

AvokTech API Documentation

V 0.1.7

6/23/2015

Schemas

Account:

id: String,	<i>unique for each user</i>
firstName: String,	
lastName: String,	
email: String,	<i>functions as a username (unique)</i>
password: String,	<i>the value is hashed for security</i>
phone: String,	
photo: Object,	
paid: boolean,	
billingAddress: {	<u>Address class</u>
street: String,	
city: String,	
state: String,	
zipcode: String	
},	
creditCard: String,	<i>TBD</i>
lots: Map<String, String>,	<i>maps a lot id to the corresponding role,</i> <i>including:</i> <i>owner, manager, patrol, etc.</i>
roles: Set<String>	<i>set of systems admin roles, including:</i> <i>super, analytics, onboarding, etc.</i>

Lot:

id: String,	<i>unique for each lot</i>
name: String,	
type: String,	<i>`residential` vs. `parking lot`</i>
address: {	<u>Address class</u>
street: String,	
city: String,	
state: String,	
zipcode: String	
},	
location: double[],	<i>(lng, lat) pair</i>
photo: Object,	

rate: Map<String, double>,	<i>maps the rate type (hourly, daily, etc.) to the value of the rate</i>
rateRatio: double,	<i>scale multiplier of rate (default 1.0)</i>
capacity: int,	
occupied: int,	
available: boolean,	<i>if the lot is available (does not include reservable spaces)</i>
reservable: boolean,	
reserveMax: int,	<i>the maximum number of reservable spaces</i>
calendar: long[]	<i>a list of the weekly reservable times</i>

Reservation:

id: String,	<i>unique for each reservation</i>
accountId: String,	<i>the account associated with the reservation</i>
firstName: String,	
lastName: String,	
lotId: String,	<i>the lot associated with the reservation</i>
space: int,	<i>the space in the lot</i>
start: long,	<i>start time</i>
end: long,	<i>end time</i>
duration: int,	<i>duration of reservation (in hours)</i>
status: String	<i>the status of the reservation, including: active, cancelled, deleted, etc.</i>

LotHistory:

lotId: String,	<i>the id of the associated lot</i>
history: List<{	<i>list of <u>LotHistoryPoint</u> class</i>
time: long,	
occupied: int	
>},	

SearchHistory:

accountId: String,	<i>the id of the associated account</i>
locations: List<double[]>	<i>list of locations of searches</i>

Endpoints

Required fields are listed under the endpoint.

All `date` fields are represented in milliseconds.

GET

Account:

<code>/listAccounts</code>	<i>return all accounts</i>
<code>/accountById</code> <code>accountId</code>	<i>returns account with corresponding id</i>
<code>/accountByEmail</code> <code>email</code>	<i>returns account with corresponding email</i>
<code>/checkAccount</code> <code>email</code>	<i>returns a boolean of whether the account exists</i>
<code>/getLots</code> <code>accountId</code>	<i>returns all of the lots of the user</i>

Lot:

<code>/listLots</code>	<i>returns all lots</i>
<code>/lotById</code> <code>lotId</code>	<i>returns lot with corresponding id</i>
<code>/searchNear</code> <code>latitude</code> <code>longitude</code>	<i>returns lots within 1 mile of the location</i>

Reservation:

<code>/listReservations</code>	<i>returns all reservations</i>
<code>/reservationById</code> <code>reservationId</code>	<i>returns reservation with corresponding id</i>
<code>/reservationsByAccountId</code> <code>accountId</code>	<i>returns reservations for corresponding account</i>
<code>/reservationsByLotId</code> <code>lotId</code>	<i>returns reservations for corresponding lot</i>
<code>/activeReservationsByAccountId</code> <code>accountId</code>	<i>returns active reservations for corresponding account</i>
<code>/activeReservationsByLotId</code> <code>lotId</code>	<i>returns active reservations for corresponding lot</i>

LotHistory:

<code>/listLotHistory</code>	<i>returns all lot history</i>
<code>/lotHistory</code> <code>lotId</code>	<i>returns lot history for corresponding lot</i>

SearchHistory:

/listSearchHistory

returns all search history

/searchHistory

returns search history for corresponding account

accountId

POST

Account:

/login

handles login

email

password

/signup

handles signup and creates a new account

firstName

lastName

email

password

/addRole

adds a systems admin role to the account

accountId

role

/addLot

adds a lot and corresponding role to the account

accountId

lotId

role

Lot:

/newLot

creates a new lot

accountId

name

type

address

latitude

longitude

capacity

reserveMax

calendar

a list of the times for the calendar

/entered

increases the occupancy of the lot by 1

lotId

/exited

decreases the occupancy of the lot by 1

lotId

/changeReserveMax

changes the number of reservable spots

lotId

reserveMax

/setCalendar

adds a member to the lot

lotId

calendar

a list of the times for the calendar

Reservation:

<code>/reserve</code>	<i>creates a new reservation</i>
<code> accountId</code>	
<code> lotId</code>	
<code> start</code>	<i>start time of reservation</i>
<code> duration</code>	<i>duration of reservation (in hours)</i>

SearchHistory:

<code>/addSearchHistory</code>	<i>adds a new search history point</i>
<code> accountId</code>	
<code> latitude</code>	
<code> longitude</code>	

DELETE

Account:

<code>/deleteAccount</code>	<i>deletes account with corresponding id</i>
<code> accountId</code>	

Lot:

<code>/deleteLot</code>	<i>deletes lot with corresponding id</i>
<code> lotId</code>	

LotHistory:

<code>/deleteAllLotHistory</code>	<i>deletes all lot history</i>
<code>/deleteLotHistory</code>	<i>deletes lot history for corresponding lot</i>
<code> lotId</code>	

SearchHistory:

<code>/deleteAllSearchHistory</code>	<i>deletes all search history</i>
<code>/deleteSearchHistory</code>	<i>deletes search history for corresponding account</i>
<code> accountId</code>	