**Documentação da API**

SEÇÃO 1 - REQUISIÇÕES

LOGIN

* GET

1. (“/login/verify/token”) – Utilizada para verificar se o token de um usuário foi expedido pela aplicação.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "token": "token"  } | {  "success": true,  "error": null  } | {  "success": false,  "error": error,  "status": 400,  } | {  "success": false,  "error": error,  "status": 500,  } |

* POST

1. (“/login/auth”) – Utilizada para logar o usuário na plataforma.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "email": "email",  "password": "password"  } | {  "success": true,  "error": null,  "user": User  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

USER

* GET

1. (“/users”) – Lista todos os usuários.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Header  Header Name – Value  x-access-token - tokenValue | {  "success": true,  "error": null,  "users": Users  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

1. (“/users/id”) – Busca por um usuário com o ID passado na requisição.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| No Body  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "user": User  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

1. (“/users/search/key") – Busca por um usuário com uma chave de busca.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| No Body  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "user": User  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

* POST

1. (“/users”) – Cria um novo usuário.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "email": "email",  "password": "password",  "name": "name"  } | {  "success": true,  "error": null,  "user": User  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

* UPDATE

1. (“/users/id”) – Atualiza parâmetros do usuário por ID.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "email": "email",  "password": "password",  "name": "name"  }  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "user": User  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

* DELETE

1. (“/users/id”) – Deleta um usuário por ID.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| No Body  Header  x-access-token - tokenValue | {  "success": true,  "error": null  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

MOVIES

* GET

1. (“/movies”) – Lista todos os filmes.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| No Body  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "movies":Movies,  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

1. (“/movies/id”) – Busca por um filme com o ID passado na requisição

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| No Body  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "movie":Movie,  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

1. (“/movies/search/key”) – Busca por um filme com uma chave de busca.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| No Body  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "movies":Movies,  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

* POST

1. (“/movies”) – Cria um novo filme.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "title": "title",  "director": "director",  "avaliableCopies": 10  }  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "movie":Movie,  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

1. (“/movies/user/rent”) – Aluga um filme a um usuário.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "user\_id": "user\_id",  "movie\_id": "movie\_id"  }  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "movie":Movie,  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

* UPDATE

1. (“/movies/id”) – Atualiza parâmetros de um filme por ID.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "title": "title",  "director": "director",  "avaliableCopies": 10  }  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "movie":Movie,  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

* DELETE

1. (“/movies/id”) – Deleta um filme por ID.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| No Body  Header  x-access-token - tokenValue | {  "success": true,  "error": null  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

1. (“/movies/user/rent”) – Remove a locação de um filme.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "user\_id": "user\_id",  "movie\_id": "movie\_id",  }  Header  x-access-token - tokenValue | {  "success": true,  "error": null  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

VIDEO STORE

* GET

1. (“/stores”) – Busca as locadoras de filmes.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| No Body  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "stores": Stores,  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

* POST

1. (“/stores”) – Cria uma locadora de filme.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "user\_id":"user\_id",  "name": "name"  }  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "store": Store,  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

* UPDATE

1. (“/stores/id”) – Atualiza uma locadora de filme por ID

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| Body  {  "name": "name"  }  Header  x-access-token - tokenValue | {  "success": true,  "error": null,  "store": Store,  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

* DELETE

1. (“/stores/id”) – Deleta uma locadora por ID.

|  |  |  |  |
| --- | --- | --- | --- |
| FORMAT | 200 OK | 400 NON OK | 500 NON OK |
| No Body  Header  x-access-token - tokenValue | {  "success": true,  "error": null  } | {  "success": false,  "error": error  } | {  "success": false,  "error": error  } |

SEÇÃO 2 – RETORNO DE ERROS E MODELOS DE DATA

Status 200 – Todo o processo foi feito sem nenhum problema.

Status 400 – O processo não foi feito por falta de parâmetros.

Status 500 – Erro no servidor, o processo não foi feito.

LOGIN

* User :

{

"token": "token",

"name": "name"

}

* Error :

{

"error": "O usuário não existe"

}

{

"error": "Faltam parametros"

}

{

"error": "Token Inválido"

}

{

"error": "Aplicação fora do ar"

}

USER

* User :

{

"id": id,

"name": name,

"email": email,

"updatedAt":Date,

"createdAt": "Date

}

* Users:

[{

"id": id,

"name": name,

"email": email,

"updatedAt":Date,

"createdAt": "Date

}]

Array of User

* Error :

{

"error": "O usuário não existe"

}

{

"error": "Faltam parametros"

}

{

"error": "Token Inválido"

}

{

"error": "Aplicação fora do ar"

}

MOVIES

* Movie :

{

"title": "title",

"director": "director",

"avaliableCopies": 10

}

* Movies:

[{

"title": "title",

"director": "director",

"avaliableCopies": 10

},

{

"title": "title",

"director": "director",

"avaliableCopies": 10

}]

.

* Error :

{

"error": "O filme não existe"

}

{

"error": "Faltam parametros"

}

{

"error": "Token Inválido"

}

{

"error": "Não é possível alugar este filme, cópias insuficientes"

}

{

"error": "Aplicação fora do ar"

}

VIDEO STORE

* Store :

{

"user ": User,

"name": "name"

}

* Stores:

[{

"user ": User,

"name": "name"

},

{

"user ": User

"name": "name"

}]

* Error :

{

"error": "A locadora não existe"

}

{

"error": "Faltam parametros"

}

{

"error": "Token Inválido"

}

{

"error": "Aplicação fora do ar"

}