

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, 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,	

price: Map<String, double>, *maps the price type (hourly, daily, etc.) to the value of the price*
priceRatio: double, *scale multiplier of price (default 1.0)*
capacity: int,
occupied: int,
available: boolean, *if the lot is available (does not include reservable spaces)*
reservable: boolean,
reserveMax: int, *the maximum number of reservable spaces*
calendar: Set<Long> *a set of the weekly reservable times*

Reservation:

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

LotHistory:

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

SearchHistory:

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

Endpoints

Required fields are listed under the endpoint.

Optional fields are in italics.

All `date` fields are represented in milliseconds.

GET

Account:

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

Lot:

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

Reservation:

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

LotHistory:

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

SearchHistory:

/listSearchHistory
/searchHistory
 accountId

returns all search history
returns search history for corresponding account

POST

Account:

/login
 email
 password
/signup
 firstName
 lastName
 email
 password
/addRole
 accountId
 role
/addLot
 accountId
 lotId
 role

handles login (returns false if incorrect password)

handles signup and creates a new account

adds a systems admin role to the account

adds a lot and corresponding role to the account

Lot:

/newLot
 accountId
 name
 type
 address
 latitude
 longitude
 capacity
 reserveMax
 calendar
/entered
 lotId
/exited
 lotId
/changeReserveMax
 lotId
 reserveMax
/setCalendar
 lotId
 calendar

creates a new lot (returns updated account)

a list of the times for the calendar

increases the occupancy of the lot by 1

decreases the occupancy of the lot by 1

changes the number of reservable spots

adds a member to the lot

a list of the times for the calendar

Reservation:

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

SearchHistory:

/addSearchHistory	<i>adds a new search history point</i>
accountId	<i>this field is optional</i>
latitude	
longitude	

DELETE

Account:

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

Lot:

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

LotHistory:

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

SearchHistory:

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