URI	Verb	Zweck der Operation	request Daten-format	response Daten-format	request Payload	response Payload
/user/	get	alle Nutzerdaten abrufen	JSON	JSON	session_id	HTTP-Statuscode, name, uri, _id, artist_name[], artist_uri[], location
/user/{uri}	get	einen Nutzer abrufen	Query Parameter	JSON	user_uri, session_id	HTTP-Statuscode, name, uri, _id, artist_name[], artist_uri[], location
/user/	post	einen Nutzer anlegen	JSON	JSON	name, uri, _id, artist_name[], artist_uri[], location	HTTP-Statuscode, _id
/user/	delete	Löscht einen Nutzer	Query Parameter	JSON	user_uri, session_id	HTTP-Statuscode, _id
/user/	patch	Aktualisiert Nutzerdaten	Query Parameter	JSON	name, uri, _id, artist_name[], artist_uri[], location	HTTP-Statuscode, _id
/tasteScore/	get	gibt eine Liste aller TasteScore aus	JSON	JSON	session_id	HTTP-Statuscode, user1, user2, tasteScore, artists[], _id
/tasteScore/{uri}	get	zeigt alle Matches einer uri an	Query Parameter	JSON	user_uri, session_id	HTTP-Statuscode, user1, user2, tasteScore, artists[], _id
/tasteScore/	post	legt ein neues Match an	JSON	JSON	user1, user2, tasteScore, artists[]	HTTP-Statuscode, _id
/tasteScore/	patch	aktualisiert die Matches				
/tasteScore/{uri}	delete	löscht alle Datensätze einer uri	Query Parameter	JSON	user_uri, session_id	HTTP-Statuscode, _id
/tasteScores	get	vergleicht alle Nutzer und matcht sie	JSON	JSON	session_id	HTTP-Statuscode, user1, user2, tasteScore, artists[], _id
api.spotify.com/v1/me	get	Spotify Nutzerdaten abrufen	JSON	JSON	access_token	HTTP-Statuscode, username, useru
api.spotify.com/v1/me/top/artists	get	Spotify Nutzer top artists abrufen	JSON	JSON	access_token	HTTP-Statuscode, topartists_name, topartists_uri