**Követelmény Dokumentáció - Használt Hardver Árusító Platform**

**Szakdolgozat Téma: Használt hardver árusító webalkalmazás fejlesztése**

**1. Bevezetés**

A szakdolgozat célja egy online platform létrehozása, amely lehetőséget biztosít használt számítástechnikai hardverek értékesítésére és vásárlására. Az alkalmazás hasonlóan fog működni, mint a már ismert **hardverapro.hu**, de egyedi funkciókkal bővítve, felhasználóbarát felülettel és modern technológiai megoldásokkal.

Ez a dokumentum specifikációkat, technikai követelményeket és egy Proof of Concept (POC) tervet tartalmaz a projekt megvalósításához.

**2. Funkcionális követelmények**

**2.1 Regisztráció és bejelentkezés**

* **Leírás**: A felhasználók regisztrálhatnak az oldalon egy e-mail cím és jelszó megadásával.
* **Követelmények**:
  + Az e-mail alapú regisztrációhoz szükséges egy validációs folyamat.
  + Jelszó elfelejtése esetén e-mailben történő visszaállítási lehetőség.

**2.2 Termékek böngészése**

* **Leírás**: A felhasználók böngészhetnek a meghirdetett termékek között.
* **Követelmények**:
  + Kategória szerinti szűrési lehetőség (pl. CPU, GPU, alaplap, stb.).
  + Ár szerinti szűrés és rendezés.
  + Termék állapot szerinti szűrés (új, használt, felújított).
  + Termékek képekkel és részletes leírással jelenjenek meg.

**2.3 Termékek feltöltése**

* **Leírás**: A regisztrált felhasználók feltölthetnek eladásra szánt termékeket.
* **Követelmények**:
  + Kötelező mezők: Termék neve, állapota, ára, képek feltöltése, leírás.
  + Kategória kiválasztása.
  + A termékfeltöltés moderálására és jóváhagyására van szükség a duplikációk és hibás hirdetések elkerülése érdekében.

**2.4 Értesítések és üzenetek**

* **Leírás**: A rendszer értesítéseket küldhet eladóknak és vásárlóknak az oldalon történt interakciókról.
* **Követelmények**:
  + Üzenetküldő felület a vevők és eladók közötti kommunikációhoz.
  + Értesítések új üzenetekről és ajánlatokról.
  + Értesítések a termékek státuszának változásáról (pl. „termék eladva”).

**2.5 Adminisztrációs felület**

* **Leírás**: Az adminisztrátorok számára elérhető felület a felhasználók és hirdetések kezelésére.
* **Követelmények**:
  + Hirdetések felülvizsgálata és moderálása.
  + Felhasználói fiókok zárolása vagy inaktiválása szabályszegés esetén.
  + Statisztikai adatok megtekintése (pl. aktív hirdetések száma, eladott termékek száma).

**3. Nem funkcionális követelmények**

**3.1 Teljesítmény**

* Az oldalnak gyors válaszidőt kell biztosítania, a termékek listázása és szűrése kevesebb, mint 1 másodperc alatt történjen.
* Az adatbázis 100,000 termék kezelésére legyen optimalizálva.

**3.2 Biztonság**

* HTTPS titkosított kapcsolat kötelező.
* Felhasználói adatok (beleértve a jelszavakat) titkosítása szükséges.
* SQL Injection, XSS, és más ismert támadások ellen védettség.

**3.3 Skálázhatóság**

* A rendszernek skálázhatónak kell lennie, hogy nagy felhasználószámot és adatbázist is kezelni tudjon.
* AWS vagy más felhő alapú szolgáltatás használata a hosztoláshoz.

**4. Technológiai specifikáció**

**4.1 Frontend technológia**

* **React.js**: A felhasználói felülethez egy modern JavaScript könyvtár használata, amely gyors és dinamikus élményt biztosít.
* **CSS keretrendszer**: Bootstrap vagy Tailwind CSS a reszponzív és gyorsan fejleszthető felület kialakításához.
* **Axios**: HTTP kérések kezelésére.

**4.2 Backend technológia**

* **Node.js**: A backend szolgáltatásokhoz és API-khoz gyors és skálázható szerveroldali környezet.
* **Express.js**: Webszerver keretrendszer a REST API-k kialakításához.
* **MongoDB**: NoSQL adatbázis a rugalmasság és gyors lekérdezések érdekében.

**4.3 Felhőszolgáltatások**

* **AWS S3**: A termékek képeinek tárolására.
* **AWS Lambda**: A skálázható és költséghatékony működéshez.
* **AWS RDS**: Relációs adatbáziskezelő rendszerként, ha szükséges SQL alapú adattárolás.

**4.4 Felhasználói hitelesítés és jogosultságkezelés**

* **JWT (JSON Web Tokens)**: A felhasználói hitelesítéshez egy biztonságos és gyors token alapú megoldás.
* **OAuth 2.0**: Harmadik fél általi bejelentkezés (Google, Facebook).

**5. Proof of Concept (PoC)**

A PoC célja, hogy egy minimálisan működő prototípust készítsünk a platform legfontosabb funkcióival. A következő funkciókat tartalmazza:

1. **Regisztráció és bejelentkezés**: E-mail alapú regisztráció, Google OAuth integrációval.
2. **Termék böngészés**: Alap termékkeresési és szűrési funkciók.
3. **Termékfeltöltés**: Egy egyszerű felület, ahol a felhasználók feltölthetnek egy terméket kép és leírás megadásával.
4. **Adminisztráció**: Moderálási eszközök hirdetések és felhasználói fiókok kezelésére.

**5.1 Példa**

**Frontend:**

* A felhasználók a főoldalon kereshetnek termékek között, láthatják a legnépszerűbb kategóriákat (pl. videokártyák, alaplapok).
* Egy felhasználó regisztrálhat, bejelentkezhet, majd feltölthet egy terméket a **„Termék feltöltése”** oldalon, ahol egy egyszerű űrlapot kell kitölteni.

**Backend:**

* Az API útvonalak segítségével biztosítja a termékek adatainak feltöltését, lekérdezését és a felhasználói jogosultságok ellenőrzését.

**Adatbázis felépítés:**

* **Users**: Felhasználói adatok tárolása (ID, név, e-mail, jelszó, stb.).
* **Products**: Termékek adatainak tárolása (ID, név, ár, állapot, kategória, feltöltő azonosító).

**6. Tesztelés és hibakezelés**

* **Unit tesztek**: Backend funkciók és API végpontok tesztelése.
* **Integrációs tesztek**: Felhasználói műveletek teljes folyamatának tesztelése (pl. termék feltöltése, majd annak megtekintése).
* **Load tesztelés**: A rendszer teljesítményének vizsgálata nagy terhelés alatt.

**7. Tesztelés és hibakezelés**

**7.1 Unit tesztek**

* **Cél**: A rendszer egyes komponenseinek tesztelése külön-külön annak érdekében, hogy megbizonyosodjunk arról, hogy a funkcionalitás helyesen működik. A tesztekhez olyan eszközök használhatók, mint a **Jest** vagy a **Mocha**.
* **Példa**:
  + Tesztelni kell a felhasználói bejelentkezési folyamatot, hogy a helyes e-mail és jelszó kombináció sikeres hitelesítést eredményezzen.
  + A termékfeltöltési funkció unit tesztje, amely biztosítja, hogy a kötelező mezők ki legyenek töltve, és az adatok megfelelően elérjék az adatbázist.

**7.2 Integrációs tesztek**

* **Cél**: Biztosítani, hogy a rendszer különböző részei együttműködjenek a várt módon. Például a frontend és a backend közötti adatáramlás zökkenőmentes legyen.
* **Példa**:
  + A felhasználó regisztrálása, bejelentkezése és egy termék sikeres feltöltése.
  + A termék szűrésének tesztelése kategóriák és ár alapján, hogy meggyőződjünk a szűrési algoritmus helyességéről.

**7.3 Load tesztek**

* **Cél**: A rendszer terhelés alatti teljesítményének mérése és annak biztosítása, hogy az alkalmazás nagy felhasználószám esetén is megfelelően működik.
* **Eszközök**: **Apache JMeter**, **Locust**.
* **Példa**:
  + 1000 egyidejű felhasználó termékkeresése, illetve böngészése, és annak tesztelése, hogy az oldal továbbra is 1 másodpercen belül válaszol.
  + 500 egyidejű termékfeltöltés terheléses tesztje, hogy az adatbázis megfelelően skálázódik.

**7.4 Hibakezelés**

* **Cél**: A hibák hatékony kezelése és naplózása annak érdekében, hogy a felhasználók felé megfelelő hibaüzenetek jelenjenek meg, és a fejlesztők gyorsan azonosíthassák a problémákat.
* **Eszközök**: **Sentry** vagy más hibanaplózó eszköz integrálása.
* **Példa**:
  + Hibás adatbevitel esetén a rendszer azonnal figyelmeztető üzenetet küld a felhasználónak (pl. hiányzó mező vagy érvénytelen adat).
  + Az adatbázis-kapcsolati hibák vagy egyéb szerveroldali problémák esetén a rendszer egy általános hibaüzenetet küld a felhasználónak (pl. „Jelenleg nem elérhető, próbálja újra később.”), miközben a háttérben a hibanaplózási eszköz részletes adatokat gyűjt a hibáról.

**8. Felhasználói élmény (UX) és felület (UI)**

**8.1 Felhasználói felület tervezése**

* **Cél**: Az alkalmazásnak egy egyszerű, könnyen navigálható és vizuálisan vonzó felületet kell biztosítania.
* **Elvárások**:
  + Tiszta, modern dizájn.
  + A főoldalon jól elkülönített szűrési lehetőségek és gyors elérhetőség a különböző termékkategóriákhoz.
  + A felhasználói profil könnyen kezelhető, ahol az eladó egyszerűen nyomon követheti a feltöltött termékeket és azok státuszát.

**8.2 Reszponzív dizájn**

* **Cél**: Az oldalnak különböző képernyőméreteken (mobil, tablet, asztali gép) is jól használhatónak kell lennie.
* **Technológia**: **Bootstrap** vagy **Tailwind CSS** alkalmazása a reszponzív dizájn kialakításához.
* **Példa**:
  + Mobilon a terméklisták és szűrési lehetőségek harmonikusan illeszkednek a kisebb kijelzőhöz, míg az asztali nézetben több adat látható egyszerre.

**8.3 Felhasználói visszajelzések kezelése**

* **Cél**: A felhasználók által visszajelzést adhatnak az oldalon szerzett tapasztalataikról, amit a rendszer a folyamatos fejlesztéshez felhasznál.
* **Megoldások**:
  + Felhasználói értékelési rendszer, ahol a vevők és eladók értékelhetik egymást.
  + Egy „Kapcsolat” menüpont, ahol a felhasználók visszajelzéseket küldhetnek az oldallal kapcsolatban.

**9. Telepítés és üzembe helyezés**

**9.1 Fejlesztői környezet**

* A platform fejlesztése során egy fejlesztői környezet kialakítása szükséges, amely tartalmazza a szükséges eszközöket a kód írásához, teszteléséhez és telepítéséhez.
* **Technológiák**:
  + **Git** a verziókezeléshez.
  + **Docker** konténerizált fejlesztési környezet létrehozására.

**9.2 Telepítési lépések**

* **Felhőkörnyezet**: A platform hosztolásához felhő alapú infrastruktúra használata javasolt (pl. AWS, Azure).
* **CI/CD (folyamatos integráció és telepítés)**:
  + **Jenkins** vagy **GitHub Actions** integrálása a kód automatikus telepítéséhez és teszteléséhez.
  + Automatizált tesztelési folyamatok futtatása minden kódváltoztatás után.

**9.3 Üzemeltetés és karbantartás**

* A rendszer folyamatos monitorozása szükséges annak érdekében, hogy az esetleges problémákat gyorsan lehessen orvosolni.
* **Monitoring eszközök**: **AWS CloudWatch** vagy **Grafana** a szerverek és alkalmazás teljesítményének figyeléséhez.
* **Biztonsági frissítések**: A rendszer folyamatos frissítése az új biztonsági javításokkal és funkciókkal.

**10. Projekt menedzsment**

**10.1 Fejlesztési szakaszok**

A fejlesztési folyamat három fő szakaszra osztható:

1. **Tervezés**: Ebben a szakaszban a követelmények pontos meghatározása és az architektúra tervezése történik meg.
   * Szükséges idő: 2 hét.
2. **Fejlesztés**: A platform funkcióinak lépésről lépésre történő fejlesztése, kezdve az alapvető funkcióktól (regisztráció, termékfeltöltés) a bonyolultabb rendszerekig (értesítések, adminisztráció).
   * Szükséges idő: 8-10 hét.
3. **Tesztelés és üzembe helyezés**: A kód teljes körű tesztelése és telepítése az éles környezetbe.
   * Szükséges idő: 2-3 hét.

**10.2 Projekt ütemezése**

A projekt becsült időtartama 12-14 hét, figyelembe véve a tervezést, fejlesztést, tesztelést és üzemeltetést.

**10.3 Erőforrások**

* **Fejlesztői csapat**: 1-2 frontend fejlesztő, 1 backend fejlesztő, 1 DevOps mérnök.
* **Technikai eszközök**: Laptopok, fejlesztői szoftverek (pl. Visual Studio Code, Docker).

**11. Összefoglalás**

A szakdolgozat keretében létrehozandó használt hardver árusító platform egy olyan webalkalmazás, amely modern technológiai megoldásokat használ a felhasználók számára gyors, biztonságos és kényelmes hirdetési lehetőséget biztosít. Az oldal biztosítja a termékek gyors feltöltését, keresését, és vásárlását, miközben lehetőséget ad a felhasználók értékelésére és a kommunikációra. A technikai specifikációk részletesen meghatározzák a felhasznált eszközöket és rendszereket, amelyek biztosítják az alkalmazás skálázhatóságát, teljesítményét és biztonságát. A Proof of Concept (POC) segít abban, hogy az alapvető funkciók prototípusaként bizonyítsuk a projekt megvalósíthatóságát, és a későbbi fejlesztési szakaszok irányát. Az alkalmazás fő célja, hogy a felhasználók gyorsan és hatékonyan tudjanak használt számítástechnikai hardvereket hirdetni, keresni és vásárolni egy felhasználóbarát felületen.

A projekt szempontjából kiemelten fontos a biztonságos adatkezelés, a teljesítmény optimalizálása és a skálázhatóság biztosítása. Az alkalmazás hosztolása felhőalapú megoldásokra épül, amelyek biztosítják a rugalmasságot és az infrastruktúra költséghatékonyságát. A folyamatos tesztelés és hibakezelés kulcsfontosságú a stabil és megbízható működés érdekében.

A szakdolgozat zárásaként a platform fejlesztéséhez kapcsolódó tapasztalatokat és kihívásokat, valamint a jövőbeni fejlesztési lehetőségeket kell összegyűjteni. További javasolt fejlesztések lehetnek az új funkciók bevezetése, például aukciós rendszer vagy bővített keresési algoritmusok, illetve az integráció különböző fizetési rendszerekkel, hogy a felhasználók számára még kényelmesebbé váljon a vásárlás és eladás.

Ez a követelménydokumentáció részletes áttekintést nyújtott a használt hardver árusító platform tervezéséről és fejlesztéséről. A platform modern technológiai megoldásokkal, átgondolt architektúrával és reszponzív felhasználói élménnyel kíván megfelelni a célcsoport igényeinek, biztosítva a könnyű kezelhetőséget és a biztonságos felhasználást.

**12. Jövőbeli fejlesztési lehetőségek**

**12.1 Mobilalkalmazás fejlesztése**

Az oldalon elérhető funkciók mobilalkalmazás formájában történő megvalósítása, amely Android és iOS platformon is elérhető lenne. Az alkalmazás push értesítéseket biztosíthat a felhasználók számára, ezzel fokozva a felhasználói élményt.

**12.2 Aukciós rendszer bevezetése**

Az eladók számára lehetőséget biztosítani arra, hogy aukciós formában értékesítsék termékeiket, ahol a felhasználók licitálhatnak a termékekre.

**12.3 Fizetési rendszerek integrálása**

Olyan online fizetési rendszerek, mint például a PayPal vagy Stripe, integrálása, amelyek segítségével a vásárlók közvetlenül a platformon keresztül fizethetnek az eladóknak.

**12.4 Gépi tanulás alapú ajánlórendszer**

Egy gépi tanulás alapú algoritmus kidolgozása, amely a felhasználói viselkedés alapján személyre szabott termékajánlásokat kínál, növelve ezzel az eladások esélyét és a felhasználói elégedettséget.