SZAKDOLGOZAT

|  |
| --- |
| Váci Szakképzési Centrum  Boronkay György Műszaki Technikuma és Gimnáziuma |
| **VIZSGAREMEK**  Arcalite |
| Konzulens: Készítők:  Gyombolainé Cserny Zsuzsanna Bakó Borka  Bali István Gábor  Katona Bálint  Vác  2025 |

Hallgatói nyilatkozat

Alulírottak, ezúton kijelentjük, hogy a szakdolgozat saját, önálló munkánk, és korábban még sehol nem került publikálásra. Szakdolgozatunk a Váci Szakképzési Centrum Boronkay György Műszaki Technikum és Gimnázium Szoftverfejlesztő és tesztelő technikus képzésén készítettük. Tudomásul vesszük, hogy szakdolgozatunkat a Váci Szakképzési Centrum Boronkay György Műszaki Technikum és Gimnázium tárolja.

Bakó Borka Bali István Gábor Katona Bálint

Konzultációs lap

Vizsgázók neve:

* Bakó Borka
* Bali István Gábor
* Katona Bálint

Szakdolgozat címe: Arcalite

Program által nyújtott szolgáltatások:

* Platformer játék
* A játékbeli haladás nyomonkövetése a weboldalon
* Tudástár a játékban lévő ellenfelekhez és tárgyakhoz
* E-mailek (visszaigazoló, jelszó módosítás)

|  |  |  |
| --- | --- | --- |
| **Sorszám** | **A konzultáció időpontja** | **Konzulens aláírása** |
| 1. | 2024. október 11. |  |
| 2. | 2024. november 15. |  |
| 3. | 2024. december 13. |  |
| 4. | 2025. január 17. |  |
| 5. | 2025. február 14. |  |
| 6. | 2025. március 14. |  |

A szakdolgozat beadható: A szakdolgozatot átvettem:

Vác, 2025. Vác, 2025.

Konzulens A szakképzést folytató

intézmény felelőse

# Tartalomjegyzék

[1. Tartalomjegyzék 5](#_Toc193450397)

[2. Témaválasztás 7](#_Toc193450398)

[3. Fejlesztői dokumentáció 8](#_Toc193450399)

[3.1. Fejlesztői környezet 8](#_Toc193450400)

[3.1.1. Visual Studio Code 8](#_Toc193450401)

[3.1.2. XAMPP 8](#_Toc193450402)

[3.1.3. Visual Studio 8](#_Toc193450403)

[3.1.4. GODOT 8](#_Toc193450404)

[3.2. Használt technológiák, nyelvek 8](#_Toc193450405)

[3.2.1. HTML 8](#_Toc193450406)

[3.2.2. CSS 9](#_Toc193450407)

[3.2.3. JavaScript 9](#_Toc193450408)

[3.2.4. jQuery 9](#_Toc193450409)

[3.2.5. JSON 9](#_Toc193450410)

[3.2.6. Bootstrap 9](#_Toc193450411)

[3.2.7. PHP 10](#_Toc193450412)

[3.2.8. HTTP, REST API 10](#_Toc193450413)

[3.2.9. MySQL 10](#_Toc193450414)

[3.2.10. C# 10](#_Toc193450415)

[3.2.11. Git, GitHub 10](#_Toc193450416)

[3.2.12. ChatGPT 11](#_Toc193450417)

[3.3. Adatszerkezet 12](#_Toc193450418)

[3.3.1. Alapadatok 12](#_Toc193450419)

[3.3.2. Kapcsolati ábra 12](#_Toc193450420)

[3.3.3. Táblák 13](#_Toc193450421)

[3.3.3.1. A profile tábla 13](#_Toc193450422)

[3.3.3.2. Az achievement tábla 13](#_Toc193450423)

[3.3.3.3. A proach tábla 14](#_Toc193450424)

[3.3.3.4. A player tábla 14](#_Toc193450425)

[3.3.3.5. Az avatar tábla 15](#_Toc193450426)

[3.3.3.6. A level tábla 15](#_Toc193450427)

[3.3.3.7. Az enemy tábla 16](#_Toc193450428)

[3.3.3.8. Az enemplay tábla 16](#_Toc193450429)

[3.3.3.9. Az item tábla 17](#_Toc193450430)

[3.3.3.10. Az itemplay tábla 17](#_Toc193450431)

[3.3.3.11. A lang tábla 18](#_Toc193450432)

[3.3.3.12. Az achdesc tábla 18](#_Toc193450433)

[3.3.3.13. Az avatardesc tábla 19](#_Toc193450434)

[3.3.3.14. Az enemydesc tábla 19](#_Toc193450435)

[3.3.3.15. Az itemdesc tábla 20](#_Toc193450436)

[3.3.3.16. A saves tábla 20](#_Toc193450437)

[3.4. A weboldal felépítése 21](#_Toc193450438)

[3.5. Az alkalmazás felépítése 21](#_Toc193450439)

[3.6. Tesztelés 21](#_Toc193450440)

[3.7. Továbbfejlesztési lehetőségek 21](#_Toc193450441)

[4. Felhasználói dokumentáció 22](#_Toc193450442)

[4.1. A program rövid ismertetése 22](#_Toc193450443)

[4.2. A program feltelepítése 22](#_Toc193450444)

[4.2.1. A játék telepítése 22](#_Toc193450445)

[4.2.2. A weboldal üzembe helyezése 22](#_Toc193450446)

[4.2.3. Az adatbázis felállítása 22](#_Toc193450447)

[4.3. Használati útmutató 22](#_Toc193450448)

[4.3.1. A játék irányítása 22](#_Toc193450449)

[4.3.2. A weboldal használata 22](#_Toc193450450)

[5. Mellékletek 23](#_Toc193450451)

# Témaválasztás

Ide egy szép szöveg, hogy miért csináljuk ezt <3

# Fejlesztői dokumentáció

## Fejlesztői környezet

### Visual Studio Code

A Visual Studio Code (VSC) egy nyílt forráskódú kódszerkesztő alkalmazás. Nagy mértékben testre szabható, illetve rengeteg kiegészítő tartalom készült hozzá, melyekkel könnyen fejleszthetünk a legtöbb programozási nyelven. Mi a weboldal HTML vázához, formázásához, a funkcionalitást biztosító JavaScript szkriptekhez, illetve az adatbázissal kommunikáló PHP RestAPI fejlesztéséhez használjuk.

### XAMPP

A XAMPP egy nyílt forráskódú és platformfüggetlen webszerver-szoftver csomag. A csomag fő elemei – amelyek miatt mi is használjuk – az Apache szerver, PHP támogatás, a MariaDB adatbázis-motor, illetve a phpMyAdmin adatbázis-kezelőfelület.

### Visual Studio

A Visual Studio egy fejlesztői kódszerkesztő, amit kód szerkesztésére, debug-olására, de akár alkalmazásunk publikálására is használhatunk. Az IDE a standard kódszerkesztésen és debug-oláson felül tartalmaz fordítókat, kódkiegészítő, grafikus tervező, és még sok egyéb hasznos eszközzel, melyek elősegítik a fejlesztési élményünket. Ebben a projektben a játék kódjának megírásához használtuk.

### GODOT

A Godot egy játékfejlesztő motor, mellyel egy egységes felületen készíthetünk akár 2D-s, akár 3D-s platformfüggetlen játékokat. A motor egy mindent átfogó eszközcsomagot kínál, hogy a fejlesztők a játék fejlesztésére fókuszálhassanak, és ne kelljen már megoldott problémákat újból megoldani. A program teljesen ingyenes, és nyílt forráskódú, többnyire a közösség által fejlesztett.

## Használt technológiák, nyelvek

### HTML

A HTML (HyperText Markup Language) egy struktúrált leírónyelv, a weboldalak szerkezetének kialakításához használt szabvány. Általában együtt használatos a CSS és JavaScript nyelvekkel, mely három a webfejlesztés alapjaként szolgál. Míg a HTML a struktúra kialakítására használják, a CSS-t a formázásra, a JavaScript-et pedig a funkcionalitás kialakítására használják.

### CSS

A CSS (Cascading Style Sheets) egy stíluslapnyelv, mellyel a weboldalak megjelenítését és formázását alakíthatjuk ki. Lehetővé teszi, hogy a fejlesztők a formázást és a struktúrát elválasszák egymástól, továbbá, hogy a weboldalak különféle környezetekben is egységesen jelenjenek meg.

### JavaScript

A JavaScript egy dinamikus szkriptnyelv, amelyet általában weboldalak fejlesztéséhez használnak. Az egyik legelterjedtebb és legjelentősebb programozási nyelv a webfejlesztésben. Lehetővé teszi, hogy a weboldal interaktív legyen és dinamikusan reagáljon a felhasználó bemenetére. Emellett teljes értékű alkalmazásokat, játékokat is lehet benne készíteni, ilyenkor a weboldal egy megjelenítő szerepet vet fel. Mi a JavaScriptet a weboldal interaktív részének kialakításán felül az adatok API-ból történő lekérésére és betöltésére használjuk.

### jQuery

A jQuery egy könnyűsúlyú, gyors és keresztplatformos JavaScript könyvtár, amelyet főleg webfejlesztés során használnak. Célja, hogy egyszerűbbé és hatékonyabbá tegye a JavaScript alapú webfejlesztést. Népszerűsége és kényelme miatt sok webfejlesztő használja azáltalános feladatok megkönnyítésére és az alkalmazások gyors fejlesztésére.

### JSON

A JSON (JavaScript Object Notation) egy könnyen olvasható adatcsere formátum, amely gépi feldolgozásra alkalmas. A JSON adatokat egy szöveges formátumban ábrázolja, és adatstruktúrákat, például objektumokat és tömböket képes reprezentálni. Mi az API-ban vesszük hasznát, mivel mind a PHP, mind a JavaScript támogatja a formátumot, beépített funkcióival könnyen átalakítható és kezelhető a továbbított adat.

### Bootstrap

A Bootstrap egy ingyenes és nyílt forráskódú frontend keretrendszer weboldalak és webalkalmazások fejlesztéséhez. Elsősorban HTML, CSS és JavaScript alapú. A Bootstrap célja, hogy segítsen az egyenletes és gyorsan fejlődő webes projektek létrehozásában, minimalizálva a tervezési időt és növelve a projekt hatékonyságát. A keretrendszer egy nagyon népszerű eszköz a webfejlesztők körében, mivel lehetővé teszi az egyszerű és gyorsan fejlődő webes projektek létrehozását, miközben biztosítja a reszponzív design és a konzisztens megjelenés előnyeit.

### PHP

A PHP egy általános szerveroldali szkriptnyelv, amelyet dinamikus weblapok készítésére fejlesztették ki. A hagyományos HTML lapokkal ellentétben a kiszolgáló a PHP-kódot nem küldi el az ügyfélnek, hanem a kiszolgáló oldalán a PHP-értelmező motor dolgozza fel azt. A programokban lévő HTML elemek érintetlenül maradnak, de a PHP kódok lefutnak. A kódok végezhetnek adatbázis-lekérdezéseket, létrehozhatnak képeket, fájlokat olvashatnak és írhatnak, kapcsolatot létesíthetnek távoli kiszolgálókkal. A PHP-kódok kimenete a megadott HTML elemekkel együtt kerül az ügyfélhez.

### HTTP, REST API

A REST (Representational State Transfer) egy elterjedt architektúra az alkalmazás-programozási interfészek (API-k) tervezéséhez és megvalósításához. A REST API-k az erőforrásokat (adatokat, szolgáltatásokat) ábrázolják, és az HTTP protokollt használják a kliens és a szerver közötti kommunikációhoz. A REST API-k állapotmentesek, azaz a kliens kérésének tartalmaznia kell az összes szükséges információt. A REST API-k népszerűségét az egyszerűség, az egységes interfész és a skálázhatóság adja, amely lehetővé teszi a nagyobb terhelések kezelését.

### MySQL

A MySQL egy ingyenes, nyílt forráskódú relációs adatbázis-kezelő rendszer. A MySQL könnyen használható, széles körben elterjedt és skálázható, ami azt jelenti, hogy nagy adatmennyiségeket is képes hatékonyan kezelni. Az adatok egyszerű kezelése és a megbízhatósága miatt a MySQL-t széles körben használják az üzleti alkalmazásokban, weboldalakban, mobilalkalmazásokban, játékokban és még sok más területen.

### C#

A C# (C-sharp) egy modern, objektumorientált programozási nyelv. Elsősorban a .NET platformhoz kapcsolódik, és az egyik leggyakrabban használt nyelv a Windows alkalmazások, webalkalmazások és szolgáltatások fejlesztéséhez. A C#-t széles körben használják üzleti alkalmazások, játékok, webalkalmazások, adatszolgáltatások és még sok más fejlesztésére.

### Git, GitHub

A Git egy nyílt forráskódú, elosztott verziókezelő szoftver, vagy másképpen egy szoftverforráskód-kezelő rendszer, amely a sebességre helyezi a hangsúlyt. A GitHub ingyenes internetes szolgálatatás, amely a Git segítségével szoftverfejlesztési verziókövetés-szolgáltatást nyújt. A grafikus felület segítségével egyszerűen hozhatunk létre és kezelhetünk Git repository-kat.

### ChatGPT

A ChatGPT (Generative Pre-trained Transformer)[1] az OpenAI mesterséges intelligencia (MI) kutató laboratórium által kifejlesztett chatbot, mely a felhasználókkal való folyamatos kommunikáció automatizálása során értelmezőmodelleket használ, melyek segítségével a bevitt információkat azonnal interaktívan kezeli. Segítségével a sokszor hosszas keresést, interneten kutakodást lerövidíthetjük, egy-egy problémára másodperceken belül választ kaphatunk. Emellett kisebb algoritmusokat, mint tesztadatok generálása, vagy bizonyos adatok, esetleg kódrészletek helyességének ellenőrzése, is könnyebb, és még inkább gyorsabb elvégezni a ChatGPT segítségével.

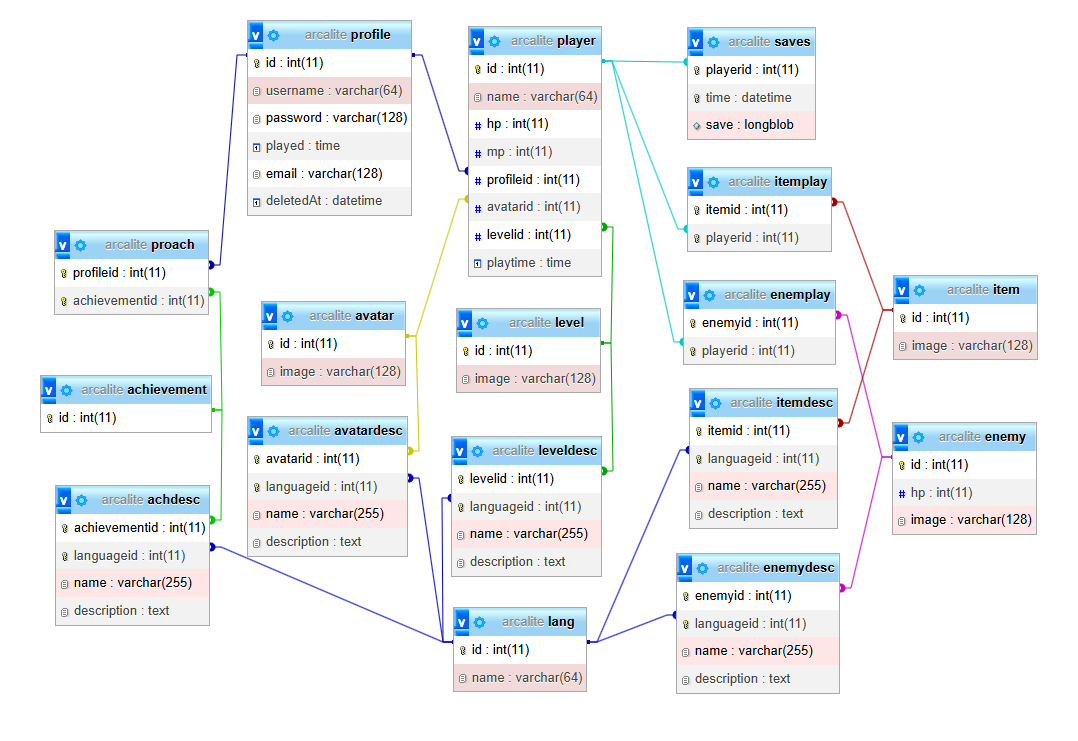
## Adatszerkezet

Adatbázisunk célja a felhasználók adatainak, játékbeli állásuknak, illetve a játék általános informáci-óinak tárolása

### Alapadatok

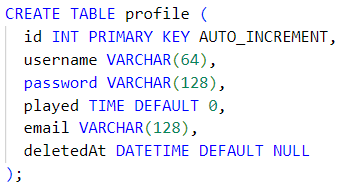
|  |  |
| --- | --- |
| MySQL kliens verzió | 10.4.28 - MariaDB |
| Adatbázis neve | arcalite |
| Tárolómotor | InnoDB |
| Alapértelmezett illesztés | UTF-8 |

### Kapcsolati ábra



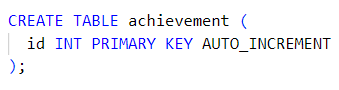
### Táblák

#### A profile tábla

A felhasználók regisztrációs adatait tárolja, illetve az összes játszott időt.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| id | Egész | PK |  |
| username | Szöveg |  | A fiók felhasználóneve |
| password | Szöveg |  | A fiók jelszava (PASSWORD függvénnyel hash-elve) |
| email | Szöveg |  | A fiók e-mail címe |
| played | Idő |  | Az összes játszott idő |
| deletedAt | Dátum és idő |  | A fiók törlésének időpontja |

#### Az achievement tábla

Az elérhető mérföldköveket tartja számon. Egyelőre nincs sokféle adat benne, későbbi fejlesztések miatt lett létrehozva.

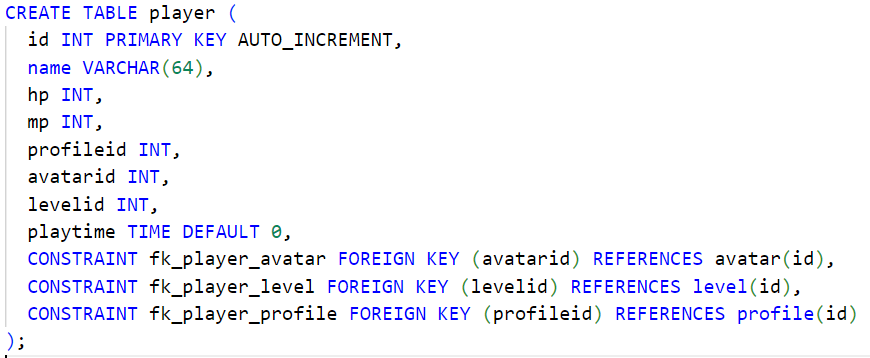
|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| id | Egész | PK |  |

#### A proach tábla

Az adott felhasználó által elért mérföldköveket tartja számon.

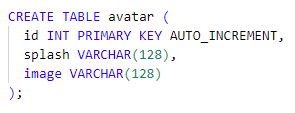
|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| profileid | Egész | PK, FK | A fiók kódja. Kapcsolat a *profile* táblával. |
| achievementid | Egész | PK, FK | A mérföldkő kódja. Kapcsolat az *achievement* táblával. |

#### A player tábla

A felhasználó által készített karakterek adatait tárolja.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| id | Egész | PK |  |
| name | Szöveg |  | A karakter neve |
| hp | Egész |  | A karakter életpontjainak maximális értéke |
| mp | Egész |  | A karakter manapontjainak maximális értéke |
| profileid | Egész | FK | A hozzátartozó fiók kódja. Kapcsolat a *profile* táblával. |
| avatarid | Egész | FK | A karakter avatárképének kódja. Kapcsolat az *avatar* táblával. |
| levelid | Egész | FK | A legutoljára elért szint kódja. Kapcsolat a *level* táblával. |
| playtime | Idő |  | A karakterrel játszott idő. |

#### Az avatar tábla

A karaktereknek választható avatár képek elérési útját tárolja.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| id | Egész | PK |  |
| splash | szöveg |  | Az avatárhoz tartozó splash art elérési útja a szerveren. |
| image | Szöveg |  | Az avatárhoz tartozó kép elérési útja a szerveren. |

#### A level tábla

A szintekhez tartozó képek elérési útját tárolja.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| id | Egész | PK |  |
| image | Szöveg |  | Az szinthez tartozó kép elérési útja a szerveren. |

#### Az enemy tábla

A játékban lévő ellenfelek adatait tárolja.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| id | Egész | PK |  |
| hp | Egész |  | Az ellenfél életpontjainak maximális értéke. |
| image | Szöveg |  | Az ellenfélhez tartozó kép elérési útja a szerveren. |

#### Az enemplay tábla

A karakter által már felfedezett ellenfeleket tárolja. Az adott karakter „csak ezeket ismeri”.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| playerid | Egész | PK, FK | A karakter kódja. Kapcsolat a *player* táblával. |
| enemyid | Egész | PK, FK | Az ellenfél kódja. Kapcsolat az *enemy* táblával. |

#### Az item tábla

A játékban megszerezhető tárgyak adatait tárolja

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| id | Egész | PK |  |
| image | Szöveg |  | A tárgyhoz tartozó kép elérési útja a szerveren. |

#### Az itemplay tábla

A karakter által már megszerzett/felfedezett tárgyakat tárolja. Az adott karakter „csak ezeket ismeri”.

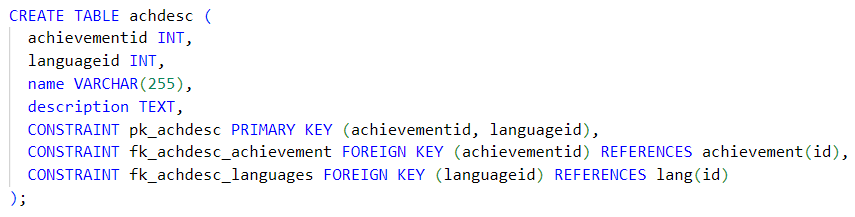
|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| playerid | Egész | PK, FK | A karakter kódja. Kapcsolat a *player* táblával. |
| itemid | Egész | PK, FK | A tárgy kódja. Kapcsolat az *item* táblával. |

#### A lang tábla

A játékban elérhető nyelveket tárolja.

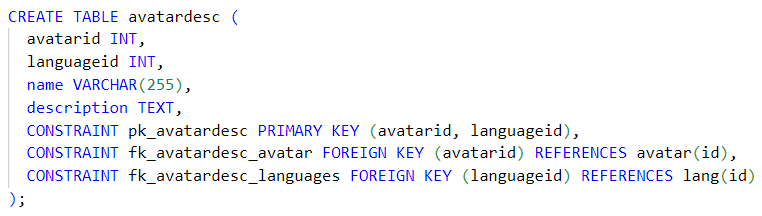
|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| id | Egész | PK |  |
| name | Egész |  | A nyelv megnevezése |

#### Az achdesc tábla

A mérföldkövek nevét és leírását tárolja adott nyelveken.

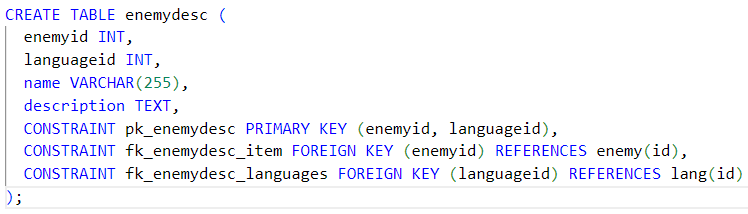
|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| achievementid | Egész | PK, FK | A mérföldkő kódja. Kapcsolat az *achievement* táblával. |
| languageid | Egész | PK, FK | A nyelv kódja. Kapcsolat a *lang* táblával. |
| name | Szöveg |  | A mérföldkő megnevezése az adott nyelven. |
| description | Szöveg |  | A mérföldkő leírása az adott nyelven. |

#### Az avatardesc tábla

Az avatárok nevét és leírását/háttértörténetét tárolja adott nyelveken.

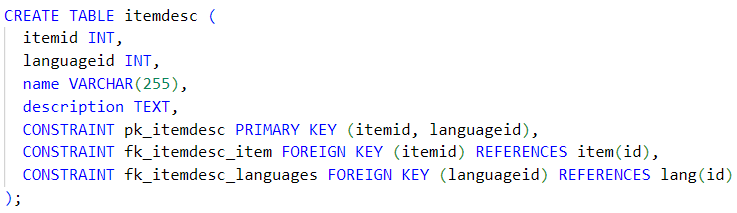
|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| avatarid | Egész | PK, FK | Az avatár kódja. Kapcsolat az *avatar* táblával. |
| languageid | Egész | PK, FK | A nyelv kódja. Kapcsolat a *lang* táblával. |
| name | Szöveg |  | Az avatár megnevezése az adott nyelven. |
| description | Szöveg |  | Az avatár leírása az adott nyelven. |

#### Az enemydesc tábla

Az ellenfelek nevét és leírását tárolja adott nyelveken.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| enemyid | Egész | PK, FK | Az ellenfél kódja. Kapcsolat az *enemy* táblával. |
| languageid | Egész | PK, FK | A nyelv kódja. Kapcsolat a *lang* táblával. |
| name | Szöveg |  | Az ellenfél megnevezése az adott nyelven. |
| description | Szöveg |  | Az ellenfél leírása az adott nyelven. |

#### Az itemdesc tábla

A tárgyak nevét és leírását tárolja adott nyelveken.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| itemid | Egész | PK, FK | A tárgy kódja. Kapcsolat az *item* táblával. |
| languageid | Egész | PK, FK | A nyelv kódja. Kapcsolat a *lang* táblával. |
| name | Szöveg |  | A tárgy megnevezése az adott nyelven. |
| description | Szöveg |  | A tárgy leírása az adott nyelven. |

#### A saves tábla

Az adott karakterrel való utolsó mentési adatokat tárolja.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mezőnév** | **Típus** | **Kulcs** | **Leírás** |
| playerid | Egész | PK, FK | A karakter kódja. Kapcsolat az *player* táblával. |
| time | Dátum és idő |  | A mentés időpontja. |
| save | Longblob |  | A mentési állomány. |

## A weboldal felépítése

### A weboldal szerkezete és formázás

A weboldal 5 darab HTML-fájlból áll, viszont a formázás egy központi CSS fájlban, a *style.css*-ben található, illetve a formázáshoz kapcsolódó JavaScript függvények a *script* mappában a *styleScript.js*-be kerültek.

A weboldalon használt színek többségét CSS-változókkal adjuk meg, és ezek segítségével alakítottunk ki egy sötét és világos témát az oldalnak. Ezt az oldal tetején a menüsávon található kapcsolóval állíthatjuk át. A kapcsoló a háttérben meghívja a *themeChange()* függvényt, mely egy cookie-ban eltárolja a jelenleg kiválasztott témát, majd a *setTheme()* függvény meghívásával beállítja azt. Az oldal *body* eleme kap egy *bsTheme* adattagot, mely értéke a jelenlegi téma, és amely alapján állítja be a CSS a változóit.

### Saját felugróablakok

A weboldalon hibaüzenetek megjelenítésére, illetve a profilműveletek szükséges kommunikációra saját készítésű felugróablakokat használunk. Maga a felugróablak egy div, benne egy címmel, paragrafussal, szöveges bemeneti mezővel és három gombbal. A közös *common.js* szkriptfájlban találhatjuk a felugróablak működéséhez és használatához szükséges függvényeket:

* showPopup(): Az oldal tetejére csúsztatja a felugróablakot.
* closePopup(): Az oldal teteje fölé csúsztatja („elrejti”) a felugróablakot.
* setupPopup(…): A felugróablak mindig megjelenő részeinek beállítását végzi el. Ezek a címszöveg, az üzenet, a „*Mégsem*” gomb beállítása, illetve a többi gomb funkciójának reset-elése.
* setAlert(…): Egy bemenet nélküli felugróablak beállítását végzi el. Paraméterekből beállítja a „*Rendben*” gomb tényleges szövegét, funkcióját, illetve elrejti a nemhasznált részeket.
* setPrompt(…): Egy bemenetet váró felugróablak beállítását végzi el. Paraméterekből beállítja az „*Elküld*” gomb tényleges szövegét, funkcióját, illetve megjeleníti a szükséges elemeket, és elrejti a nemhasználtakat. A paraméterek között szerepel a bemenet helyességét ellenőrző függvény, illetve a helytelen bemenetre visszaadott hibaüzenet is.
* Alert(…): A bemenet nélküli felugróablak használatát lehetővé tevő függvény. Meghívja a megjelenítést, beállítást és az esetleges előző ablak bezárását is.
* Prompt(…):A bemenetet váró felugróablak használatát lehetővé tevő függvény. Meghívja a megjelenítést, beállítást és az esetleges előző ablak bezárását is.

### Az API és a weboldal közti kommunikáció

Az API és a kliens közti kommunikációt JavaScript-tel kezeljük, a jQuery *$.ajax()* függvényének segítségével.

Több végpontnak is szüksége van információkra, mint a felhasználó azonosítója, neve, vagy a megjelenítés nyelvének azonosítója (jelen állapotban mindig 1 (magyar)). Ezeket az oldal bejelentkezéstől kezdve cookie-kban tárolja, és elérésükhöz a common.js állományban áll rendelkezésünkre a *getCookie()* függvény, mely a megadott nevű cookie értékét, illetve ha nincs ilyen nevű cookie, null értéket ad vissza.

A tényleges API kérések függvényei a weboldalakhoz tartozó szkriptekben találhatóak.

|  |  |  |
| --- | --- | --- |
| Fájl | Függvény | Feladat |
| index.js | ready\* | Karakteradatok lekérése az API-n keresztül |
| showCharacters | A karakteradatok betöltése a kártyákba. |
| login.js | ready\* | Hozzárendeli az űrlapokhoz és gombokhoz az őket kezelő függvényeket |
| register | A regisztrációs űrlap adatait ellenőrzi, és kiküldi a verifikációs e-mailt. |
| verifyRegister | Bekéri a kiküldött kódot és validálja a bemenetet. |
| uploadNewProfile | Elküldi a regisztrációt rögzítő API kérést. |
| login | A bejelentkezési űrlap adatai ellenőrzi, elküldi a bejelentkezési adatokat lekérő API kérést. Sikeres bejelentkezés esetén beállítja a cookie-kat és a főoldalra visz. |
| forgotten\_  sendCode | Az elfelejtett jelszó funkció használata után ellenőrzi az e-mail bemenetet, hogy van-e hozzá fiók (ezt egy API kéréssel), és ha van, kiküldi az ellenőrző kódot e-mailben. |
| validEmail | Ellenőrzi e-mail bemenet helyességét. |
| generate  Verification  Code | Egy hatjegyű ellenőrzőkódot generál. |
| getTemplateCode | A küldendő e-mail sablonazonosítóját adja vissza. |
| sendVerification  Email | Elküldi az e-mailt az EmailJS szolgáltatás API-ján keresztüli hívással. |
| forgotten\_  verifyCode | Ellenőrzi a az ellenőrző kódra adott bemenetet. |
| forgotten\_  setNewPassword | Elküldi az új jelszót beállító API kérést. |
| switchToLogin | Előhozza a bejelentkezési űrlapot. |
| switchToRegister | Előhozza a regisztrációs űrlapot. |
| profile.js | changePassword | Megnyit egy felugróablakot, ami a jelszó átállításának módjáról tájékoztat. |
| changeName | Felugróablakokkal megerősítést kér és bekéri az új beállítandó nevet. |
| updateData | Elküldi a felhasználónevet vagy e-mail címet megváltoztató API kérést. |
| askEmailChange | Megerősítést kér az e-mail megváltoztatásáról. |
| getNewEmail | Bekéri felugróablakon az új e-mail címet. |
| verifyEmail | Kiküldi, bekéri és ellenőrzi a hitelesítő kódot. |
| validateAuthcode | Ellenőrzi a megadott hitelesítő kód helyességét. |
| validEmail | Visszaadja, hogy a bevitt email formátuma helyes-e |
| generate  VerificationCode | Generál egy hatjegyű hitelesítő kódot. |
| sendVerification  Email | Kiküldi a hitelesítő kódot tartalmazó e-mailt. |
| askLogout | Megerősítést kér és kijelentkezteti a felhasználót, majd visszadobja a főoldalra. |
| logout | Törli a bejelentkezési adatokat tároló cookie-kat. |
| askDeletion | Megerősítést kér a profil törlésére. |
| deleteProfile | Elküldi a profiltörlést intéző API kérést. |
| ready\* | Lekéri a megjelenítendő karakteradatokat a profil oldalra az API-n keresztül |
| FillTabs | Feltölti az oldalt a lekért tartalommal. |
| ranking.js | ready\* | Meghívja az adatokat lekérő függvényt. |
| FetchRankings | Lekéri az adatokat az API-n keresztül, majd meghívja a feltöltést végző függvényt. |
| FillTable | Feltölti a megadott táblázatot a megkapott adatokkal. |
| sortTable | Rendezi a megadott táblázat sorait a megadott oszlop szerint. (A fejléccellák *click* event-jére hívódik meg) |
| wiki.js | ready\* | Meghívja az adatokat lekérő függvényt. |
| FetchData | Lekéri az adatokat az API-n keresztül, majd meghívja a típushoz megfelelő feltöltő függvényt. |
| writeStats | A címekben a *„megszerzett / összes”* adatokat jeleníti meg a megkapott adatok alapján |
| showItems | Feltölti a tárgyakról szóló kártyákat a megkapott adatokkal |
| showEnemies | Feltölti az ellenfelekről szóló kártyákat a megkapott adatokkal |

\*a *ready* függvények a jQuery *window.onload* eseményre használt rövidítését jelölik (*$(document).ready*)

### Egyéb funkciók a weboldalon

A menüsávon a bejelentkezésfüggő beállítások a *common.js* fájlban találhatóak. Itt ellenőrzi az oldal, hogy van-e *userid* eltárolva: ha nincs, a *Lexikon* menüpont nem lesz használható, illetve a *Bejelentkezés* menüpont fog megjelenni; ha van *userid*, a *Lexikon* elérhető lesz, illetve a *Bejelentkezés* helyett a *Profil* oldalra tudunk eljutni a menüből.

Kijelentkezéskor a már sokszor említett cookie-kat egyszerűen töröljük (felhasználó azonosító, -név), erre a *common.js* fájlban található a *logout* függvény.

## Az API felépítése

## Az alkalmazás felépítése

## Tesztelés

## Továbbfejlesztési lehetőségek

# Felhasználói dokumentáció

## A program rövid ismertetése

## A program feltelepítése

### A játék telepítése

### A weboldal üzembe helyezése

### Az adatbázis felállítása

## Használati útmutató

### A játék irányítása

### A weboldal használata

# Mellékletek