

campaigns

index
show

customers

sign_in
sign_out
check_in
check_out
notification_token
show

locations

show
fetch_nearby
fetch_map

receipts

create
index

redemptions

index
create

social links

show
create

Authentication

Base API URL: zsdev.herokuapp.com/api/1

API Secret Key: 4s8d3e2z1s8w75ab374e4d91cb007d2e

For all requests, you **must** pass an authorization **token**.

In addition, for all requests (except sign_in), you will need to pass the customer **id** and **social_token**.

Example using CURL: curl -IH "Authorization: Token token=4reg64er6gre4grg65r4gr6e654g,id=5,social_token=12345"
zsdev.herokuapp.com/api/1/customers/profile

[Zippyspot](#) / Campaigns

campaigns

These are the coupons, rewards, and specials

GET /api/:version/locations/:id/campaigns

Retrieve all campaigns related to a given location

Examples

```
"count" => 1,
"response" => Array [
  Hash {
    "id" => ,
    "type_of" => ,#coupon, reward, special
    "title" => ,
    "discount_amount" => ,
    "discount_type" => ,#percent, amount
    "share_reward" => ,
    "image_file_name" => ,
    "schedule" => Hash { Schedule Object },
    "start_at" => ,#datetime
    "end_at" => ,#datetime
    "created_at" => ,#datetime
    "status" => ,#inactive, active, featured
    "pos" => ,#the UPC for generating barcodes
    "description" => #
  },
  # ...
]
```

Errors

· Location not found

--
..

Params

Param name

Description

required

ID of Location

Value:
Must be a number.

GET /api/:version/campaigns/:id

Retrieve a specific campaign

Examples

```
"response" => Hash {
  "id" => ,
  "type_of" => ,#coupon, reward, special
  "title" => ,
  "discount_amount" => ,
  "discount_type" => ,#percent, amount
  "share_reward" => ,
  "image_file_name" => ,
  "schedule" => Hash { Schedule Object },
  "start_at" => ,#datetime
  "end_at" => ,#datetime
  "created_at" => ,#datetime
  "status" => ,#inactive, active, featured
  "pos" => ,#the UPC for generating barcodes
  "description" => #
}
```

Errors

· Campaign not found

--
..

Params

Param name	Description
id	ID of Campaign
required	Value: Must be a number.

Customers

the people who use the mobile app

POST /api/:version/customers/sign_in

signs the customer in by updating the social_token and social_type

Examples

200 OK

Parameters

Parameter name	Description
customer_token required	Value: Must be a Hash
customer[email] required	Email address of customer Value: Must be String
customer[social_token] required	The social token returns by Facebook/Google+ Value: Must be String
customer[social_type] required	The type of social account connected Value: Must be one of: facebook, google+.

Parameter name	Description
customer[social_id] required	The social ID Value: Must be String
customer[first_name] required	Customer's first name Value: Must be String
customer[last_name] required	Customer's last name Value: Must be String
customer[social_friends] optional	An array of the customer's social friends. Example: [1,4,5,6] Value: Must be an array of any type

[Zippyspot](#) / [Customers](#) / [sign_out](#)

POST /api/:version/customers/sign_out

Signs the customer out by removing the social elements

Examples

200 OK

[Zippyspot](#) / [Customers](#) / [check_in](#)

POST /api/:version/customers/check_in

Checks in the customer to a location

Examples

```
"response" => {
  "welcome_message",          # The welcome greeting
  "points_earned",            # The amount of points earned for checking in
  "special_campaign" => {      # The exit campaign from their prior checkout (if available)
    #Campaign Meta
  }
}
```

arams

aram name	Description
ocation_id quired	The ID of the location Value: Must be a number.

[Zippyspot](#) / [Customers](#) / [check_out](#)

POST /api/:version/customers/check_out

acks OUT the customer FROM a location and returns the exit message

xamples

```
"response" => {
  "exit_message",            # The exit greeting
  "wait_time",                # The amount time until the special campaign expires
  "special_campaign" => {      # The exit campaign that can be redeemed on their next check in
    #Campaign Meta
  }
}
```

arams

aram name	Description
ocation_id quired	The ID of the location

Parameter name	Description
	Value: Must be a number.

[Zippyspot](#) / [Customers](#) / notification_token

POST /api/:version/customers/notification_token

Updates the notification_token for the current customer

Examples

200 OK

Parameters

Parameter name	Description
notification_token	The notification token to be stored
required	Value: Must be String

[Zippyspot](#) / [Customers](#) / show

GET /api/:version/customers/profile

Returns the current_customer in JSON

Examples

```
"response" => Hash {
  first_name,
  last_name,
  email,
  social_id,
  social_type,
  social_friends,
  notification_token
}
```

Locations

Retrieve individual locations of a business

GET /api/:version/beacons/:uuid/location

GET /api/:version/locations/:id

Returns a location object and includes information relevant to the current customer.

There are two ways to retrieve data for a single location, by inputting the beacon's UUID, or by inputting the location's ID. The response is different per route depending on what you're trying to do.

Examples

```
"count" => 0,
"response" => Array[
  Hash {
    "id" => ,
    "business" => Hash { Business Object },
    "title" => ,
    "address" => ,
    "address2" => ,
    "city" => ,
    "state" => ,
    "zipcode" => ,
    "latitude" => ,
    "longitude" => ,
    "status" => ,
    "points" => ,      # The current customer's points for this location
    "visits" =>        # The number of checkins for the current customer at this location
  },
  # ...
]
```

Parameters

Parameter name	Description
location_id	The location ID. Required if: Using the ID route
	Value: Must be a number.
uuid	The beacon's UUID. Required if: Using the UUID route
	Value: Must be a number.

[Zippyspot](#) / [Locations](#) / fetch_nearby

GET /api/:version/locations/near/:lat/:lon

Fetches nearby locations according to the given coordinates

Examples

```
"count" => 0,
"response" => Array[
  Hash {
    "id" => ,
    "business" => Hash { Business Object },
    "title" => ,
    "address" => ,
    "address2" => ,
    "city" => ,
    "state" => ,
    "zipcode" => ,
    "latitude" => ,
    "longitude" => ,
    "status" => ,
    "distance" =>
  },
  # ...
]
```

Parameters

Parameter name	Description
lat required	Latitude of start Value: Must be String
lon required	longitude of start Value: Must be String

[Zippyspot](#) / [Locations](#) / [fetch_map](#)

GET /api/:version/locations/map/:lat/:lon/:distance

Fetches locations in a given map square (according to distance limit)

Examples

```
"count" => 0,
"response" => Array[
  Hash {
    "id" => ,
    "business" => Hash { Business Object },
    "title" => ,
    "address" => ,
    "address2" => ,
    "city" => ,
    "state" => ,
    "zipcode" => ,
    "latitude" => ,
    "longitude" => ,
    "status" => ,
    "distance" =>
  },
  # ...
]
```

Parameters

Param name	Description
lat required	Latitude of center of map Value: Must be String
lon required	Longitude of center of map Value: Must be String
distance required	Distance from center of map (in miles) Value: Must be a number.

[Zippyspot](#) / Receipts

Receipts

Receipts are associated with redemptions and are uploaded by the customer for approval.

[Zippyspot](#) / [Receipts](#) / create

POST /api/:version/receipts

Create a new receipt with the given params

Examples

200 OK

Errors

400 Invalid receipt params were given (or invalid image).

--

..

arams

aram name	Description
ceipt quired	Value: Must be a Hash
ceipt[redemption_id] quired	The associated redemption_id Value: Must be a number.
ceipt[location_id] quired	The location_id of where the redempetion was made (MAY REMOVE THIS) Value: Must be a number.
ceipt[image] quired	The receipt image Value: Must be File.

[Zippyspot](#) / [Receipts](#) / [index](#)

ET /api/:version/receipts/:status

ows the receipts and status

xamples

```
"count" => 1,
response => Array[
  Hash {
    "id" => ,
    "purchased_on" => ,#is updated by the admin
    "amount" => ,#is 0 until admin sets amount
    "reject_reason" => ,#is blank unless rejected
    "created_at" => ,#when the receipt was uploaded initially
    "status" => ,
    "actioned_on" => ,#when the receipt was approved/rejected
```

```
"image_file_name" => ,
"location" => Hash {Location Data}
}
]
```

arams

aram name	Description
atus ntional	The status of the receipts shown Value: Must be one of: untouched, approved, rejected.

[Zippyspot](#) / Redemptions

edemptions

edemptions are what the customer created upon every redeemed campaign.

[Zippyspot](#) / [Redemptions](#) / index

ET /api/:version/redemptions

the current user's redemptions

xamples

```
"count" => 1,
response => Array[
  Hash {
    "id" => ,
    "campaign" => Hash {"title" => ""}
    "location" => Hash {"title" => ""}
  }
]
```

POST /api/:version/redemptions

the current user's redemptions

Examples

200 OK

Errors

400 Invalid redemption parameters. Redemption did not save

--
..

Parameters

Parameter name	Description
redemption_code required	Value: Must be a Hash
redemption[campaign_id] required	The campaign ID Value: Must be a number.
redemption[location_id] required	The location ID Value: Must be a number.

Share links

When a customer shares a campaign, a share link is created.

GET /api/:version/share_links/:id

Fetches the campaign via the share link (code) provided

Examples

```
response => Hash {Campaign Data}
```

Parameters

Parameter name

Description

Required

The share link's ID (code)

Value:
Must be String

POST /api/:version/share_links

Create a new share link for a campaign and return the URL

Examples

```
"response" => Hash{
  "url" => #the complete URL of the share link
}
```

Errors

! Invalid parameters. Link was not created

--
..

arams

aram name	Description
are_link quired	Value: Must be a Hash
are_link[campaign_id] quired	The ID of the campaign being shared Value: Must be a number.