

Ressource	HTTP-Verb	Semantik	Content-Type Request	Content-Type Response	Statuscode
/users	GET	Gibt die Liste aller Benutzer zurück Funktion: getAllUsers()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "mail" : {"type": "string"} "age" : {"type": "int"} }	{ "id": {"type": "UUID"}, "mail": {"type": "string"}, "name": {"type": "int"} }	201 - Die Liste aller Benutzer wurde zurückgegeben. 400 - Es sind keine User vorhanden, NoSuchElementException wird ausgegeben.
/user	POST	Erstellt einen Benutzer Funktion: createUser()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "mail" : {"type": "string"} "age" : {"type": "int"} }	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "mail" : {"type": "string"} "age" : {"type": "int"} }	201 – Der Benutzer wurde erstellt.
/user/{userID}	PATCH	Update eines Benutzers Funktion: updateUser()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "mail" : {"type": "string"} "age" : {"type": "int"} }	{ "name" : {"type": "string"} "mail" : {"type": "string"} }	201 - Der Benutzer wurde geupdated. 400 - Der Benutzer wurde nicht gefunden, es wird eine Exception geworfen.
/user/{userID}	GET	Gibt einen Benutzer zurück Funktion: getUser()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "mail" : {"type": "string"} "age" : {"type": "int"} }	{ "id": {"type": "UUID"}, "mail": {"type": "string"}, "name": {"type": "int"} }	201 - Der Benutzer wurde zurückgegeben. 400 - Der Benutzer wurde nicht gefunden, es wird eine Exception geworfen.

/user/{userID}	DELETE	Löscht einen Benutzer Funktion: deleteUser()	{ "id" : {"type": "UUID"} }	{ "id" : {"type": "UUID"} }	201 - Der Benutzer wurde gelöscht. 400 - Der Benutzer wurde nicht gefunden, es wird eine Exception geworfen.
/movies	GET	Gibt alle Filme zurück Funktion: getAllMovies()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "dateOfRelease" : {"type": "Date"} "fsk" : {"type": "Int"} "duration" : {"type": "Int"} "genre" : {"type": "String"} }	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "dateOfRelease" : {"type": "Date"} "fsk" : {"type": "Int"} "duration" : {"type": "Int"} "genre" : {"type": "String"} }	201 - Die Liste aller Filme wurde ausgegeben.
/movies	POST	Erstellt einen Film Funktion: generateMovie()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "dateOfRelease" : {"type": "Date"} "fsk" : {"type": "Int"} "duration" : {"type": "Int"} "genre" : {"type": "String"} }	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "dateOfRelease" : {"type": "Date"} "fsk" : {"type": "Int"} "duration" : {"type": "Int"} "genre" : {"type": "String"} }	201 – Der Film wurde erstellt.
/movies/{movieID}	PATCH	Update eines Films Funktion: updateMovie()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "dateOfRelease" : {"type": "Date"} "fsk" : {"type": "Int"} "duration" : {"type": "Int"} "genre" : {"type": "String"} }	{ "id" : {"type": "UUID"} "name" : {"type": "string"} }	201 – Der Film wurde geupdated. 400 – Der Film konnte nicht gefunden werden. Eine NoSuchElementException wird ausgeworfen.

/movies/{movieID}	GET	Gibt einen Film zurück Funktion: getMovie()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "dateOfRelease" : {"type": "Date"} "fsk" : {"type": "Int"} "duration" : {"type": "Int"} "genre" : {"type": "String"} }	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "dateOfRelease" : {"type": "Date"} "fsk" : {"type": "Int"} "duration" : {"type": "Int"} "genre" : {"type": "String"} }	201 – Der Film wird ausgegeben. 400 - Der Film konnte nicht gefunden werden. Eine NoSuchElementException wird ausgeworfen.
/movies/{movieID}	DELETE	Löscht einen Film Funktion: deleteMovie()	{ "id" : {"type": "UUID"} }	{ "id" : {"type": "UUID"} }	201 – Der Film wurde gelöscht. 400 – Der Film ist nicht vorhanden und konnte daher nicht gelöscht werden. Eine NoSuchElementException wird ausgeworfen.
/favoriteMovies	GET	Gibt alle Lieblingsfilme zurück Funktion: getAll()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "content" : {"type": "MutableList"} }	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "content" : {"type": "MutableList"} }	201 - Die Liste aller Lieblingsfilme wurde zurückgegeben. 400 - Es sind keine Lieblingsfilme vorhanden, NoSuchElementException wird ausgegeben.
/favoriteMovies	POST	Erstellt einen Lieblingsfilm: Funktion: createFavMovie()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "content" : {"type": "MutableList"} }	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "content" : {"type": "MutableList"} }	201 – Der neue Lieblingsfilm wurde erstellt.
/favoriteMovies/{movieID}	PATCH	Update eines Lieblingsfilms:	{ "id" : {"type": "UUID"} "name" : {"type": "string"} }	{ "id" : {"type": "UUID"} }	201 - Der Lieblingsfilm wurde geupdated.

		Funktion: updateFavMovie()	"content" : {"type": "MutableList"} }	"name" : {"type": "string"} }	400 - Der Lieblingsfilm wurde nicht gefunden, es wird eine Exception geworfen.
/favoriteMovies/{movieID}	GET	Gibt einen Lieblingsfilm zurück Funktion: getFavMovie()	{ "id" : {"type": "UUID"} "name" : {"type": "string"} "content" : {"type": "MutableList"} }	{ "id" : {"type": "UUID"} "name" : {"type": "string"} }	201 - Der Lieblingsfilm wurde zurückgegeben. 400 - Der Lieblingsfilm wurde nicht gefunden, es wird eine Exception geworfen.
/favoriteMovies/{movieID}	DELETE	Löscht einen Lieblingsfilm Funktion: deleteFavMovie()	{ "id" : {"type": "UUID"} }	{ "id" : {"type": "UUID"} }	201 - Der Lieblingsfilm wurde gelöscht. 400 - Der Lieblingsfilm wurde nicht gefunden, es wird eine Exception geworfen.