Schnittstelle von Lusio

# Einleitung

Werte eingeschlossen von geschweiften Klammern stellen Parameter dar.

Beispiel: /resource/get/{name}

Vor jeder URL muss der Pfad zum Servlet enthalten sein (z.B. http://lusio.viascom.ch/lusio).

## Rückgabetyp

Die Antwort sämtlicher Service-Funktionen weisen die gleiche Struktur auf. Der Typ ServiceResult<T> widerspiegelt dies.

Struktur:

{

"content": {

NUTZDATEN

},

"status": "successful",

"tag": null

}

Die Eigenschaft content beinhaltet die eigentlichen Nutzdaten. Die Eigenschaft status gibt an, ob die Anfrage erfolgreich ausgeführt wurde oder ein Fehler vorliegt (failed).

tag kann beliebige zusätzliche Daten enthalten (ggf. Debugging).

## Fehlerhandling

Tritt ein unerwarteter Fehler auf, so wird eine ServiceException und HTTP Status 500 zurückgegeben. Wird eine Ressource nicht gefunden, so wird HTTP Status 404 zurückgegeben (keine ServiceException enthalten).

{

"content": {

"message": "Failed to execute SQL query.",

"exception": "java.sql.SQLException: ...",

"responseStatusCode": 500,

"requestedType": null,

"requestUrl": null,

"code": "DB\_SQL\_EXECUTE\_ERROR",

"requestParams": [

{

"name": "sql",

"value": "SELECT ERROR"

}

]

},

"status": "failed",

"tag": null

}

# Funktionen

## Login

**HTTP Methode**

POST

**URL**

http://lusio.viascom.ch/lusio/login

**Beschreibung**

Lädt die Basis-Mitgliedsdaten für die übergebene Kundennummer.

Parameter

| Name | Typ | Beschreibung |
| --- | --- | --- |
| username | String | Benutzername des Benutzers |
| password | String | Passwort des Benutzer im Klartext |

**Rückgabewert**

Typ: Session

**Fehler**

Bei einem unerwarteten Fehler wird der HTTP-Statuscode 500 zurückgegeben und eine JSON-ServiceException.

**Beispiel**

{  
 "content": {  
 "id": " 9eeee900-87f9-11e2-9e96-0800200c9a66",  
 },  
 "status": "successful",  
 "tag": null  
}

## Logout

## getAccountInformations

## getGameStats

## getGames

## createTipp