A képen kerék, közlekedés, Szárazföldi jármű, Biciklikerék látható

Automatikusan generált leírás

Kerekegyensúly

készítette:

Munkácsi Gergő, Varga Márk, Sinkó Kristóf

Tartalom jegyzék

Tartalom jegyzék 2.oldal

Projektmunka kezdete 3.oldal

Projektmunka tervezés 8.oldal

Backend kezdete 10.oldal

Backend folytatása 15.oldal

Backend újra tervezés 18.oldal

Backend kibővítése 22.oldal

Frontend 23.oldal

Projektmunka kezdete:

Ezen a napon a csoport összeült, hogy megbeszéljük mit szeretnénk csinálni, így tartottunk egy kisebb ötlet börzét.

Több mindent kigondoltunk, mint például egy olyan oldalt, amin az emberek értékelhetik az adott sorozatot/filmet és emellett írhatnak véleményt is róluk. A következő ötlet pedig egy olyan oldal lett volna, ami szintén egy filmes és sorozatos témában van.

Ezen az oldalon pedig mutatta volna a népszerű sorozatokat és filmeket plusz betudta volna állítani a kategóriát ezzel segítve, hogy mit nézzen.

A következő ötlet egy biciklikölcsönző volt. Ezen az oldalon pedig a felhasználó kibírná választani az adott biciklit, hogy elektromos vagy hagyományos legyen. Mindezek mellett megtudja adni a felhasználó, hogy mennyi időre szeretné kibérelni az adott.

Sok tanakodás és megbeszélés után, nagy nehezen döntöttünk, hogy melyik ötlet jön be nekünk végül pedig az utolsó, a biciklikölcsönző mellett maradtunk, mert az mindegyikünknek bejött és emellett, úgy gondoltuk hasznos is tud lenni, olyan embereknek, akik kirándulni vannak és nem tudják magukkal vinni a saját biciklijüket. Így a következő oldalakon a projektünk lépéseit fogjuk bemutatni, leírni és elmondani.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

Először létrehoztunk a projektünknek egy repository-t, így a későbbiékben ide tudjuk, majd feltölteni a projektünk egyes szakaszait. Ezáltal mindegyikünk tudni fogja, hogy hol tartunk/tartottunk legutoljára. Így a GitHub lesz az egyik olyan eszköz, ami segít nekünk a projekt végre hajtására. Ezt a weboldalt többnyire projektek megvalósításához és program kódok feltöltésére találták ki.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

Itt a Git Bash látható segítségével az előző képen látható repositoryba tudunk fájlokat letölteni és emellett képes feltölteni is. A képen éppen a Git Hub-ról hívjuk le a repositorynkat aminek a neve Project, hogy lehívtuk a Projectet így már képesek vagyunk bele fájlokat rakni, ha raktunk bele fájlokat akkor azokat hasonló képpen feltudjuk tölteni.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásA két képen a Visual Studio 2022-ben létrehoztunk egy ASP.NET Core Web API-t, amit ezután elneveztünk.

Ezután pedig beállítottuk, úgy a konfigurációt ahogy mi, azt szeretnénk. Iyen például a Framework, aminek a .NET 7.0-t

adtuk meg. Authentication type-nál kivettünk mindent. Ezután engedélyeztük, hogy használjon kontrollereket, támogassa az OpenAPI-t és ne használjon top-level statemneteket. De végül nem C# program nyelv mellett döntöttötünk hanem a Java mellett.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásEzután létrehoztuk a projektünket, ahol neki tudtunk állni a backend részének. Az utolsó képen az aznapi programunkat lehet láttni. Ezt a programot a fent említett Git Bash-sel feltöltöttük a Project nevezetű repositorynkba.

Projektmunka tervezés:

Ezen a napon a csoportunk összeült egy újabb meg beszélésre, ahol elkezdtünk arról beszélni, hogy miként nézzen ki az oldal, de mindezek előtt meg néztünk egy videót az ASP.NET Core-ról, hogy hogyan kell használni.

Ez egy kisebb időbe telt mire végig néztük. Mindezek után elkezdtünk képeket és ötleteket gyűjteni, hogy miként nézzen is ki a weboldalunk főoldala. Végül együtt elkezdtük a projektünk főoldalának vázlatát megcsinálni a Figma segítségével.

A képen szöveg, szoftver, Multimédiás szoftver, Számítógépes ikon látható

Automatikusan generált leírásAz alábbi képen először létrehoztunk egy üres projektet.

Majd ezt elkezdtük feltölteni, képekkel és pár alakzattal, hogy mi hol legyen. Ezt a következő oldalon majd látni lehet, hogyan oldottuk meg.

A képen szöveg, szoftver, Számítógépes ikon, Weblap látható

Automatikusan generált leírás

A képen a projektünk főoldalának vázlata látszik. Felül az oldal jobb sarkába gondoltuk elhelyezni a weboldalunk nevét.

Mellette balara az oldal logója látható.

Az oldal baloldalán pedig elhelyeztünk két gombot az egyik a szállásokért fog felelni a másik pedig a kempingekért.

Ezek alatt található egy Nav bar.

Erre még öt darab gombot ilyen például a: hírek, kerékpártúrák, sport, tartozékok és a bérlés.

Alatta található a bejelentkezés gomb, ahol a felhasználó betud majd jelentkezni az oldalra.

Bejelentkezés alatt a kép mellett egy leírás lesz a túrákról. A másik kép mellett pedig az ikonoknál pedig, hogy mikor vagyunk nyitva, milyen biciklik vannak, fontos tudnivalók és az áraink.

Mindezek alatt pedig található, majd egy footer rész, ahol az elérhetőségünket, a helyszínt és a támogatóinkat lehet, majd megtalálni.

Backend kezdete:

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásEzen a napom a csapatunk összeült egy újabb megbeszélésre, hogy mit csináljunk következőnek, de végül arra jutottunk, hogy elkezdjük meg írni a programunkat. Ezt a java és a spring segítségével hoztunk létre egy regisztrációs és bejelentkezés programot mind ezt az IntellijD IDEA segítségével. De mind ezek előtt beállítottuk a dependency-ket. Ezt az előbbiképen lehet látni.

Itt beállítottuk, hogy milyen legyen a projekt, utána milyen nyelvben íródjon, melyik spring verziót használja, mi legyen a projekt metaadatai, ezután a packaginget állítottuk be majd, hogy milyen java verziót használjon. Mindezek végén hozzáadtuk, hogy miket használjon.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásEzek után létrehoztuk a projektet, amit itt lehet látni:

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásMiután ez megvolt az src mappán létrehoztuk a User.java-t és a Role.java fájlt és egy AuthController.java-t. A User.java fájl arra szolgál, hogy a felhasználókhoz több hozzáférést is lehet adni. Ennek a kód sora látható.

A kód megírása után nekiálltunk megírni az AuthController.java fájlnak a kódját, ami arra szolgál, hogy

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásel tudjon minket vinni a weboldalunk főoldalához ennek a kódrésze látható. Ezek után neki álltunk a Role.java fájlnak a megírásához.

Ez a kódsor annyit tesz, hogy a felhasználónak oda tudjuk adni a szükséges hozzáféréseket, amik kellenek nekik.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásA kód a képen látható.

Ezek után a repository mappán létrehoztunk újabb fájlokat az egyik egy RoleRepository.java és egy UserRepository.java fájlt.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásA kódos itt található.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásEnnek a segítségével megtudjuk adni a felhasználónak milyen jogosultsága legyen. Az utóbbi a UserRepository.java kódja.Ez segít a regisztrációban és a bejelentkezésben is egyaránt.

Mindezek után meg írtuk a főoldal beléptető és regisztrációs felületét a Visual Studio Code-dal. Mindezt html, css valamint esetenként js használatával írtuk meg aminek a kódját itt lehet látni:A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

Ennek a segítségével látható lesz a weboldalunk felülete, ahova beszeretnénk, majd lépni.

Backend folytatása:

Ezen a napon csapatunk összeült egy újabb project munka megbeszéléshez. Ezen a napon folytattuk a Backendet. Kiegészítettük a Backendet egy egyedi felhasználó részlet szolgáltatással, aminek a kód sora itt található:

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásEzután neki álltunk megcsinálni a felhasználó szolgáltatást, amit Márknak köszönhetően megcsináltunk. A program sort itt lehet látni:

Mindezek után nagy nehezen ugyan, de sikerült meg írni a következő program sorunkat megírni, ami a felhasználó szolgáltatás implementációja volt. Ennek a kódsorai itt találhatók meg:

A kód sor első rész

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

A második része

A képen szöveg, képernyőkép, szoftver, számítógép látható

Automatikusan generált leírás

Végül pedig ennek a kód sornak az utolsó része:

A képen szöveg, képernyőkép, szoftver, számítógép látható

Automatikusan generált leírás

Mindezek után meg írtuk a felhasználói adatátviteli objektumot, itt megtudja adni a felhasználó az id-t, a keresztnevét, vezetéknevét, emailét és a jelszavát. Az utóbbi kettőnél megvan adva, hogy nem lehetnek üresek.

Programsor: A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

Backend újra tervezés:

Az alábbi képen a swagger api implementálása látható. Ennek

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírássegítségével tudunk az oldalhoz hozzáadni, törölni és változtatni az adott terméken vagy biciklin változtatni. Ezt persze a vásárló nem tudja megcsinálni, csak az admin jogosultsággal rendelkező személyek.

A kategóriák résznél a program kód lehetővé teszi, hogy rátudjunk keresni a kategória ablakban az adott termékre, név, leírás és kép segítségével.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírásEz által megkönnyebbítve a vásárló dolgát a webshopban, hogy ne keljen neki a sok termék között keresnie.

A kategória kontroller felelős azért, hogy a kategória kód sor helyesen fusson le.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

A kategória repository a kategóriákat listázza ki a kategóriák alapján, így egyszerűbb az adott kategóriát megtalálni.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

A kategória szerviz beolvassa, kilistázza, létrehozza és tudja frissíteni a kategóriákat és mind ezek mellett képes menteni azokat.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

Backend kibővítése:

Ezek után a csapat belehúzott és elkezdte kibővíteni a Backendünk még egy jó pár dologgal, mint például service-eket és mind ezek mellett a controllereket is. Így a service mappánkon belül például hozzá lett adva az AuthenticationService, CartService, FileStoreService, OrderItemService, OrderService, ProductService és a UserService.

Ezután a controllereket égészítettük ki még egy pár kódsorral, ilyen például CartController, FileUploadController, OrderController és a ProductController.

Ezeknek a leprogramozás egy jó pár hetet és órát igénybe vett. Ezeket a kódsorokat Márknak köszönhetjük, akinek a beleadása nélkül nem sikerülhetett volna végrehajtani, ezt a sok munkát.

Frontend:

Ezen a képen a kategória hozzáadását adtuk hozzá a frontendhez

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

A kód kimenetele pedig az alábbi képen látható

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

Ezen a képen pedig a kategória hozzáadásának a programkódja lett folytatva.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás