

Baranya Vármegyei SzC Radnóti Miklós Közgazdasági Technikum

Szakma megnevezése: Szoftverfejlesztő és -tesztelő

Szakma azonosító száma: 506131203

Vizsgaremek

CyberRead

Készítették:
Kiss Vivien Brigitta
Mayer Hedda Adrienn
Szirényi Vivien

Pécs

2024

Projekt ötlet bemutatása:

Az általunk választott projekt munka célja, hogy összekösse az olvasói közönséget az írókkal és a kiadókkal, egyfajta social media platform keretein belül. Lehetőséget biztosítunk a könyvkiadóknak, hogy új ügyfelekre találjanak, illetve az íróknak, hogy akár öncélúan, akár kiadó cégen keresztül publikálhassák műveiket. A weboldalunk főként az írók életét szeretné megkönnyíteni. Szeretnénk lehetőséget biztosítani mind a feltörekvő, mind a tapasztalt írók számára, hogy a hobbijukat komolyabb szintre is emelhessék és megkapják a megérdemelt figyelmet. Ezt elősegíti az is, hogy az olvasók az oldalon keresztül meg is vásárolhatják a nekik tetsző alkotásokat. Vásárlás előtt a felhasználó a könyv első 20%-át ingyenesen akár elolvashatja.

Azért szerettük volna megvalósítani ezt az ötletet, mert kevés olyan kifejezetten publikálással foglalkozó oldalt ismerünk, ahol az írók ingyenesen próbára tehetik a tehetségüket, úgy hogy az komolyabb eredménnyel szolgáljon a számukra. A legtöbb ilyen oldal általában költségekkel terheli az íróit, az ingyenes oldalakon pedig könnyű elveszni az internet forgatagában. Mi ezt a dolgot egy másik irányból közelítettük meg. Az oldalunk jelenlegi állapotában nem rendelkezik előfizetési csomagokkal – bár gondoltunk rá, arra az esetre amikor már rendelkezünk egy stabil felhasználói réteggel –, így a későbbi üzemeltetés során felmerülő költségek fedezésére szükséges összeghez az eladott könyveken keresztül jutnánk hozzá. Mind az egyénileg, mind a kiadó által forgalmazott könyvek – már előre meghatározott – árához 20%-ot hozzá számolunk, így kapjuk meg a könyvek tényleges – tehát a felhasználók számára feltüntetett – árát.

A funkciók megalkotása során nagy figyelmet fordítottunk rá, hogy ne csupán a népszerűsége múljon a feltörekvő írók sikere, hanem mindenki egyenlő eséllyel induljon el a maga útján. A feltöltött tartalmak szabályozására tervezünk a későbbiekben egy „report” funkciót létrehozni a posztok, a feltöltött könyvek, illetve a felhasználók tekintetében is, a beérkezett panaszokat pedig egy admin felhasználón keresztül tudná a fejlesztő csapat felülvizsgálni. Ezen az admin felületen keresztül lenne lehetőség további felhasználókat admin jogosultsággal felruházni. A regisztrációval minden felhasználó elfogadja az Általános Szerződési Feltételekben feltüntetett szabályozásokat, melyek bárki számára elérhetőek lesznek a kezdő oldal footer-jéből. Tervezünk egy „Help Center” nevű oldalt készíteni melyen az általános használattal kapcsolatos, illetve a gyakran felmerülő kérdésekre találhatnak válaszokat a felhasználók, akár már azelőtt, hogy regisztrálnának az oldalunkra.

A projektet az elkövetkezendő hónapokban szeretnénk feltölteni egy nyilvános szerverre is.

Az alkalmazás felhasználói bemutatása:

Alapvetően két profil típust hoztunk létre, amelyek közül a felhasználó választhat a regisztráció során. Egy „general” profil lehetőséget – azaz egy általánost, amely az olvasóknak és az íróknak készült –, illetve egy „publisher” profil lehetőséget – azaz egy kiadói felületet, amely a könyvkiadó cégek számára készült –. Mindkét profil más-más jogosultságokkal rendelkezik az oldalon.

Funkciók melyek mindkét profil típusban megtalálhatóak:

Mindkét profil típusra jellemző, hogy lehetőséget biztosítunk a felhasználóknak más profilokat követni. A „Feed” menüpontban ezektől a felhasználóktól olvashatnak, kedvelhetnek és megoszthatnak bejegyzéseket, illetve ők maguk is posztolhatnak. A bejegyzések megosztása link alapú lesz. Ezen felül mindkét profilnak meg van a lehetősége elmenteni a neki tetsző könyveket, melyet később a „Saved Books” fülön tudnak

megtekinteni. A felhasználónak lehetősége nyílik a feltöltött könyvek első 20%-át ingyenesen elolvasni.

Profiljukon megadhatnak egyéni profilképet, bemutatkozó szöveget, vagy akár elérhetőségi adatokat – a regisztrációkor már megadott adatok (például az e-mail cím) szabadon dönthet, hogy megjelenjen-e a profilján vagy sem –. Többek között ezeket az adatokat is lehetősége van a beállítás menüpontban később szerkeszteni. Ezen felül az „Explore” menüponton keresztül kereshetnek tartalmat akár kategória szerint, akár a keresőmezőt használva. A footer-ből minden profiltípus elér a projekt közösségi média oldalait és elérhetőségeit, az „About Us” fület, a Help Center-t és az ÁSZF oldalát.

General profil funkciók:

A regisztrációs form tekintetében kizárólag annyi az eltérés, hogy itt a cég neve helyett a felhasználónak a születési dátumot kell megadnia. Az oldalunk használatához a felhasználónak be kell töltenie a 15. életévét. Sikeres regisztráció és az első bejelentkezés után lehetősége van kiválasztani mely kategóriák érdeklik. Később többek között ezek alapján is ajánlunk könyveket a számukra. A general profillal rendelkező felhasználóknak lehetőségük van saját könyvet feltölteni. A feltöltés során a következő információkat kell megadni: egyénileg feltöltött könyv borító, a könyvet tartalmazó dokumentum, könyv címe és fülszövege. A választható opciók közül ki kell választania a célközönségét, a könyv kategóriáját és a nyelvet, amelyen a könyv íródott. Ha a könyvben felnőtt tartalom jelenik meg azt is fel kell tüntetni. Ezen felül választhat, hogy ő maga szeretné-e egyénileg kiadni a könyvét – ebben az esetben szükségünk van egy IBAN számra, ahova a későbbi bevételeket várja, illetve meg kell adnia egy összeget amennyiért árulni szeretné a könyvet – vagy pedig, hogy az adott könyvet kiadón keresztül szeretné-e kiadni. Egy felhasználó csak egy kiadóhoz tartozhat, viszont a kiadóhoz tartozás nem zárja ki az öncélú publikálás lehetőségét. A publikálással a felhasználó automatikusan elfogadja az ÁSZF-ben feltüntetett copyright-ra vonatkozó szabályozásokat – melyről részletes tájékoztatást az ÁSZF oldalunkon talál –. Publikálás után a könyvek címe alatt a regisztrált, teljes név lesz feltüntetve szerzői névként.

Ezeknek a felhasználóknak van lehetőségük könyveket vásárolni az oldalunkon keresztül, amelyeket később a már fent említett „Saved Books” fülön tudnak majd elérni. A megvásárolt könyveket az oldalunkon keresztül fogják tudni elolvasni. Ennek egyik fő oka, hogy szeretnénk szabályozni, hogy az oldalunkra feltöltött tartalmakat nehezebb legyen engedély nélkül megosztani / feltölteni más ingyenes platformokra. Ezen felül szeretnénk biztosítani az íróinkat és a kiadóinkat, hogy az oldalunk használatával minél nagyobb esélyük legyen bevételt szerezni. Az utolsó oldal után felbukkan egy értékelési opció, amellyel majd az olvasott könyvet tudja értékelni. Amikor a felhasználó egy könyv borítóra kattint, akkor ezen értékelések átlaga fog megjelenni a könyv „ranking” adataként.

Lehetőségük van az „Explore” fülön keresztül a választott kategóriában lehetőségük van a „legtöbbet megvásárolt”, illetve a „legtöbbet elmentett” könyvekre szűrni a találatokat, vagy akár a kereső mezőt használva rákeresni az adott művekre.

Publisher profil funkciók:

Ennél a profil típusnál a lényeges különbség az, hogy a számukra megjelenített könyv ajánlások feltételhez vannak kötve. Mivel egy general felhasználó csak egy kiadón keresztül publikálhat, így a kiadót kereső könyv feltöltője nem állhat kapcsolatban más kiadóval, vagy az adott profil tulajdonosával kell kapcsolatban állnia. Más kiadó által kiadott könyveket nem lát. Egy könyv megnyitása során menteni tudja a könyvet, illetve bele tud olvasni, a szerző nevére kattintva pedig a feltöltő profilján van lehetőségük rajtuk keresztül e-mailben értesíteni

a felhasználót az érdeklődéséről. Ezt később a felhasználónak az emailben küldött linken keresztül vissza kell igazolnia. Az író profilján csak a feltételeknek megfelelő könyveket láthatja. Terveink szerint ha lesz egy stabil felhasználó rétegünk, szeretnénk létrehozni a kiadók számára előfizetési csomagot, melyre előfizetve akár láthatná az író öncélúan kiadott könyveit is.

Ezzel a profillal rendelkező felhasználók az „Explore” fülön kiválasztott kategória eredményeit csupán ABC sorrend szerint tudják rendezni, illetve a kereső mezőt használva specifikusan is tudnak keresni.

Kiadóink az oldalunkon kiadott tartalmakat kizárólag az oldalunkon forgalmazhatják, az üzletekben / saját webshop-on való értékesítés akár fizikai, akár e-book formátumban nem engedélyezett.

Csapatmunka bemutatása:

Projektmenedzser: Szirényi Vivien

Vivien felügyeli a csapat koordinálását és a projekt hatékony menedzselését. Figyelemmel kíséri a határidőket, erőforrásokat és kommunikációt a csapat tagjai között. Biztosítja, hogy a csapat szinkronban működjön, és a projekt minden szakasza gördülékenyen haladjon előre, miközben az elvárt eredményeket biztosítja.

Frontend fejlesztő: Kiss Vivien Brigitta

Vivien feladata a weboldal felhasználói interfészének kialakítása és implementálása a design és UX/UI tervek alapján. Ez magában foglalja a HTML, CSS és JavaScript nyelvek használatát a felhasználói élmény optimalizálása és a responszív design eléréséhez. A Frontend Fejlesztő folyamatosan együttműködik a Design Szakértővel és a Backend Fejlesztővel az egységes funkcionalitás elérése érdekében.

Backend fejlesztő, Unit tesztelő: Mayer Hedda Adrienn

Hedda felelős a weboldal háttérfunkcióinak és adatkezelésének kódolásáért. A szerveroldali fejlesztés során alkalmazza az adatbázisokkal, API-kkal és szerveroldali logikával kapcsolatos szakértelmét. Kódolása magában foglalja az adatbázisokkal és szerveroldali logikával kapcsolatos szakértelmet. Szorosan együttműködik a Frontend Fejlesztővel a funkcionalitás összehangolása érdekében. Az adatbázisok tervezését és kezelését is felügyeli a weboldalon. Ő felel még az egyes kódrészek egységteszteléséért. Automatizált tesztekkel ellenőrzi, hogy a különálló kódrészletek megfelelnek-e a fejlesztői szabványoknak és a tervezett működésnek. Folyamatos visszajelzést ad a fejlesztőcsapatnak a kód minőségéről és esetleges hibákról.

Designer, Manuális tesztelő: Szirényi Vivien

Vivien felelős a weboldal vizuális megjelenéséért és az UX/UI tervek kialakításáért. A színpaletta, tipográfia és egyéb vizuális elemek kialakításával támogatja a vállalati arculatot és a felhasználói élményt. Szorosan együttműködik a Frontend Fejlesztővel és a tervezési elvárások és funkcionalitás összehangolásáért. Ő felelős még a teljes weboldal manuális ellenőrzéséért. Tesztek tervezésével és végrehajtásával biztosítja, hogy az alkalmazás minden része megfelelően működjön, és megfeleljen a specifikációknak. Aktív együttműködik a fejlesztőcsapattal, és jelentéseket készít a felfedezett hibákról.

A csapatmunka során szerzett értékes tapasztalatok számos területen kiterjedtek, hozzájárulva a projekt hatékony megvalósításához. Az egyértelmű szerepek és felelőségek kijelölése lehetővé tette, hogy minden csapattag hatékonyan koncentráljon a saját területére, ezáltal növelve a hatékonyságot és elkerülve a feladatok közötti átfedéseket. A szoros együttműködés a csapattagok között elősegítette a gyors problémamegoldást és a hatékony kommunikációt, bár a távoli munkavégzés időnként kihívásokat vetett fel ezen a területen. A folyamatos tesztelés, beleértve az egységtesztelést és a manuális tesztelést, kulcsfontosságú volt a kódminőség javításában és a hibák korai azonosításában. Az erőforrások hatékonyabb kihasználását célzó lépések, például a felhőszolgáltatások használata, pozitívan járultak hozzá a projekt skálázhatóságához. Ugyanakkor a projekt során felmerült kihívások közé tartozott a távoli munkavégzésből adódó kommunikációs nehézségek, amelyek ellenére a csapat erőfeszítéseket tett a hatékony együttműködés érdekében. Ezen tapasztalatok alapján hangsúlyozzuk a csapat rugalmasságát és a folyamatos tanulás fontosságát. Az egyértelmű szerepkörök és a hatékony kommunikáció kritikus tényezők maradtak, melyek a jövőbeli projektek tervezésénél és végrehajtásánál továbbra is kulcsfontosságúak lesznek a sikeres csapatmunka és a projektmenedzsment szempontjából.

Fejlesztési és tervezési módszertan:

A fejlesztési és tervezési módszertanunk mélyreható kidolgozása során az elsődleges célunk az volt, hogy a projekt specifikációinak és a csapat egyedi igényeinek maximálisan megfeleljünk. A tervezés fázisában kiemelt hangsúlyt fektettünk a felhasználói igények kielégítésére, miközben egy olyan vizuális dizájnt hoztunk létre, ami összhangban áll a vállalati arculattal és a felhasználói élményt optimalizálja.

A projektmenedzsment és fejlesztési folyamatokat olyan módon alakítottuk ki, hogy a hatékonyság és minőség együttesen érvényesüljön. Egyértelmű szerepek és felelőségek kiosztása segítette a csapatot hatékonyan együttműködni, és minden tagnak lehetőséget nyújtott arra, hogy a saját területén kiemelkedjen.

A tesztelési fázisban az egységtesztelés és a manuális tesztelés szorosan együttműködve biztosították a kód minőségét és a funkcionalitás teljes körű validációját. Az egységtesztelés révén a kód stabilitása került ellenőrzésre, míg a manuális tesztek a teljes rendszer működését validálták.

Az elkészült folyamatokat és dokumentációkat részletesen rögzítettük, beleértve a tervezési specifikációkat, fejlesztési iterációkat és tesztelési eredményeket. A rendszeres értekezletek és online követési eszközök pedig biztosították a csapat hatékony együttműködését és a projektfolyamatok átláthatóságát. Ez az átfogó és jól strukturált módszertan hozzájárult a projekt sikeres és minőségi megvalósításához.

Használt technológiák bemutatása

Programozási nyelvek:

A projekt során több programozási nyelvet alkalmaztunk a különböző részek fejlesztéséhez. A Backend részt Java nyelven valósítottuk meg, mivel erős támogatással rendelkezik a háttérlogika és adatkezelés területén. A Frontend rész HTML, CSS és JavaScript felhasználásával készült.

Adatbázis típusa:

Az adatbázis típusaként egy relációs adatbázist használtunk a projekt során. A relációs adatbázisok széles körben alkalmazottak, mivel hatékonyan kezelik a szorosan összekapcsolt

adatokat és lehetővé teszik a bonyolult lekérdezések végrehajtását. A strukturált adatok kezelése és az adatok közötti kapcsolatok fenntartása révén a relációs adatbázisok segítenek a rendszer megbízható adatkezelésében és hatékony lekérdezéseiben.

Verziókezelés:

A fejlesztés és csapatmunka során a Git verziókezelő rendszert alkalmaztuk. Ennek segítségével könnyedén nyomon követhettük a kódbázis változásait, kezelhettük a különböző verziókat és segített a hatékony csapatmunkában.

Project Management Szoftver:

A projektmenedzsmentet a Jira nevű szoftverrel végeztük, mely egy kiváló eszköz a feladatok kezeléséhez, a projektidővonal hatékony ütemezéséhez és a csapatmunka támogatásához. A Jira segítségével részletesen definiálhattuk a projekt feladatokat, azokhoz határidőket rendelhettünk, és nyomon követhettük az előrehaladást.

Project során elsajátított soft skillek:

A projekt során olyan "soft skillek" fejlődtek ki, amelyek nélkülözhetetlenek a hatékony csapatmunkához és a sikeres projektvezetéshez. A hatékony kommunikáció, a csapatmunka, az időmenedzsment és az agilis munkamódszer alkalmazása mind hozzájárult a projekt eredményességéhez.

Design szoftver:

A vizuális tervezéshez Figma nevű eszközt alkalmaztuk. Ez a kollaboratív platform lehetővé tette a tervezők és fejlesztők számára a szoros együttműködést, biztosítva a felhasználói élmény és dizájn egységes kialakítását a projekt teljes körében. A platformon belül könnyen létrehozhattunk wireframe-eket, prototípusokat és design elemeket, amelyek hozzájárultak a felhasználói élmény és dizájn egységes kialakításához.

Szakmai dokumentáció – Frontend:

Az oldalunk egyszerű HTML, CSS és JavaScript használatával épül fel, melyet a hatékonyabb responzivitás érdekében a Bootstrap legfrissebb verziójával egészítettünk ki.

Az API hívásokat kiszolgáló fő komponensünk egy API mappa, melyben egy JavaScript függvényeken keresztül bonyolítottuk le a kéréseket. Ezek az aszinkronos függvények fetch api-t használnak a POST és GET kérések kezelésére. A válaszkezelések során ellenőrizzük a visszakapott státuszkódot, és ennek megfelelően struktúrálja a válaszokat json formátumú objektumokba, mely végül a return érték lesz – ebben a státuszkód mellett megtalálható vagy a kapott hiba üzenet vagy pedig a backend által küldött objektum –. A HTTP protokoll esetében a klasszikus kliens-szerver modellt használtuk.

Az oldalaink rendelkeznek önálló JavaScript fájlokkal, melyekben többek között megtalálhatóak a konkrét API hívások kezelése összekötve az aktuális interakcióval (például az oldal betöltésével vagy egy gomb megnyomásával). Itt a fetch fejlécében minden esetben beállítottuk a Content-Type értékét application/json-re, hogy jelezzük a szerver felé a küldött adatok formátumát és csökkentsük a hibákat.

Ezekben az önálló JavaScript fájlokban a legtöbb adat esetén vizsgáljuk a bejelentkezéskor, localStorage-ban eltárolt token-t, amely a felhasználó adatai mellett magával hordozza a profilhoz tartozó jogosultsági kört (rank-ot). Ha a felhasználó valamilyen oknál fogva olyan oldalt szeretne betölteni, amihez neki nincs jogosultsága, akkor a vizsgálat lefutása után

automatikusan az ő jogosultságának megfelelő oldalra kerül átirányításra, ha pedig a token érvényessége lejárt akkor a bejelentkezés oldalra irányítjuk át. Ha a felhasználó egyáltalán nem rendelkezik token-el akkor a „Landing page” fog betölteni.

Az oldal felépítése során a kapott adatok kezelésekor a fetch használata mellett döntöttünk, mivel ezzel egyszerűbb felépítésen keresztül egy könnyebben értelmezhető, modernebb kódot kapunk, mintha az XMLHttpRequest-et használnánk.

Azon oldalak esetében ahol input mezőkkel dolgozunk (például a regisztráció, vagy a könyv feltöltés), frontend szinten is elvégeztük a megfelelő validációkat, így csökkentjük annak az esélyét, hogy a backend felé rossz adat kerüljön elküldésre és kiküszöböljük a felesleges kérés küldéseket. Ezeknél a vizsgálatoknál figyelembe vettük a függvényeink létrehozásakor, hogy minél több alkalommal tudjuk használni őket, és minél egyszerűbben fedjük le az adott problémát.

Az oldal esztétikai megjelenése szempontjából egyénileg letöltött és beimportált betűstílusokat használtunk, illetve a hagyományos, böngésző által megszabott scroll-bar is egyedi külsőt kapott. Sok helyen megtalálhatóak a root-ban változóként előre felvett színek az átláthatóság és a gyorsabb haladás érdekében. Ikonok tekintetében vannak általunk készített, egyéni svg fájlok (mint például az oldalunk logója), illetve használtuk a Bootstrap által biztosított ikonokat is.

Szakmai dokumentáció – Backend:

Ez a dokumentáció bemutatja a fejlesztés alatt álló webalkalmazás Java backendjét. A projekt többrétegű architektúrára épül, ahol különböző rétegek segítik a rendszer funkcionalitásának és karbantarthatóságának javítását.

A backend alkalmazás modell, service, controller, config és DTO (Data Transfer Object) rétegekből épül fel.

Modell réteg: A modell réteg végzi el az adatbázis kapcsolat létesítését és az onnan kapott adatokat adja vissza a Service rétegnek.

Service réteg: A service rétegen határozzuk meg az üzleti logikát és validáljuk, hogy a felhasználótól érkező adat minden elvárásnak megfelelő-e, illetve a modell réteggel való kommunikációt kezeli.

Controller réteg: A controller réteg felelős a HTTP kérések fogadásáért és a megfelelő service réteg meghívásáért, valamint a válaszok visszaküldéséért.

Config réteg: A config rétegen kaptak helyet például a Token.java osztály, ami felelős a token létrehozásáért, dekódolásáért és a benne tárolt adatok visszaadásáért.

DTO réteg: A DTO rétegen helyezkednek el azok az osztályok, amik nem hozhatóak létre egy entity-ként, viszont segít a frontend-ről érkező adathalmaz egyszerű feldolgozását.

A projekt Vanilla Java-ban van írva, tehát nincs szükség külső keretrendszerekre. Az egyszerűsített megközelítés lehetővé teszi a fejlesztők számára a teljes kontrollt a kód felett és minimalizálja a felesleges függőségeket.

Az alkalmazás egy RESTful API-t kínál a frontend számára. A kérések és válaszok JSON formátumban történnek, és a következő HTTP módszereket támogatja:

GET: Az adatok lekérése a szerverről.

POST: Új adatok hozzáadása a szerverhez.

DELETE: Adatok törlése a szerverről.

Adatbázis:

A fenti képen látható a jelenleg használt relációs adatbázis struktúra. A képen látható, hogy külön táblákban tároljuk a “general” és “publisher” felhasználók adatait, azonban azok az adatok, amik mindkét felhasználónál jelen vannak, a “user” táblában kaptak helyet.

Tároljuk azokat a kategóriákat, amiket a felhasználó az első bejelentkezésnél ki tud választani, hogy melyikek érdeklik őt. Illetve külön táblákban mentjük azokat az adatokat is, ha egy felhasználó bejelent egy könyvet vagy éppen felhasználót. Ennek a funkció az “admin” felület megírásával lesz értelme, amikor kezelni is lehet majd ezeket a jelentéseket.

Lehetőség van a felhasználók és a könyvek értékelésére is, ezeknek is látható, hogy külön táblát szántunk. Ez alapján egy adott írónak, kiadónak vagy könyvnek jobb megítélése lehet az olvasók körében.

A fájloknak csak az elérési útját tároljuk az adatbázisban, és a későbbiekben ezzel hivatkozunk rájuk. Vásárlás esetén például mentésre kerül a számlának az elérési útja is, illetve minden fontos adatot tárolunk róla, hogy a későbbiekben reklamáció esetén ez ne okozzon gondot, illetve a felhasználó kérésére bármikor ki tudjuk neki adni azt.

Az adatbázisban külön táblát szántunk az Általános Szerződési Feltételek dokumentumainak a tárolására. Így amikor az változik, ide egy új rekordként elmentjük a jelenlegi verziót is, ezáltal szépen visszakövethető, hogy mikortól meddig volt érvényes az adott ÁSZF.

Illetve amire még különös figyelmet fordítottunk, az a mentési lehetőség, ahol azt tároljuk, hogy egy felhasználó milyen könyveket mentett el magának.

Mindenhol tárolásra kerül a rekord létrejöttének a pontos ideje is, ami egyes helyeken visszakövethető az oldalon, más esetekben felénk lehet referencia.