Továbbá megfigyelve több modernebb kialakítású weboldalt, azt tapasztaltuk, hogy az a dizájn melyet a mobileszközökön való adatmegjelenítésnél használnak teljes képernyős nézetben is egyre nagyobb teret hódít, ez a megközelítés tetszett nekünk így hasonló dizájn fejlesztésébe kezdtünk.

Ami a menüpontokat illeti az eredeti elképzelés szerint a weboldal dizájn teljes mértékben dinamikus lett volna és minden adatot adatbázisból kértünk volna le. De végül ebből az elképzelésből engedtünk az adatforgalom csökkentése érdekében.

Footer komponens:

A footer manapság egy standard komponensnek számít minden weboldalon, a megléte gyakorlatilag már-már elvárt. Ami a funkcióját illeti elsősorban két célunk volt a létrehozásával. Először is bizonyos információk megjelenítése, többek között a felhasznált technológiák logóinak feltűntetése, copyright jogok feltűntetése és a weboldal logója.

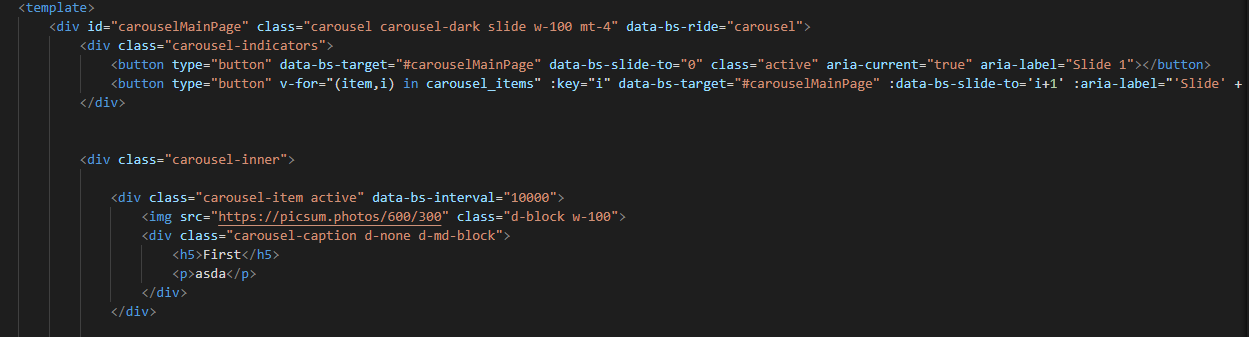
Másodszor pedig némely navigációs link ismételt megjelenítése. Erre azért van szükség, mert a látogatónak a figyelmét nagyobb valószínűséggel tudjuk megragadni, hogyha egyes menüpontok a weboldal tetején és a footer segítségével a weboldal alján is megjelennek, hiszen így a további szolgáltatásainkra például a fő oldal áttekintése után ismét felhívjuk a figyelmet.

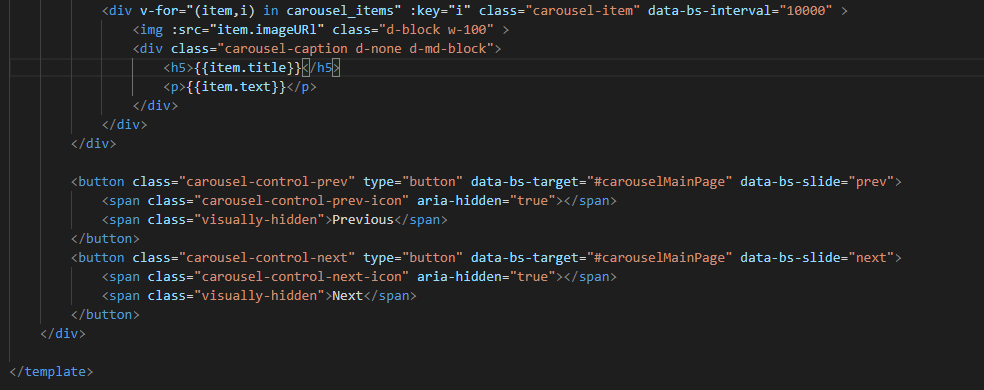
Home page:

A fejlesztés során több elképzelés is született a fő oldal megvalósításával kapcsolatban. Az alapelv az volt, hogy a fő oldalnak figyelemfelkeltőnek kell lennie, de nem is szabad túl sok információval elárasztania a felhasználót. Az imént említett navigációs panel kialakításának módját figyelembe véve és azt, hogy alapértelmezetten a navigation komponens nem jelenik meg, így, több adatot tudunk megjeleníteni úgy, hogy az még ne legyen zavaró.

A fő oldal elsődleges célja a látogató figyelmének a felkeltése és a további böngészésre való ösztönzés, ezért a fontosabb szolgáltatásaink linkjei a főoldalon akár háromszor is megjelenhetnek különböző formában.

További szempont volt a megfelelő vizuális megjelenítés is. Tehát az információ ízléses tálalása. Az egyik ilyen módszer volt a carousel komponens felhasználása, mely periodikusan egy képlistán megy végig magyarázó szöveg kíséretében. Ez a fajta megjelenítés napjainkban nagyon elterjedt és jól mutat mind típusú kijelzőn, hiszen a komponens reszponzívitás könnyen megoldható. A weboldalunkon a bootstrap carousel külső komponenst használtuk fel.





A forráskód átírásával a komponens dinamikusan a megadott adatok alapján generálódik le.

About us:

Az about us oldal egy egyszerű oldal ahol magunkról, a projekt céljáról és olyan dolgokról melyek, a látogatónak szükségesek lehetnek, osztunk meg alap információkat. Talán ez a legegyszerűbb oldal a projektben ugyanis, egyszerű kiíratásról van itt szó listákba a pragrafusokba rendezve.

Products (user):

Az oldal elsődleges célja hogy az elérhető termékek közül a látogató tudjon böngészni és egyéb műveleteket végre hajtani. A termékek úgynevezett cardok segítségével vannak feltűntetve. Mind a bootstrapnek és a primevue-nak vannak card moduljai, de ezek nemteljesen feleltek meg az elvárásainknak.

A cardokkal kapcsolatban a következő elvárásaink voltak:

1. A cardnak tartalmazni kell egy lekicsinyített képet a termékről.
2. A termék nevét és árát felkell tudnunk rajta tűntetni.
3. Ha a felhasználó vagy egy gombra kattintva vagy mouseover esemény következtében a cardnak a méretét megkell növelnünk annyira, hogy az utóbbi információk mellett még egy részletesebb leírást is megtudjunk jeleníteni.

A products oldalon a felhasználó több műveletet és képes végrehajtani, névlegesen a terméket kosárba tudja helyezni, a választott termékről bővebb információhoz juthat, ehez a művelethez a product-details oldalra való átlépés szükséges, és a lehetőség nyílik a termék kívánságlistára, való felvételére is.

Ezekhez a műveletekhez backend api-hívásokra lesz szükség.

A termék kosárba való felvételéhez, a felhasználónak nemszükséges regisztrálnia és bejelentkeznie, hiszen a kosár tartalmát nem adatbázisban tároljuk, azonban a kívánságlistára való felvételhez alap követelmény hogy a felhasználó be legyen jelentkezve, az api hívás csak így lehetséges, a hívás előtt egy session api hívásával, illetve a visszakapott boolean változó értékével validáció történik. Igaz, már az oldal renderelése előtt lefut egy hasonló validáció, hogy olyan gombokat, menüket és információkat ne osszunk meg melyekre a felhasználónak nincs jogosultsága. Ez azért fontos, mert így eltudjuk kerülni, azt hogy összezavarjuk a felhasználót.

Signup és Login:

A signup oldal teszi lehetővé, hogy a felhasználó regisztrálni tudja magát az oldalra, és hogy aztán élvezhesse annak előnyeit.

A regisztrációs folyamathoz egy form kitöltése szükséges, melyen több hiba lehetőséget is figyelembe kellet vennünk.

1. A megadott adatokat validálni kell, hogy véletlenül se lehessen hibás adatokat megadni.
2. A felhasználónak további segítséget kell nyújtani az által hogy:
   1. Segítünk megfelelő erősségű jelszót választani a Primevue: password komponens segítségével.
   2. Redukáljuk az elírás lehetőségét azáltal, hogy validációt készítünk a jelszó kétszeres lekérése után.
   3. A form kitültése után validáljuk a megadott adatokat.

Továbbá az kitöltött mezőket adatkötésekkel tároljuk.

A validációk sikeressége után, egy api hívás segítségével folytatódik a regisztrációs folyamat.

Az emailben elküldött link által való fiók aktiválását követően, a fiók adatbázisban való tárolása megtörténik.

A login oldalon hasonló képpen, a validációs műveletek után egy login api hívást követően a frontend átnavigálja a felhasználót a főoldalra, ami egy ismételt session api hívással megállapítja, hogy a kliensoldalon való session valid, így a bejelentkezett userek számára elérhető tartalmak renderelésre kerülnek.