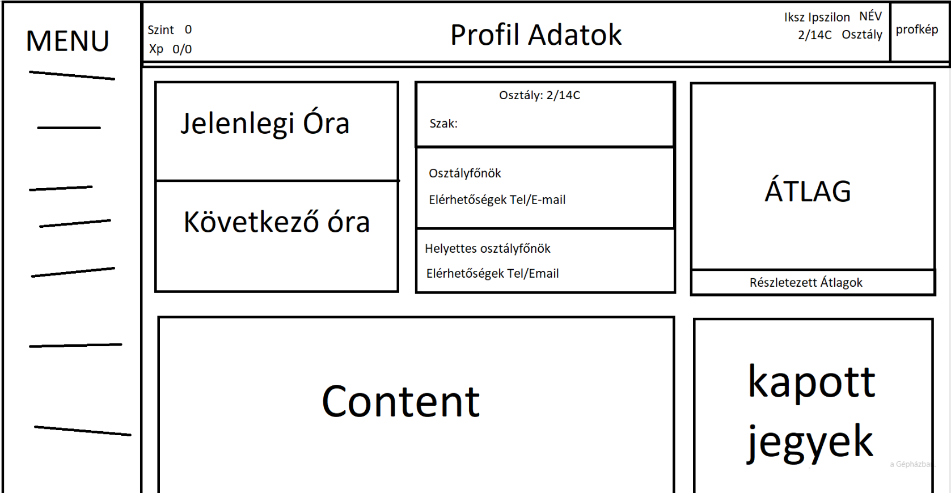
## Json szerepe az adatbázisban

Jelen álláspont szerint (2023.11.28) a json fájlok az órarend tárolását és egyéb felhasználói adatot tárolnának de probléma lépett fel az sql és a json együttműködésével.

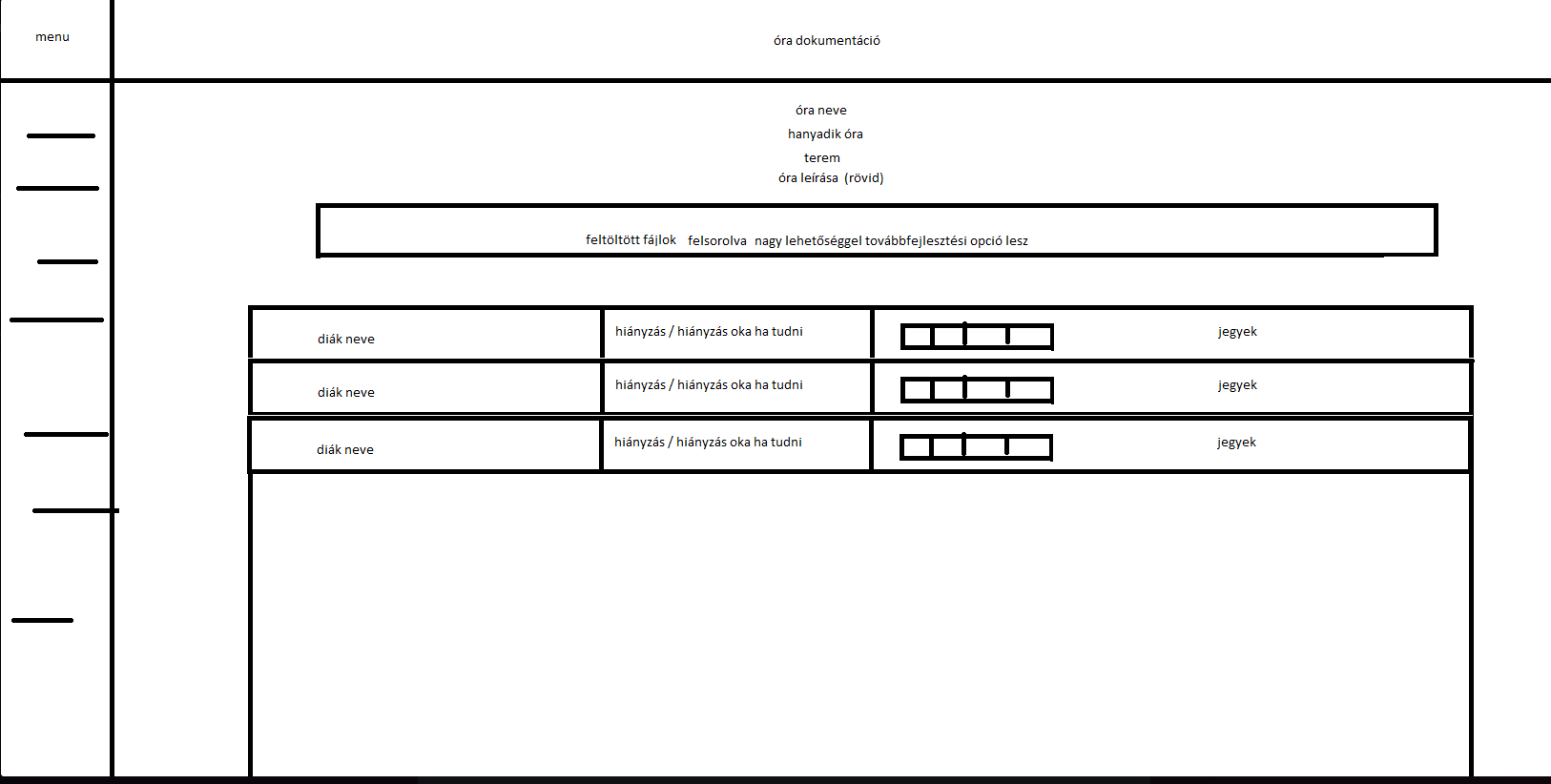
Eddig elkészült Json fájl egyike az órarend.json melynek tartalmát és felépítését a következő két kép ábrázolja

# Felhasználói felület

A felhasználói felület során próbáltuk a legjobbat kihozni a hatalmas információ mennyiség minnél egyszerűbb és könnyen navigálhatóvá tenni ezeket jelenleg (2023.11.28) csak vázlatrajzokkal ábrázolnánk.

Főoldal:

Óra dokumentációs felület



# Backend server

|  |  |  |  |
| --- | --- | --- | --- |
| route | method | működés | visszatérés |
| /regisztracio | POST | Bejövő json adatok alapján regisztrája a felhasználót  regisztrációs adatok :  {„name”:felhasználónév”, „password”:”jelszó”  } | res.status(200).json({          "response":"sikeres regisztráció"      });  sikeres regisztráció |
| res.status(400).json({  "error": "adjon meg egy felhasználónevet"          });  felhasznlónév értéke unidentified. Adja meg a felhasználónevet |
| res.status(400).json({      "error": "adjon meg egy jelszót"          });  fenti hiba jelszavas megfelelője |
| /bejelentkezes | POST | a bejövő json adattal megpróbál bejelentkezni  bejelentkezési adatok:  {  „name”:felhasználónév”, „password”:”jelszó”  } | res.status(200).json({    "response":"sikeres bejelentkezés!",    "name": bejelentkezesAdatok.name  });  bejelentkezés sikeres  **hiba esetek**  res.status(400).json({   "error":"felhasználónév nincs megadva"          });  felhasznlónév értéke unidentified. Adja meg a felhasználónevet  res.status(400).json({        "error":"jelszó nincs megadva"  });  előző probléma jelszavas megfelelője   res.status(400).json({     "error":"nincs ilyen felhasználónév"              });  a program nem találta a megadott felhasználónevet ellenőrizze a felhasználónév helyességét vagy regisztrálja a felhasználót  res.status(400).json({  "error":"ilyen bejelentkezési adatokkal nincs fiók"              })  felhasználónév vagy jelszó hibás próbálja meg újra |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |