Java alkalmazások tárgy beadandó

Dokumentáció

<https://github.com/Zsotti44/Forma1-sb>

|  |  |
| --- | --- |
| **Készítették** | **Feladatkör** |
| Katona Zsolt Krisztián | Fejlesztés |
| Sándor Adrián | Fejlesztés |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| Dokumentum azonosító: | JAVAA-D-1 |
| Verzió: | 0.01 |
| Dátum: | 2024.10.26 |

|  |
| --- |
| **Lektorálta** |
| - |

**©** Minden jog fenntartva!

Tartalomjegyzék

[1. Bevezetés 4](#_Toc180840553)

[1.1. Dokumentum célja 4](#_Toc180840554)

[1.2. Választott projekt leírása 4](#_Toc180840555)

[1.3. Feladatok 4](#_Toc180840556)

[2. Fejlesztési környezet 5](#_Toc180840557)

[2.1. A web szerver: Apache Tomcat és XAMPP 5](#_Toc180840558)

[2.2. Verziókövetés: GitHub 5](#_Toc180840559)

[2.3. Használt technológiák 5](#_Toc180840560)

[3. Megvalósítás 6](#_Toc180840561)

[3.1. Adatbázis 6](#_Toc180840562)

[**3.1.1.** **Adatbázis táblák** 6](#_Toc180840563)

[3.1.1.1. gp tábla: 6](#_Toc180840564)

[3.1.1.2. eredmeny tábla: 7](#_Toc180840565)

[3.1.1.3. pilota tábla: 7](#_Toc180840566)

[3.1.1.4. uzenet tábla: 7](#_Toc180840567)

[3.1.1.5. felhasznalo tábla: 8](#_Toc180840568)

# Bevezetés

## Dokumentum célja

Jelen leírás a(z) „L-LA01 Java alkalmazások [L-K-GINFBAL-JAVAALKA-1-LA01]” elnevezésű kurzus, Sándor Adrián és Katona Zsolt által fejlesztés alatt álló számítástechnikai projekt, weboldal fejlesztői környezetét, koncepcióját, design elemeinek tervezett megvalósítását és fejlesztési tevékenységeit ismerteti.

* 1. Választott projekt leírása

A választott adatbázis a Formula 1-es történelmi versenyek eredményeit követi, kiemelve a pilóták teljesítményét, a csapatok részvételét, valamint az autók és a motorok technikai specifikációit különböző versenyeken.

## Feladatok

* + 1. Az első oldalon mutassa be a céget egy látványos weboldalon
    2. Legyen Regisztráció, Bejelentkezési lehetőség
       1. A „Belépés” menüpont akkor látható, ha nincs bejelentkezve a felhasználó.
       2. A „Kilépés” menüpont akkor látható, ha be van jelentkezve a felhasználó.
       3. A rendszer fejlécen jelenítse meg a bejelentkezett felhasználót, ha be van lépve
    3. Legalább 3 felhasználói szerepet különböztessen meg
       1. Admin, User, Látogató
       2. A menüpontok megjelenése és az oldalak elérhetősége változik attól függően, hogy melyik felhasználó használja az oldalt. (pl. admin oldal)
    4. Legyen egy oldal, ahol a választott adatbázisból jelenít meg adatokat. Ehhez 3 tábla adatait használja fel az adatbázisból.
    5. Az egyik oldalon legyen egy kapcsolat űrlap, amelynek segítségével üzenetet lehet küldeni az oldal tulajdonosa számára. Ellenőrizze megfelelően az űrlap helyes kitöltését szerver oldali validációval is. Az elküldött Űrlap adatokat mentse le az adatbázisba.
    6. Tegye lehetővé megtekinteni egy hatodik oldalon táblázatban az előző pontban elküldött üzeneteket az adatbázisból fordított időrend szerint (a legfrissebb legyen elől). Írja minden üzenethez a küldés idejét és az üzenetküldő nevét. Ha nem bejelentkezett felhasználó írta, akkor: "Vendég".
    7. Valósítson meg az alkalmazásban egy RESTful API-t.Tesztelje az API funkcióit cURL-el és Postman-al is. Mindkét tesztről tegyen képernyőképeket a dokumentációba.
    8. Használják a GitHub (github.com) verziókövető rendszert.
    9. A GitHub-on a projektmunka módszert alkalmazzák: látszódjék, hogy a csoport tagjai melyik részt készítették el és kb. fele-fele arányban járuljanak hozzá a projekthez. Ne csak a kész alkalmazást töltsék fel egy lépésben, hanem a részállapotokat is még legalább 5 lépésben személyenként.

# Fejlesztési környezet

* 1. A web szerver: Apache Tomcat és XAMPP  
     A XAMPP egy ingyenes és nyílt forrású cross-platform web szerver megoldáscsomag, amelyet kifejezetten fejlesztési célokra terveztek. A XAMPP az Apache Tomcat web szervert, MariaDB adatbázis-kezelőt. A XAMPP segítségével a fejlesztők gyorsan létrehozhatnak egy helyi fejlesztési környezetet a számítógépükön, amely lehetővé teszi számukra a weboldalak és webalkalmazások fejlesztését és tesztelését anélkül, hogy szükség lenne egy élő szerverre. Ez különösen előnyös a fejlesztés korai szakaszaiban, amikor gyakoriak a változtatások, és fontos a gyors iteráció.  
     Az Apache Tomcat egy tisztán Java nyelven készült webszerver, amely implementálja a Sun-féle Java Servlet és a JavaServer Pages specifikációkat. E specifikációkat támogató webszervereket szokás a servlet container, a servlet engine illetve a web engine összetételekkel is illetni.
  2. Verziókövetés: GitHub  
     A GitHub, mint vezető szoftverfejlesztési platform, lehetővé teszi a csapat számára, hogy hatékonyan kezelje a kódbázist, nyomon követhesse a változásokat, és elősegítse a kollaboratív munkát azáltal, hogy egy központosított helyen tárolja a projekt fájljait és dokumentációját. Az integrált verziókezelési rendszer előnyei közé tartozik a fejlesztési folyamatok nagyfokú automatizálása, a visszaállíthatóság, valamint a különböző fejlesztési ágak (branches) kezelése, ami lehetővé teszi a csapat számára, hogy párhuzamosan dolgozzon különböző funkciókon. A GitHub használatával a fejlesztőcsapat kihasználhatja a kódmegosztás és a közösségi hozzájárulás előnyeit is, így gyorsítva a fejlesztési ciklusokat és javítva a projekt átláthatóságát és nyomon követhetőségét.
  3. Használt technológiák  
     A csapat a fejlesztés során számos modern és bevált technológiát alkalmaz, hogy biztosítsák a webalkalmazásuk gyors, biztonságos és felhasználóbarát működését:
     1. **Java**: A Java általános célú, objektumorientált programozási nyelv mely kiválóan használható akár bonyolultabb, nagy elérésű alkalmazások akár kisebb weboldalak szerveroldali megoldásaként.
     2. **Spring framework**: A Spring egy nyílt forráskódú, IoC-t megvalósító Java alkalmazás keretrendszer.
     3. **Spring Boot framework**: A Spring Boot a Spring keretrendszer kiterjesztése, amely egyszerűsíti az új Spring alkalmazások kezdeti beállítását és fejlesztését.
     4. **HTML** (HyperText Markup Language): Az internet alapját képező leíró nyelv, amely a weboldalak struktúráját adja meg. Az HTML5 a legújabb verziója, amely további szemantikai elemeket és jobb multimedia támogatást kínál, ezáltal elősegíti a reszponzív webdesign kialakítását.
     5. **Thymeleaf**: A Thymeleaf egy Java XML/XHTML/HTML5 sablonmotor, amely webes (servlet-alapú) és nem webes környezetben egyaránt működik. Alkalmas az XHTML/HTML5 kiszolgálására az MVC-alapú webes alkalmazások nézeti rétegében, de bármilyen XML-fájlt képes feldolgozni offline környezetben is. Teljes Spring Framework integrációt biztosít
     6. Bootstrap: Egy ingyenes és nyílt forrású front-end keretrendszer, amely elősegíti a gyors és egyszerű reszponzív webdesign kialakítását. Készen álló HTML és CSS sablonokat, valamint JavaScript komponenseket kínál, amelyekkel könnyedén lehet készíteni felhasználóbarát és esztétikailag vonzó weboldalakat.
     7. Reszponzív webdesign: Az a tervezési és fejlesztési módszertan, amelynek célja, hogy a weboldalak minden eszközön - asztali számítógéptől kezdve a mobiltelefonokig - jól nézzenek ki és jól működjenek. A Bootstrap keretrendszer kiválóan támogatja a reszponzív webdesign kialakítását.

# Megvalósítás

## Adatbázis

Az Adatbázis a weboldalunk alapvető és kritikus eleme, amely két fő célra használatos. Elsődlegesen szolgál a felhasználók hitelesítésére, ugyanis itt tároljuk azokat az adatokat, amelyek az azonosításhoz és az egyedi felhasználói profilok létrehozásához szükségesek, mint a felhasználónevek, jelszavak, jogosultságok és id. Másrészt az adatbázis tartalmazza a feladat eredetileg meghatározott tábláit és az importálást követően a letöltött adatokat.

A felhasználóknak lehetőségük van üzenetet küldeni az üzemeltetőnek. Ezek az üzenetek szintén az adatbázisban vannak tárolva. Egy-egy rekordon, a felhasználó azonosítójával megállapítható ki küldte az üzenetet.

Miután bejelentkezés nélkül is van lehetőség üzenetküldésre, ilyenkor pedig nincs azonosítható felhasználó, null értékkel kerülnek rögzítésre az üzenetek, megjelenítéskor ezek a rekordok a „Vendég” megnevezést kapják.

### **Adatbázis táblák**

* + - 1. gp tábla:

A gp tábla a nagydíjak (Grand Prix) adatait tárolja, amelyek a versenysorozat különböző helyszínein zajlottak. Minden rekord egy nagydíjat reprezentál, amely tartalmazza a verseny dátumát, nevét és helyszínét. Ez a tábla tehát a versenyek alapvető információit rögzíti, amellyel összekapcsolhatók a versenyzők eredményei az eredmeny táblából.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

* + - 1. eredmeny tábla:

Az eredmeny tábla a motorsport versenyek eredményeit tárolja. Minden sor egy adott versenyző (pilóta) egy konkrét versenyen elért eredményét tartalmazza. A tábla különböző oszlopokban tárolja az eredménnyel kapcsolatos adatokat, beleértve a verseny dátumát, ami egy idegenkulcs a gp táblára, továbbá a pilóta azonosítóját, ami szintén egy idegen kulcs, a pilota táblára, a helyezést, a verseny során előforduló hibákat, valamint az autó típusát és motorját.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

* + - 1. pilota tábla:

A pilota tábla a versenyzők, azaz a pilóták adatait tárolja. Minden sor egy versenyzőt reprezentál, beleértve a nevüket, nemüket, születési dátumukat és nemzetiségüket. Ez a tábla a motorsport versenyzői adatainak tárolására szolgál, és lehetővé teszi az egyes pilóták azonosítását az eredményeik alapján.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

* + - 1. uzenet tábla:

Az **uzenet** tábla egy üzenetek tárolására szolgáló tábla, amelyben a felhasználók által létrehozott üzenetek adatai kerülnek mentésre. Minden rekord egy adott üzenetet tartalmaz, amit egy felhasználó küldött, beleértve az üzenet szövegét, a felhasználó azonosítóját, valamint a rögzítés időpontját.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

* + - 1. felhasznalo tábla:

A felhasznalo tábla a felhasználók adatait tárolja, akik a rendszerbe beléphetnek. Minden rekord egy felhasználót reprezentál, amely tartalmazza a felhasználó nevét, jelszavát és jogosultsági szintjét. Ez a tábla kulcsfontosságú a felhasználói hitelesítéshez és jogosultságkezeléshez a rendszerben.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás