REST-Spezifikation zu "Interne Mitfahrgelegenheit"

Moritz Basel

31. Dezember 2017

1 Einführung

1.1 Zweck des Dokuments

Dieses Dokument dient als Handbuch für Fronteddeveloper zur Software Ïnterne Mitfahrgelegenheit". Es wird versucht, die REST-API vollständig aufzuzählen und alle möglichen Anwendungsfälle zu erfassen. Dennoch besteht die Möglichkeit, dass einzelne Spezialfälle (noch) nicht abgedeckt werden. Bei Auffinden undokumentierten Verhaltens, bei Featureanfragen und Sonstigen bitten wir um Rückmeldung unter <incomplete>.

1.2 Terminologie

Endpoints bezeichnen eine spezifische (z.b. `/api/check_api` oder gruppierte (z.B. `/api/users/<uid>`) URIs

1.3 Allgemeines

Alle Endpunkte lesen und schreiben ausschließlich JSON. Der Content-Type in jedem Http-Request-Header muss als 'application/json' vorliegen.
Für Endpunkte, bei denen Autorisierung nötig ist, wird diese erbracht, indem der Http-Header 'Authorization' gesetzt ist und den Access-Token der von `/api/auth` erhalten wurde, in folgender Form enthält
'Authorization' : 'Bearer ey.....', z.B.:
'Authorization' : 'Bearer eyJhbGciOiJI.3u4zudusua.asudfuu23'

Inhaltsverzeichnis

2 JSON-Objekte

2.1 User

Die JSON-Repräsentation eines Users enthält folgende Felder:

1. username
2. email
3. phoneNumber
4. globalAdminStatus
5. id

3 User-Management

3.1 Überblick

POST /api/users	Erstellt einen neuen Benutzer
GET /api/users/<UID>	User-Profil
GET /api/users/<Username>	User-Profil
POST /api/auth	Login-Token Erstellung
PUT, UPDATE /api/users/uid	User-Update

3.2 POST /api/users

Erwartet eine vollständige JSON-Repräsentation eines Benutzers und zusätzlich das password - Feld.

3.3 POST /api/users

Erwartet eine JSON-Repräsentation eines Benutzers und erstellt diesen nach Validierung der Daten. HTTP-Status 201 bei Erfolg.

3.4 PUT,PATCH /api/users

Erwartet eine JSON-Repräsentation eines Benutzers, sowie optional ein neues Password im Feld password und überschreibt die alten Nutzerdaten.

3.5 GET /api/users/<UID>

Agiert wie erwartet. Nur der Benutzer selbst sowie Globale Administratoren haben Zugriffsrechte.

3.6 POST /api/auth

Gibt den Login-Token im JSON-Eintrag 'access_token' zurück. Dieser muss bei Zugriffen auf geschützte http-Endpunkte im Header-Feld 'Authorization' in allen Requests mitgegeben werden.
Außerdem werden alle Nutzerdaten in den Feldern 'username', 'email', 'phoneNumber', 'globalAdminStatus' versendet. Dies wird nur zur Unterstützung getan, da im JWT Body alle Felder in Base64 vorhanden sind.

4 Appointment-Funktionalitäten

4.1 Überblick

GET /api/appointments/<appointmentID>	Daten zum gewähltem Appointment

4.2 GET /api/appointments/<appointmentID>

Not yet Implemented

5 Entwicklungs-API

5.1 Überblick

DELETE /api/dev/removeUser/<uname>	Löscht bestimmten Nutzer
GET /api/dev/check_token	User-ID des Nutzers
GET /api/dev/check-api	Simpler Test
GET /api/dev/log	Log

5.2 DELETE /api/dev/removeUser/<username>

Löscht den gewählten Nutzer vollständig. Kann nur vom Nutzer selbst und von globalen Administratoren durchgeführt werden.