**Interface specifikáció**

Ez a dokumentum leírja, hogy hogyan kell kommunikálni az alkalmazáshoz tartozó REST API-val.

Minden egyes URI-hoz tartozó HTTP metódus definiálva van.

A dátumok [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) szabvány szerint vannak eltárolva.

Általános formátum egy kéréshez:

KÉRÉS TÍPUSA(GET, POST, PUT, DELETE) /OktatoAppAPI/elérésiURI

|  |  |
| --- | --- |
| **Request Header**  Ide kerülnek a kéréshez szükséges headerek. | |
| **Request JSON**  Ide kerülnek a kéréshez szükséges adatok, JSON formátumban. | |
| **Response Header**  Ide kerülnek a válaszhoz tartozó headerek. | |
| **Response JSON if successful**  Ha sikeres a kérés, ide kerülnek JSON formátumban a válaszul kapott adatok.  {  „status\_code” : A sikeres kéréshez tartozó HTTP státuszkód,  „description” : „A sikeres kéréshez tartozó leírás.”  „data” : {  Ide kerülnek a JSON-ban az adatok.  }  } | **Response JSON if unsuccessful**  {  „status\_code” : A sikertelen kéréshez tartozó HTTP státuszkód ,  „description” : „ A sikertelen kéréshez tartozó leírás.”  } |

**Felhasználó adatainak lekérdezése:**

GET /OktatoiAppAPI/myself

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : ”Sikeres lekérdezés!”,  „data” : {  „username” : „username”,  „email”: „email”,  „password” : null,  „first\_name” : „first\_name”,  „last\_name” : „last\_name”,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „last\_login” : „2014-03-12T13:37:27+00:00”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Bárki aki rendelkezik érvényes tokennel, le tudja kérdezni a hozzá tartozó adatokat.

**Felhasználó jogainak lekérdezése:**

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Sikeres lekérdezés!”,  „data” : {  „role\_name”: „role\_name”,  „role\_description” : „role\_description”,  „permissions” : [  {  „permission\_id”:id\_1, „permission\_name”:”permission\_name\_1”  },  .  .  .  {  „permission\_id”:id\_n, „permission\_name”:”permission\_name\_n”  }  ]  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

GET /OktatoiAppAPI/myself/permissions

Leírás: Bárki aki rendelkezik érvényes tokennel, le tudja kérdezni a saját jogait.

**Bejelentkezés**:

POST /OktatoiAppAPI/users/login

|  |  |
| --- | --- |
| **Request Header**  Content-Type: application/json | |
| **Request JSON**  {  „username” : „username”,  „password” : „password”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : „201”,  „description” : „Sikeres bejelentkezés!”,  „data” : {  „id”: id,  „token” : „token”,  „created\_at”: „2014-03-12T13:37:27+00:00”  „expires\_at” : „2014-03-12T13:37:27+00:00”,  „is\_active” : „true”  „user\_id”:id,  „username”: „username”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : „401”,  „description” : „Sikertelen bejelentkezés!  } |

Leírás: JSON body-ban megadott username és jelszó ellenőrzés után készít egy tokent, amit utána a response JSON-ban visszaad, bárki elérheti.

**Kijelentkezés:**

POST /OktatoiAppAPI/logout

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : „200”,  „description” : „Sikeres kijelentkezés!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : „401”,  „description” : „Helytelen token!”  } |

Leírás: A headerben elküldött tokent inaktívvá teszi az adatbázisban, érvényes tokennel lehet elérni.

**Regisztráció:**

POST /OktatoAppAPI/users/signup

|  |  |
| --- | --- |
| **Request Header**  Content-Type: application/json | |
| **Request JSON**  {  „username” : „username”,  „email”: „email”,  „password” : „password”,  „first\_name” : „first\_name”,  „last\_name” : „last\_name”,  „account\_type” : „account\_type”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : „201”,  „description” : „Sikeres regisztáció!”,  „data” : [  {  „id”: „id”,  „username” : „username”,  „email”: „email”,  „first\_name” : „first\_name”,  „last\_name” : „last\_name”,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „last\_login” : null  „account\_type” : „account\_type”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : „400”,  „description” : „Hiányzó adatok!”  }  **or**  {  „status\_code” : „409”,  „description” : „Már létezik ilyen felhasználó!”  } |

Leírás: A regisztációt megvalósító művelet, account type a következő lehet:

* STUDENT
* TEACHER
* ADMIN

Bárki elérheti.

**Userek lekérdezése:**

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Az összes felhasználó sikeresen lekérdezve!”,  „data” : [  {  „id”: id\_1,  „username” : „username\_1”,  „email”: „email”,  „first\_name” : „first\_name\_1”,  „last\_name” : „last\_name\_1”,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „last\_login” : „2014-03-12T13:37:27+00:00”,  „account\_type” : „account\_type\_1”  },  .  .  .  {  „id”: id\_n,  „username” : „username\_n”,  „email”: „email”,  „first\_name” : „first\_name\_n”,  „last\_name” : „last\_name\_n”,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „ last\_login” : „2014-03-12T13:37:27+00:00”,  „account\_type” : „account\_type\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

GET /OktatoAppAPI/users

Leírás: Az összes user lekérdezése, ADMIN vagy TEACHER típusú felhasználóhoz tartozó tokennel lehet lekérdezni.

**User adatainak lekérdezése username szerint:**

GET /OktatoiAppAPI/users/{username}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „A felhasználó sikeresen lekérdezve!”,  „data” : [  {  „id” : id,  „username” : „username”,  „email”: „email”,  „first\_name” : „first\_name”,  „last\_name” : „last\_name”,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „last\_login” : „2014-03-12T13:37:27+00:00”,  „account\_type” : „account\_type”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Felhasználó nem található!”  } |

Leírás: Felhasználó adatainak lekérdezése username szerint, ADMIN vagy TEACHER felhasználóhoz tartozó tokennel lehet lekérni.

**Userhez tartozó statisztikák lekérdezése:**

GET /OktatoAppAPI/users/{username}/statistics

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Userhez tartozó statisztika lekérdezése sikeres!” ,  „data” : [  {  „id” : id\_1,  „gained\_points” : gained\_points\_1,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „user\_id”:id\_1,  „username”: „username\_1”,  „game\_session\_id”:id\_1,  „game\_session\_name” : „game\_session\_name\_1”  },  .  .  .  {  „id” : id\_n,  „gained\_points” : gained\_points\_n,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „user\_id”:id\_n,  „username”: „username\_n”,  „game\_session\_id”:id\_n,  „game\_session\_name” : „game\_session\_name\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Ilyen felhasználó nem létezik!”  } |

Leírás: Visszaadja a {username} felhasználóhoz tartozó összes statisztikát, bármilyen érvényes tokennel le lehet kérdezni.

**Ennek a URI-nak a létjogosultsága még kérdéses.**

**Userhez tartozó statisztika felvétele:**

POST /OktatoAppAPI/users/{username}/statistics

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON**  {  „gained\_points” : gained\_points,  „game\_session\_name” : „game\_session\_name”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : ,  „description” : ,  „data” : {  „id” : id,  „gained\_points” : gained\_points,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „user\_id”:id,  „username”: „username”,  „game\_session\_id”:id,  „game\_session\_name” : „game\_session\_name”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Felhasználó nem található!”  }  **or**  {  „status\_code” : 404,  „description” : „Játékmenet nem található!”  } |

Leírás: {username} azonosítójú felhasználóhoz felvisz egy új statisztikát. Csak a tokenhez tartozó username

**Szerepkörök lekérdezése:**

GET /OktatoAppAPI/roles

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Szerepkörök sikeresen lekérdezve!”,  „data” : [  {  „id”: id\_1,  „role\_name”: „role\_name\_1”,  „role\_description” : „role\_description\_1”  },  .  .  .  {  „id”: id\_n,  „role\_name”: „role\_name\_n”,  „role\_description” : „role\_description\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Szerepkörök és hozzájuk tartozó jogok lekérdezése, bármilyen érvényes tokennel le lehet kérdezni.

**Szerepkör létrehozása:**

POST /OktatoAppAPI/roles

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „role\_name”: „role\_name”,  „role\_description” : „role\_description”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Szerepkör sikeresen létrehozva!”,  „data” : {  „id” : id,  „role\_name”: „role\_name”,  „role\_description” : „role\_description”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 400,  „description” : „Helytelen paraméterek!”  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen szerepkör!”  } |

Leírás: Új szerepkör létrehozása, csak ADMIN felhasználóhoz tartozó tokennel lehetséges.

**Szerepkör törlése**

DELETE /OktatoAppAPI/roles/{role\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Szerepkör sikeresen törölve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Szerepkör nem található!  } |

Leírás: Szerepkör törlése, csak admin felhasználóhoz tartozó tokennel lehet alkalmazni.

**Szerepkör módosítása:**

PUT /OktatoAppAPI/roles/{role\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „role\_name”: „role\_name”,  „role\_description” : „role\_description”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Szerepkör sikeresen módosítva!”,  „data” : {  „id” : id,  „role\_name”: „role\_name”,  „role\_description” : „role\_description”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Szerepkör nem található!  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen szerepkör!”  } |

Leírás: Szerepkör módosítása id szerint, csak admin felhasználóhoz tartozó tokennel lehet elérni.

**Szerepkörhöz tartozó jogosultságok lekérdezése:**

GET /OktatoAppAPI/roles/{role\_name}/permissions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Szerepkörhoz tartozó jogosultságok sikeresen lekérdezve!”,  „data” : {  „role\_name”: „role\_name”,  „permissions”: [  „permission\_name”:”permission\_name\_1”,  .  .  .  „permission\_name”:”permission\_name\_n”  ]  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Szerepkör nem található!”  } |

Leírás: {role\_name} nevű szerepkörhöz tartozó jogokat lekérdezi, érvényes tokennel lehet lekérdezni.

**Szerepkörhöz jogosultság hozzáadása:**

POST /OktatoAppAPI/roles/{role\_name}/permissions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „permission\_name” : „permission\_name”  { | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Jogosultság sikeresen hozzáadva a szerepkörhöz!”,  „data” : {  "role\_name": role\_name,  "permission\_name" „:permission\_name”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Szerepkör nem található!”  }  **or**  {  „status\_code” : 409,  „description” : „Ehhez ez szerepkörhöz már tartozik ilyen jogosultság!”  } |

Leírás: {role\_name} nevű szerepkörhöz hozzáad egy {permission\_name} nevű jogosultságot, csak adminhoz tartozó tokennel végezhető el.

**Szerepkör jogainak változtatása:**

PUT /OktatoAppAPI/roles/{role\_name}/permissions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  [  {  „permission\_name” : „permission\_name\_1”  },  .  .  .  {  „permission\_name” : „permission\_name\_n”  }  ] | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” :200,  „description” : „Jogosultságok sikeresen módosítva!”,  „data” : [  {  „id” : id\_1,  „permission\_name” : „permission\_name\_1”  },  .  .  .  {  „id” : id\_n,  „permission\_name” : „permission\_name\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Szerepkör nem található!”  }  **or**  {  „status\_code” : 409,  „description” : „Ehhez ez szerepkörhöz már tartozik ilyen jogosultság!”  } |

Leírás: A megadott {role\_name} nevű szerepkör jogainak változtatása, csak adminhoz tartozó tokennel végezhető el.

**Szerepkör jogának törlése jog név szerint:**

DELETE /OktatoAppAPI/roles/{role\_name}/permissions/{permission\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Jogosultság sikeresen törölve a szerepkörről!”,  „data” : {  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Szerepkör nem található!”  }  **or**  „status\_code” : 404,  „description” : „Jogosultság nem található a szerepkörnél!”  } |

Leírás: A megadott {role\_name} nevű szerepkör {permission\_name} nevű jogának törlése, csak adminhoz tartozó tokennel végezhető el.

**Jogok lekérdezése:**

GET /OktatoAppAPI/permissions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Jogok sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „permission\_name”: „permission\_name\_1”  },  .  .  .  {  „id” : id\_n,  „permission\_name”: „permission\_name\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Az összes jogosultságfajta lekérdezése, bármilyen tokennel lekérdezhető.

**Jogosultság létrehozása:**

POST /OktatoAppAPI/permissions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „permission\_name”: „permission\_name”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Jogosultság sikeresen létrehozva!”,  „data” : {  „id” : id\_,  „permission\_name”: „permission\_name”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen jogosultság!”  } |

Leírás: Új jogosultság létrehozása, csak admin felhasználóhoz tartozó tokennel lehet elérni.

**Jogosultság törlése:**

DELETE /OktatoAppAPI/permissions/{permission\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Jogosultság sikeresen törölve!”,  „data” : {  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Jogosultság nem található!”  } |

Leírás: Jogosultság törlése {permission\_name} alapján, csak admin felhasználóhoz tartozó tokennel lehet elérni.

**Osztályok** **lekérdezése**:

GET /OktatoAppAPI/classrooms

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Osztálytermek sikeresen lekérdezve!”,  „data” : {  [  {  „id” : id\_1,  „name” : „name\_1”  },  .  .  .  {  „id” : id\_n,  „name” : „name\_n”  }  ]  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Összes osztály adatainak lekérdezése, bármilyen tokennel lekérdezhető.

**Osztály lekérdezése név alapján:**

GET /OktatoAppAPI/classrooms/{classroom\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Osztály sikeresen lekérdezve!”,  „data” : {  „id” : id,  „name” : „name”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 401,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Osztály nem található!”  } |

Leírás: Osztály lekérdezése {classroom\_name} alapján, bármilyen tokennel elvégezhető.

**Osztály törlése név alapján:**

DELETE /OktatoAppAPI/classrooms/{classroom\_name}

Leírás: Osztály törlése {classroom\_name} alapján, csak TEACHER vagy ADMIN típúsú felhasználóhoz tartozó tokennel végezhető el.

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Osztály sikeresen törölve!”,  „data” : {  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 401,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Osztály nem található!”  } |

**Osztály tagjainak megjelenítése:**

GET /OktatoAppAPI/classrooms/{classroom\_name}/users

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Osztály tagjainak lekérdezése sikeres!”,  „data” : [  {  „username” : „username\_1”,  „password” : null,  „first\_name” : „first\_name\_1”,  „last\_name” : „last\_name\_1”,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „last\_login” : „2014-03-12T13:37:27+00:00”  },  .  .  .  {  „username” : „username\_n”,  „password: null,  „first\_name” : „first\_name\_n”,  „last\_name” : „last\_name\_n”,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „last\_login” : „2014-03-12T13:37:27+00:00”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Osztály nem létezik!”  } |

Leírás: az adott nevű osztály összes tagjainak adatainak lekérdezése, bármilyen tokennel lekérdezhető.

**Osztályhoz új tag hozzáadása:**

POST /OktatoAppAPI/classrooms/{classroom\_name}/users

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „classroom\_name” : „classroom\_name”,  „username”: „username”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Felhasználó sikeresen hozzáadva az osztályhoz!”  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Osztály nem létezik!”  }  **or**  {  {  „status\_code” : 400,  „description” : „Felhasználónév nem létezik!”  }  **or**  {  „status\_code” : 409,  „description” : „Már benne van ebben az osztályban ez a felhasználó!”  } |

Leírás: {username} alapján hozzáad egy {classroom\_name} nevű osztályba egy felhasználót, csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Osztályból törlés felhasználónév alapján:**

DELETE /OktatoAppAPI/classrooms/{classroom\_name}/users/{username}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Felhasználó sikeresen törölve az osztályból!”  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem található ilyen osztály!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem található ilyen felhasználó az osztályban!”  } |

Leírás: {classroom\_name} nevű osztályból töröl {username} nevű felhasználót, csak TEACHER vagy ADMIN típusú felhasználóhoz tartozó tokennel végezhető el.

**Összes osztályhoz tartozó összes játékmód lekérdezése:**

GET /OktatoAppAPI/classrooms/game-modes

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Osztályokhoz tartozó játékok sikeresen lekérdezve!”,  „data” : [  {  „game\_name” : „game\_name\_1”,  „game\_id” : game\_id\_1,  „classroom\_name” : „classroom\_name\_1”,  „classroom\_id” : classroom\_id\_1,  „is\_active” : „is\_active\_1”,  },  .  .  .  {  „game\_name” : „game\_name\_n”,  „game\_id” : game\_id\_n,  „classroom\_name” : „classroom\_name\_n”,  „classroom\_id” : classroom\_id\_n,  „is\_active” : „is\_active\_n”,  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Lekérdezi az összes osztályhoz tartozó összes játékotmódot, illetve hogy aktívak-e, bármilyen tokennel lekérdezhető.

**Osztályhoz tartozó játékok lekérdezése:**

GET /OktatoAppAPI/classrooms/{classroom\_name}/game-modes

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Az osztályhoz tartozó játékok sikeresen lekérdezve!”,  „data” : [  {  „game\_name” : „game\_name\_1”,  „game\_id” : game\_id\_1,  „classroom\_name” : „classroom\_name\_1”,  „classroom\_id” : classroom\_id\_1,  „is\_active” : „is\_active\_1”,  },  .  .  .  {  „game\_name” : „game\_name\_n”,  „game\_id” : game\_id\_n,  „classroom\_name” : „classroom\_name\_n”,  „classroom\_id” : classroom\_id\_n,  „is\_active” : „is\_active\_n”,  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem található ilyen osztály!”  } |

Leírás: Visszaadja a {classroom\_name} nevű osztályhoz tartozó összes játékmódot, illetve hogy aktívak-e vagy sem, bármilyen tokennel lekérdezhető.

**Osztályhoz új játékmód felvétele:**

POST /OktatoAppAPI/classrooms/{classroom\_name}/game-modes

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „game\_name” : „game\_name”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Az osztályhoz a játék sikeresen felvéve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem található ilyen osztály!”  }  **or**  {  „status\_code” : 409,  „description” : „Ehhez ez osztályhoz már tartozik ilyen játék!”  } |

Leírás: {classroom\_name} nevű osztályhoz felvesz egy {game\_name} nevű játékmódot, csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel lehet elvégezni.

**Osztályhoz tartozó játékmód törlése:**

DELETE /OktatoAppAPI/classrooms/{classroom\_name}/game-modes/{game\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Osztályhoz tartozó játék sikeresen törölve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Osztály nem található!”  }  **or**  {  „status\_code” : 404,  „description” : „Játék nem található!”  } |

Leírás: {classroom\_name} nevű osztályból töröl egy {game\_name} nevű játékmódot, csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel lehet elvégezni.

**Osztályhoz tartozó játékmód módosítása:**

PUT /OktatoAppAPI/classrooms/{classroom\_name}/game-modes/{game\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „game\_name” : „game\_name”,  „classroom\_name” : „classroom\_name”,  „is\_active” : „is\_active”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Osztályhoz tartozó játék adatainak módosítása sikeres!”,  „data” : {  „game\_name” : „game\_name”,  „game\_id” : game\_id,  „classroom\_name” : „classroom\_name”,  „classroom\_id” : classroom\_id,  „is\_active” : „is\_active”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Osztály nem található!”  }  **or**  {  „status\_code” : 404,  „description” : „Játék nem található!”  } |

Leírás: {classroom\_name} nevű osztály {game\_name} nevű játékmódjának adatainak módosítása. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel lehet végezni. **Felhasználása játék de/aktiválására.**

**Játékmódok lekérdezése:**

GET /OktatoiAppAPI/game-modes

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200  „description” : „Játékok sikeresen lekérdezve!”  „data” : [  {  „id” : id\_1,  „name” : „name\_1”,  „description” : „description\_1”,  }  .  .  .  {  „id” : id\_n,  „name” : „name\_n”,  „description” : „description\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Lekérdezi az összes játékmód adatait, bármilyen tokennel lekérdezhető.

**Játékmód lekérdezése név szerint:**

GET /OktatoAppAPI/game-modes/{game\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Játék sikeresen lekérdezve!”,  „data” : {  „id” : id,  „name” : „name”,  „description” : „description”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  „status\_code” : 404,  „description” : „Játékmód nem található!”  } |

Leírás: {game\_name} szerint lekérdezi a játékmód adatait, bármilyen tokennel lekérdezhető.

**Új játékmód létrehozása:**

POST /OktatoAppAPI/game-modes

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „name” : „name”,  „description” : „description”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Játék sikeresen létrehozva!”,  „data” : {  „id” : id,  „name” : „name”,  „description” : „description”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen játék!”  } |

Leírás: Létrehoz egy új játékmódot az adatbázisban, csak ADMIN felhasználóhoz tartozó tokennel végezhető el.

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Játék sikeresen törölve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Játék nem található!”  } |

**Játékmód törlése:**

DELETE /OktatoAppAPI/game-modes/{game\_name}

Leírás: Játékmód törlése {game\_name} szerint, csak ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Játékmód módosítása név szerint:**

PUT /OktatoAppAPI/game-modes/{game\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „name” : „name”,  „description” : „description”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Játék sikeresen módosítva!”,  „data” : {  „id” : id,  „name” : „name”,  „description” : „description”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Játék nem található!”  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen játék!”  } |

Leírás: {game\_name} nevű játék adatainak módosítása, csak ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Játékmenetek lekérdezése játékmód szerint:**

GET /OktatoAppAPI/game-modes/{game\_name}/game-sessions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „A játékmódhoz tartozó játékmenetek sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „session\_name” : „session\_name\_1”,  „difficulty\_level” : difficulty\_level\_1,  „max\_points” : max\_points\_1,  „game\_name” : „game\_name\_1”,  „game\_id” : game\_id\_1,  },  .  .  .  {  „id” : id\_n,  „session\_name” : „session\_name\_n”,  „difficulty\_level” : difficulty\_level\_n,  „max\_points” : max\_points\_n,  „game\_name” : „game\_name\_n”,  „game\_id” : game\_id\_n  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Játékmód nem létezik!”  } |

Leírás: Visszaadja a {game\_name} nevű játékmódhoz tartozó összes játékmenetet, bármilyen tokennel lekérdezhető.

**Játékmódhoz tartozó osztályok lekérdezése:**

GET /OktatoAppAPI/game-modes/{game\_name}/classrooms

Leírás: Visszaadja a {game\_name} nevű játékmódhoz tartozó osztályokat, bármilyen tokennel lekérdezhető.

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „A játékhoz tartozó adatok sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „classroom\_name” : „classroom\_name \_1”,  „classroom\_id” : classroom\_id\_1,  „is\_active” : „is\_active\_1”,  },  .  .  .  {  „id” : id\_n,  „classroom\_name” : „classroom\_name \_n”,  „classroom\_id” : classroom\_id\_n,  „is\_active” : „is\_active\_n”,  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem található ilyen játékmód!”  } |

**Nehézségi szintek lekérdezése:**

GET /OktatoAppAPI/difficulties

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Nehézségi szintek sikeresen lekérdezve!”,  „data” : [  {  „difficulty\_level” : difficulty\_level\_1,  „difficulty\_name” : „difficulty\_name\_1  },  .  .  .  {  „difficulty\_level” : difficulty\_level\_n,  „difficulty\_name” : „difficulty\_name\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Az összes nehézségi szint lekérdezése, bármilyen tokennel lekérdezhető.

**Nehézségi szint lekérdezése szint alapján:**

GET /OktatoAppAPI/difficulties/{difficulty\_level}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Nehézségi szint sikeresen lekérdezve!”,  „data” : {  „difficulty\_level” : difficulty\_level,  „difficulty\_name” : „difficulty\_name”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem létezik ilyen nehézségi szint!”  } |

Leírás: {difficulty\_level} azonosítójú nehézségi szintet kérdezi le, bármilyen tokennel elvégezhető.

**Új nehézségi szint felvétele:**

POST /OktatoAppAPI/difficulties

Leírás: Új nehézségi szintet visz fel, csak TEACHER vagy ADMIN típúsú felhasználóhoz tartozó tokennel lehet elvégezni.

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „difficulty\_level” : difficulty\_level,  „difficulty\_name” : „difficulty\_name”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Nehézségi szint sikeresen felvéve!”,  „data” : {  „difficulty\_level” : difficulty\_level,  „difficulty\_name” : „difficulty\_name”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen nehézségi szint!”  } |

**Nehézségi szint törlése:**

DELETE /OktatoAppAPI/difficulties/{difficulty\_level}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Nehézségi szint sikeresen törölve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Ilyen nehézségi szint nem létezik!”  } |

Leírás: Nehézségi szint törlése {difficulty\_level} alapján, csak TEACHER vagy ADMIN típúsú felhasználóhoz tartozó tokennel végezhető el.

**Összes játékmenet lekérdezése:**

GET /OktatoAppAPI/game-sessions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Játékmenetek sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „game\_name” : „game\_name\_1”,  „game\_id” : game\_id\_1,  „difficulty\_level” : difficulty\_level\_1,  „max\_points” : max\_points\_1  },  .  .  .  {  „id” : id\_n,  „game\_mode” : „game\_mode\_n”,  „game\_id” : game\_id\_n,  „difficulty\_level” : difficulty\_level\_n,  „max\_points” : max\_points\_n  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Lekérdezi az összes játékmenetet. A válasz tartalmazza, hogy mi a játékmód típusa, milyen a nehézségi szintje, illetve hogy maximum hány pontot lehet megszerezni ebben a játékmenetben. Bármilyen tokennel lekérdezhető.

**Játékmenet lekérdezése név szerint:**

GET /OktatoAppAPI/game-sessions/{session\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „A játékmenethez tartozó adatok sikeresen lekérdezve!”,  „data” : {  „id” : id,  „session\_name” : „session\_name”,  „difficulty\_level” : difficulty\_level,  „max\_points” : max\_points,  „game\_id” : game\_id,  „game\_name” : „game\_name”,  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Játékmenet nem létezik!”  } |

Leírás: Visszaadja a {session\_name} nevű játékmenetet adatait. Bármilyen tokennel elvégezhető.

**Játékmenethez tartozó összes statisztika lekérdezése:**

GET /OktatoAppAPI/game-sessions/{session\_name}/statistics

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Játékmenethez tartozó statisztika sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „gained\_points” : gained\_points\_1,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „username” : „username\_1”,  „user\_id”: user\_id\_1,  „game\_session\_id” : game\_session\_id\_1,  „game\_session\_name” : „game\_session\_name\_1”  },  .  .  .  {  „id” : id\_n,  „gained\_points” : gained\_points\_n,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „username” : „username\_n”,  „user\_id”: user\_id\_n,  „game\_session\_id” : game\_session\_id\_n,  „game\_session\_name” : „game\_session\_name\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Játékmenet nem található!”  } |

Leírás: {session\_name} nevű játékmenethez tartozó összes statisztikát visszaadja. Bármilyen tokennel elvégezhető.

**Játékmenethez tartozó összes kérdés lekérdezése:**

GET /OktatoAppAPI/game-sessions/{session\_name}/questions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Játékmenethez tartozó kérdések sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „question” : „question\_1”,  „points” : points\_1,  „game\_session\_name”: „game\_session\_name\_1”,  „game\_session\_id” : „game\_session\_id\_n”  },  .  .  .  {  „id” : id\_n,  „question” : „question\_n”,  „points” : points\_n,  „game\_session\_name”: „game\_session\_name\_1”,  „game\_session\_id” : „game\_session\_id\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Játékmenet nem található!”  } |

Leírás: {game\_session} nevű játékmenethez lekérdezi az összes kérdést. Bármilyen tokennel lekérdezhető.

**Játékmenet létrehozása:**

POST /OktatoAppAPI/game-sessions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „session\_name” : „session\_name”,  „game\_name” : „game\_name”,  „max\_points” : max\_points,  „difficulty\_level” : difficulty\_level  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Játékmenet sikeresen létrehozva!”,  „data” : {  „id” : id,  „session\_name” : „session\_name”,  „game\_name” : „game\_name”,  „game\_id” : game\_id,  „max\_points” : max\_points,  „difficulty\_level” : difficulty\_level  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen nevű játékmenet!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem létezik ilyen nevű játékmód!”  } |

Leírás: Létrehoz egy új játékmenetet a {game\_name} nevű játékmódhoz {game\_session} néven. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel lehet elvégezni.

**Játékmenet törlése név szerint:**

DELETE /OktatoAppAPI/game-sessions/{session\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Játékmenet sikeresen törölve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem létező játékmód!”  } |

Leírás: {session\_name} nevű játékmenetet töröl. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel elvégezhető.

**Játékmenet módosítása név alapján:**

PUT /OktatoAppAPI/game-sessions/{session\_name}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „session\_name” : „session\_name”,  „game\_name” : „game\_name”,  „max\_points” : max\_points,  „difficulty\_level” : difficulty\_level  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Játékmód sikeresen módosítva!”,  „data” : {  „id” : id,  „session\_name” : „session\_name”,  „max\_points” : max\_points,  „difficulty\_level” : difficulty\_level,  „game\_id” : game\_id,  „game\_name”: „game\_name”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen nevű játékmenet!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem létezik ilyen nevű játékmenet!”  } |

Leírás: Játékmenet adatait módosítja {session\_name} alapján. Csak TEACHER vagy ADMIN felhasználhóz tartozó tokennel lehet elvégezni.

**Összes statisztika lekérdezése:**

GET /OktatoAppAPI/statistics

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Statisztika sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „gained\_points” : gained\_points\_1,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „username” : „username\_1”,  „user\_id” : user\_id\_1,  „session\_name” : „session\_name\_1”,  „session\_id”: session\_id\_1  },  .  .  .  {  „id” : id\_n,  „gained\_points” : gained\_points\_n,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „username” : „username\_n”,  „user\_id” : user\_id\_n,  „session\_name” : „session\_name\_n”,  „session\_id”: session\_id\_n  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Az összes játékmenethez tartozó összes statisztika lekérdezése. Bármilyen tokennel lekérdezhető.

**Új statisztika hozzáadása:**

POST /OktatoAppAPI/statistics

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „username” : „username”,  „gained\_points” : gained\_points,  „session\_name” : „session\_name”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Új statisztika sikeresen hozzáadva!”,  „data” : {  „id” : id,  „gained\_points” : gained\_points,  „created\_at” : „2014-03-12T13:37:27+00:00”,  „username” : „username”,  „session\_name” : „session\_name”,  „session\_id” : session\_id  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: „username” nevű felhasználóhoz új statisztikát visz fel az adatbázisban a „session\_name” nevű játékmenethez. Bármilyen tokennel elvégezhető.

**Statisztika törlése id alapján:**

DELETE /OktatoAppAPI/statistics/{id}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Statisztika sikeresen törölve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem létező statisztika!”  } |

Leírás: {id} azonosítóval rendelkező statisztikát töröl. Csak ADMIN típúsú felhasználóval végezhető el.

**Összes kérdés lekérdezése:**

GET /OktatoAppAPI/questions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Összes kérdés sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „question” : „question\_1”,  „points” : points\_1,  „session\_id” : session\_id\_1,  „session\_name” : „session\_name\_1”  },  .  .  .  {  „id” : id\_n,  „question” : „question\_n”,  „points” : points\_n,  „session\_id” : session\_id\_n,  „session\_name” : „session\_name\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Lekérdezi az összes kérdést. Bármilyen tokennel elvégezhető.

**Kérdéshez tartozó rossz válaszok lekérdezése id alapján:**

GET /OktatoAppAPI/questions/{question\_id}/wrong-answers

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Kérdéshez tartozó rossz válaszok sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „wrong\_answer” : „wrong\_answer\_1”,  „question\_id” : question\_id,  „question”: „question”  },  .  .  .  {  „id” : id\_n,  „wrong\_answer” : „wrong\_answer\_n”,  „question\_id” : question\_id,  „question” : „question”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Kérdés nem található!”  } |

Leírás: {question\_id} azonosítóhoz tartozó kérdésnek visszaadja az összes rossz válaszát. Bármilyen tokennel elvégezhető.

**Kérdéshez tartozó helyes válasz lekérdezése:**

GET /OktatoAppAPI/questions/{question\_id}/correct-answer

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Kérdéshez tartozó helyes válasz sikeresen lekérdezve!”,  „data” : {  „id” : id,  „correct\_answer” : „correct\_answer”,  „question\_id” : question\_id  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Kérdés nem található!”  } |

Leírás: {question\_id} azonosítójú kérdéshez tartozó helyes választ kérdezi le. Bármilyen tokennel lekérdezhető.

**Új kérdés felvétele:**

POST /OktatoAppAPI/questions

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „question” : „question”,  „points” : points,  „session\_name” : „session\_name”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Új kérdés sikeresen felvéve!”,  „data” : {  „id” : id,  „question” : „question”,  „points” : points,  „session\_id” : session\_id,  „session\_name” : „session\_name”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen kérdés!”  } |

Leírás: „session\_name” nevű játékmenethez visz fel egy új kérdést. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Kérdés törlése id alapján:**

DELETE /OktatoAppAPI/questions/{question\_id}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Kérdés sikeresen törölve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Kérdés nem található!”  } |

Leírás: Megadott {question\_id} alapján töröl egy kérdést. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Kérdés módosítása id alapján:**

PUT /OktatoAppAPI/questions/{question\_id}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „question” : „question”,  „points” : points,  „session\_name” : session\_name  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Kérdés sikeresen megváltoztatva!”,  „data” : {  „id” : id,  „question” : „question”,  „points” : points,  „session\_id” : session\_id,  „session\_name” : „session\_name”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 409,  „description” : „Már létezik ilyen kérdés!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem létezik ilyen kérdés!”  } |

Leírás: {question\_id} azonosítójú kérdést megváltoztatja. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Rossz válaszok lekérdezése:**

GET /OktatoAppAPI/wrong-answers

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Rossz válaszok sikeresen lekérdezve!”,  „data” : [  {  „id” : id\_1,  „wrong\_answer” : „wrong\_answer\_1”,  „question\_name” : „question\_name\_1”,  „question\_id” : „question\_id\_1”,  },  .  .  .  {  „id” : id\_n,  „wrong\_answer” : „wrong\_answer\_n”,  „question\_name” : „question\_name\_n”,  „question\_id” : „question\_id\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Lekérdezi az összes rossz kérdést, bármilyen tokennel lekérdezhető. Bármilyen tokennel lekérdezhető.

**Új rossz válasz felvitele:**

POST /OktatoAppAPI/questions/{question\_id}/wrong-answers

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token>  Content-Type: application/json | |
| **Request JSON**  {  „wrong\_answer” : „wrong\_answer”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Rossz válasz sikeresen felvéve!”,  „data” : {  „id”:id,  „wrong\_answer”:”wrong\_answer”,  „question\_id”:question\_id  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Nem létezik ilyen kérdés!”  } |

Leírás: {question\_id} azonosítójú kérdéshez visz fel egy új rossz választ. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Rossz válasz módosítása id alapján:**

PUT /OktatoAppAPI/questions/{question\_id}/wrong-answers/{answer\_id}

|  |  |
| --- | --- |
| **Request Header**  Content-Type: application/json  Authorization: Bearer <token> | |
| **Request JSON**  {  „wrong\_answer”:”wrong\_answer”,  „question\_id”:question\_id  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Rossz válasz sikeresen megváltoztatva!”,  „data” : {  „id”:id,  „wrong\_answer”:”wrong\_answer”,  „question\_id”:question\_id  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Kérdés nem található!”  }  **or**  {  „status\_code” : 404,  „description” : „Rossz válasz nem található!”  } |

Leírás: {question\_id} azonosítójú kérdésnek a {answer\_id} azonosítójú rossz válaszát változtatja meg. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Rossz kérdés törlése id alapján:**

DELETE /OktatoAppAPI/questions/{question\_id}/wrong-answers/{answer\_id}

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Rossz válasz sikeresen törölve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Kérdés nem található!”  }  **or**  {  „status\_code” : 404,  „description” : „Rossz válasz nem található!”  } |

Leírás: {question\_id} azonosítójú kérdéshez tartozó {answer\_id} azonosítójú rossz választ töröl. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Helyes válaszok lekérdezése:**

GET /OktatoAppAPI/correct-answers

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Helyes válaszok sikeresen lekérdezve!”,  „data” : [  {  „question\_id”:question\_id\_1,  „question”:”question\_1”,  „correct\_answer”:”correct\_answer\_1”  },  .  .  .  {  „question\_id”:question\_id\_n,  „question”:”question\_n”,  „correct\_answer”:”correct\_answer\_n”  }  ]  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  } |

Leírás: Az összes kérdéshez tartozó helyes válaszokat lekérdezi. Bármilyen tokennel elvégezhető.

**Helyes válasz felvétele kérdéshez:**

POST /OktatoAppAPI/questions/{question\_id}/correct-answers

|  |  |
| --- | --- |
| **Request Header**  Content-Type: application/json  Authorization: Bearer <token> | |
| **Request JSON**  {  „question\_id”:question\_id,  „correct\_answer”:”correct\_answer”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 201,  „description” : „Helyesválasz sikeresen felvéve!”,  „data” : {  „question\_id”:question\_id,  „correct\_answer”:”correct\_answer”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 409,  „description” : „Ehhez a kérdéshez már tartozik helyes válasz!”  } |

Leírás: {question\_id} azonosítóhoz tartozó kérdéshez visz fel helyes választ. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Helyes válasz módosítása:**

PUT /OktatoAppAPI/questions/{question\_id}/correct-answers

|  |  |
| --- | --- |
| **Request Header**  Content-Type: application/json  Authorization: Bearer <token> | |
| **Request JSON**  {  „correct\_answer”:”correct\_answer”  } | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Helyes válasz sikeresen megváltoztatva!”,  „data” : {  „question\_id”:question\_id,  „correct\_answer”:”correct\_answer”  }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Helyes válasz nem található!”  } |

Leírás: {question\_id} azonosítójú kérdéshez módosítja a hozzátartozó helyes választ. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.

**Helyes válasz törlése kérdés id alapján:**

DELETE /OktatoAppAPI/questions/{question\_id}/correct-answers

|  |  |
| --- | --- |
| **Request Header**  Authorization: Bearer <token> | |
| **Request JSON** | |
| **Response Header**  Content-Type: application/json | |
| **Response JSON if successful**  {  „status\_code” : 200,  „description” : „Kérdéshez tartozó helyes válasz sikeresen törölve!”,  „data” : {    }  } | **Response JSON if unsuccessful**  {  „status\_code” : 400,  „description” : „Jogtalan hozzáférés!”  }  **or**  {  „status\_code” : 404,  „description” : „Helyes válasz nem található!”  }  **or**  {  „status\_code” : 404,  „description” : „Kérdés nem található!”  } |

Leírás: {question\_id} azonosítójú kérdéshez tartozó helyes választ törli ki. Csak TEACHER vagy ADMIN felhasználóhoz tartozó tokennel végezhető el.