

# Programrendszerek beadandó

## Eredmények

- Az adatbázisban lévő söröket listázzuk és többlet információt is megjelenítünk róluk. Ugyanígy működik a sörfőzdéket bemutató oldal is. Ezt a két oldalt regisztráció nélkül is eléri a felhasználó.
- A felhasználó képes regisztrálni, ha szeretne, ezáltal már képes lesz sörfőzés akciót indítani
- A felhasználó képes jelszavának megváltoztatására.
- Az oldal egyfajta receptkönyvnek felel meg, ami nyilvántartja, hogy egyes folyamatok mennyi ideig tartanak a sörfőzés során.
- Képesek vagyunk a főzésben lévő söröket kilistázni, ez a lista mutatja, ha valamelyik főzés épp akcióra vár (piros szín), vagy ha készen van (zöld).
- A főzésben lévő sör oldalán látható, hogy a folyamat melyik részénél tartunk, azok leírása, hogy mit kell elvégezni és, hogy mennyi időbe telik, hogy elérjük a következő részt. Ez az idő folyamatosan magától frissül. Egy folyamat elindításához a folyamat nevére kell kattintani, majd a Végrehajtás gombra. Csak a már elvégzett és éppen folyamatban lévő eseményt tudjuk megtekinteni.
- Ha van olyan főzésünk, amely akcióra vár, vagyis egy esemény ideje lejárt, és egy új nyílt meg, akkor a fejlécen található "Aktív főzések" gomb pirosra vált.
- A felhasználó képes főzéseit törölni, legyen az kész vagy még aktív.
- Azt, hogy a folyamat részei mennyi ideig tartanak, az adatbázisban adhatjuk meg. Mongo Atlast használtunk, ezért minden adatunk ott található.
- **Tesztelésre, a konzerv sört és a búzasör "classique"-t ajánlom, mert gyors lefutásúak.**
- Az adatbázisban végül 4 kollekciónk született: sörök, sörfőzdék, főzések, felhasználók. Adatbázisban szereplő nevük: beers, breweries, brews, users
- A backend elérési útvonalaihoz dokumentáció is készült, a könnyebb munkamegosztás érdekében. Ehhez a link a következő:  
<https://stoplight.io/p/docs/gh/skezzowski/house-of-beer>

## Sámuel Palotás

### Részem: Frontend

A frontenden a feladataim között volt a fejléc kialakítása, a sörüzemek, valamint a sörök listájának létrehozása, megjelenítése. Emellett a brewery-details és a brew-details komponensek megalkotása, ahol egy-egy sörüzem és sör részletei kerülnek bemutatásra. A home és error oldalak mellett a bejelentkezett felhasználók számára létrehoztam a felhasználói adatlapot, valamint a jelszóváltoztatás lehetőségét alakítottam ki. A bejelentkezett felhasználóknak elkészítettem a brews komponenst, ahol a felhasználóhoz

tartozó főzésben lévő söröket listázzuk ki. A felhasználónak lehetősége van a listából való törlésre. A listában jelezzük, ha új feladata van, illetve ha kész a főzés.

## Kihívások

Számomra kihívást jelentett az egyes komponensek, listák megfelelő kinézetének, elrendezésének kialakítása. A bootstrap sok lehetőséget biztosít számunkra, amellyel reszponzív és felhasználóbarát dizájnt adhatunk a felhasználói felületnek. Azonban a megfelelő elemek ismerete nélkülözhetetlen. Ez sok időt vett igénybe, hogy megismerjem az egyes classok megfelelő használatát.

Ami még egy kis nehézséget okozott, az az volt, hogy az aktív főzések közül megfelelő módon töröljem az egyes listaelemeket és a lista pedig frissüljön. Emellett a jelszóváltásnál a sikeres módosítás esetében a visszaigazolás megoldása okozott gondot, amit végül időzítő segítségével oldottam meg.

## Sipos Balázs

### Részem:

- Backend
- Adatbázis és példa adatok
- RestApi dokumentáció

### Kihívások:

A legtöbb időt a főzés logika megalkotása jelentette, nehezen épült fel a kép bennem, hogy ezt hogyan is kéne megcsinálnunk, mi is pontosan a feladat. Végül arra jutottam, hogy az adatbázisban lévő sörtől fog függni az, hogy mi is a főzés folyamata, és csak ezen fogunk végighaladni. Kicsit féltem, hogy a passportot nehéz lesz konfigurálni typescripttel, de végül nem vettem észre nagy különbséget a javascriptes implementációtól.

Sok időt vett még el annak kigondolása, hogy milyen endpointokra is lesz majd szükségünk és azok miket adjanak vissza. Ekkor gondoltam, hogy kellene egy dokumentáció, amely alapján a frontend majd dolgozhat. Ezután megterveztem azt endpointokat, az alapján, hogy mire is lehet majd szükségünk. Majd megbeszéltük közösen őket, hogy mindenkinek megfelel-e. Ezzel a módszerrel könnyen tudtunk már haladni, már ami a kapcsolatot frontend és backend között illeti.

Először lokális adatbázissal kezdtem el dolgozni, de gyorsan rájöttünk, hogy az nem lesz megfelelő, ha mindannyian tesztelünk. Ekkor jött Terhes Balázs ötlete, hogy használjunk egy online adatbázist és így született meg a Mongo Atlas adatbázisunk.

# Terhes Balázs

Részem: Főrészt a frontenden tevékenykedtem, de sürgős esetekben a backendbe is belenyúltam, ha Sipos Balázs nem volt elérhető

Elkészítettem az active-brew, a login, a register és az app komponensét, valamint az auth-guard service-t. Továbbá részt vettem a header kinézetének és logikájának megalkotásában, a brew-details-t újra felhasználhatóvá tettem az active-brew komponensben, a header és a user service-ben létrehoztam az automatikus frissítésekhez a logikát, valamint az autentikációt server központúvá tettem.

## Kihívások

A legnagyobb kihívást az automatikus frissítések megoldása jelentette anélkül, hogy elárasszam a server-t kérésekkel. Ehhez az Observable típussal bajlódtam sokat, de sokat kellett keresgélnem az async pipe, BehaviorSubject és az rxjs operátorai után is. Így egy konkrét főzés oldala automatikusan frissül 3s-enként, így a felhasználó egyből láthatja, ha egy új eseményt kell elvégeznie. A jobb felső sarokban lévő *Aktív főzések* felirat is pirosra vált, ha van új esemény valamelyik főzésnél, valamint ez a frissítés megtörténik automatikusan egy főzés indításakor, valamint bejelentkezéskor is.

A fentebb említett dolgokra volt szükségem az autentikáció szerverközpontúvá tételénél is, ugyanis a localStorage megoldás nem tetszett amiatt, hogy egy hozzáértő user könnyedén belenyúlhat annak tartalmába. Sok küzdelem árán, de ezt is sikerült úgy megoldani, hogy csak a weboldal elindulásakor (újraindításakor, újrafrissítéskor) kell lekérni a szervertől, hogy van-e bejelentkezett user a munkamenetben.

Megemlíteném még a flex-box használatát is, ugyanis evvel sem foglalkoztam még eddig, viszont ez is elengedhetetlen része egy responzív felület kialakításának.