

Wallet Manager Campaign APIs

Last Updated: Monday, July 13, 2020

Get Mobile Wallet Campaigns

The following retrieves an array of all the Mobile Wallet Campaigns currently active in the company.

```
GET /companies/:company_key/campaigns/wallet
```

A maximum of the 50 most recently updated records will be returned.

Request JSON

There is no Request JSON with a GET.

Return JSON

```
[
  {
    "wallet_id": "abCDefGHij",
    "name": "Gold Status",
    "type": "Loyalty",
    "smartlink_url": "http://mp.vibescm.com/c/abC123",
    "url": "/companies/:company_key/campaigns/wallet/abCDefGHij",
    "created_at": "2017-01-15T14:30Z",
    "updated_at": "2017-02-18T19:30Z"
  },
  {
    "wallet_id": "ab123",
    "name": "Silver Status",
    "type": "Loyalty",
    "smartlink_url": "http://mp.vibescm.com/c/ab1234",
    "url": "/companies/:company_key/campaigns/wallet/ab123",
    "created_at": "2017-01-15T14:30Z",
    "updated_at": "2017-02-18T19:30Z"
  }
]
```



Note: A *HTTP 404 - Not Found* error will be returned if the Mobile Wallet Campaigns cannot be located.

Get Mobile Wallet Campaign

The following will retrieve a particular Mobile Wallet Campaign by its *wallet_id*.

```
GET /companies/:company_key/campaigns/wallet/:wallet_id
```



Note: The *wallet_id* is the SmartLink token for a campaign.

Request JSON

There is no Request JSON with a GET.

Return JSON

```
{
  "wallet_id": "abCDefGHij",
  "name": "Gold Status",
  "type": "Loyalty",
  "smartlink_url": "http://mp.vibescm.com/c/abC123",
  "url": "/companies/:company_key/campaigns/wallet/abCDefGHij",
  "created_at": "2017-01-15T14:30Z",
  "updated_at": "2017-02-18T19:30Z"
}
```



Note: A *HTTP 404 - Not Found* error will be returned if the Mobile Wallet Campaign cannot be located.

Get Wallet Items

The following retrieves an array of all the Wallet items currently active for the specified Mobile Wallet Campaign.

```
GET /companies/:company_key/campaigns/wallet/:wallet_id/items?group_code=DEALS
```

Results are paginated. The 50 most recently updated records will be returned by default. However, you can provide a "page_size" parameter to get a different number of items. Each request returns a "X-Next-Page-Token" header. You can use the value of that header to return the items in the next page. An empty token means that there are no more items to iterate through.

Parameter	Description
page_size	The number of items returned. Up to 1000 items can be returned in a single page
page_token	Token identifier used for pagination.

```
GET /api/campaigns/abCDefGHij/items?page_token=jvi4jfhsk&page_size=10
```

The optional *group_code* parameter returns the active items which have the specified *group_code*. An empty array is returned if no items match the *group_code*.



Note: The *wallet_id* is the SmartLink token for a campaign.

Request JSON

There is no Request JSON with a GET.

Return JSON

```
[
  {
    "wallet_item_id": "abcdef23SDF2",
    "group_code": "DEALS",
    "campaign_ref": {
      "id": "abCDefGHij",
      "type": "Loyalty",
      "url": "/companies/:company_key/campaigns/wallet/abCDefGHij"
    },
    "active": true,
    "tokens": {
      "first_name": "John",
      "last_name": "Doe",
      "loyalty_balance": "552"
    },
    "locations": [
      {
        "latitude": 43.6867,
        "longitude": -85.0102,
        "relevant_text": "Visit our State St store"
      }
    ],
    "provider": "passbook",
    "passbook": {
      "notification": "This is a notification"
    },
    "url": "/companies/:company_key/campaigns/wallet/abCDefGHij/items/abcdef23SDF2",
    "created_at": "2017-01-15T14:30Z",
    "updated_at": "2017-02-18T19:30Z"
  },
  {
    "wallet_item_id": "ab1234",
    "campaign_ref": {
      "id": "ab123",
      "type": "Loyalty",
      "url": "/companies/:company_key/campaigns/wallet/ab123"
    },
    "active": true,
    "tokens": {
      "first_name": "Sarah",
      "last_name": "Johnson",
      "loyalty_balance": "1893"
    },
    "locations": [
      {
        "latitude": 42.6867,
        "longitude": -84.0102,
        "relevant_text": "Illinois"
      }
    ],
    "provider": "google_wallet",
    "google_wallet": {
      "messages": [
        {
          "header": "Message",
          "body": "This is a message",
          "action_uri": "http://vibescm.com",
          "image_uri": "http://www.vibes.com/sites/all/themes/vibes/images/logo.png"
        }
      ]
    },
    "url": "/companies/:company_key/campaigns/wallet/ab123/items/ab1234",
    "created_at": "2017-01-15T14:30Z",
    "updated_at": "2017-02-18T19:30Z"
  }
]
```



Note: A *404 - Not Found* object will be returned if the specified Mobile Wallet Campaigns or Mobile Wallet Campaign or Wallet items do not exist.

Filter Options for Get Wallet Items API

Wallet Items can be filtered by different attributes, and Invalid filters will be ignored.

Filter Type	Query Parameters	Description
Group Code	?group_code=code-1	Equivalent to exactly matches. "?group_code=" returns all items with NULL values.
Expiration Date	?expiration_date=2018-04-01	Equivalent to "on" Expiration date must be in ISO8601 format: YYYY-MM-DD Invalid dates will be ignored. Note that zero-padded dates are required: <ul style="list-style-type: none">• 2018-04-02: Valid• 2018-4-2: Not Valid
Provider	?provider=google	Equivalent to exactly matches
First Class Fields	?first_name=John	Equivalent to exactly matches

First Class Fields

See the [Wallet Manager - First Class Fields](#) page to get more information on how first class fields work.

Get Wallet Item

The following retrieves the specific Wallet item identified by the `:wallet_item_id`.

```
GET /companies/:company_key/campaigns/wallet/:wallet_id/items/:wallet_item_id
```



Notes:

- The `wallet_item_id` is the `uuid`, which is a campaign-specific identifier that is used to identify Wallet objects. This is generally customer defined, as it can be included in links and used before a Wallet object is actually created.
- The `wallet_id` is the SmartLink token for a campaign.

Request JSON

There is no Request JSON with a GET.

Return JSON

```

{
  "wallet_item_id": "abcdef23SDF2",
  "group_code": "DEALS",
  "campaign_ref": {
    "id": "abCDefGHij",
    "type": "Loyalty",
    "url": "/companies/:company_key/campaigns/wallet/abCDefGHij"
  },
  "active": true,
  "tokens": {
    "first_name": "John",
    "last_name": "Doe",
    "loyalty_balance": "552"
  },
  "locations": [
    {
      "latitude": 43.6867,
      "longitude": -85.0102,
      "relevant_text": "Visit our State St store"
    }
  ],
  "provider": "passbook",
  "passbook": {
    "notification": "This is a notification"
  },
  "url": "/companies/:company_key/campaigns/wallet/abCDefGHij/items/abcdef23SDF2",
  "created_at": "2017-01-15T14:30Z",
  "updated_at": "2017-02-18T19:30Z"
}

```



Note: A *404 - Not Found* object will be returned if the specified Mobile Wallet Campaign or Wallet item does not exist.

Update Mobile Wallet Item

The following will perform a net change update to an existing Wallet item.

```
PUT /companies/:company_key/campaigns/wallet/:wallet_id/items/:wallet_item_id
```

For Persons with Apple Wallet/Passbook Wallet items, the update to the Person's device will occur automatically if the application is configured to support server-pushed updates (this is controlled by the Person via the Passbook Mobile Application). The Person is also able to manually request an update via the Passbook Mobile Application. For Persons with Android Pay/Google Wallet items, the update to the Person's device will occur automatically via Google Wallet web services.



Note:
Passbook was renamed as Apple Wallet.
Google Wallet was renamed as Google Pay.

The *wallet_item_id*, *campaign_ref*, *url*, *provider*, *created_at*, and *updated_at* fields cannot be updated. Any values specified in the request body will be ignored.

Any fields that are specified will be updated with the new values. Any fields whose value is set to **Null** will be removed from the Wallet item. Any fields completely omitted will be ignored and any existing values will remain.



Note: The *wallet_item_id* is the *uuid*, which is a campaign-specific identifier that is used to identify Wallet objects. This is generally customer defined, as it can be included in links and used before a Wallet object is actually created.

In the example below, we will demonstrate how to update a particular wallet pass field, send an apple wallet notification, and update the copy on a relevant location notification:

Request JSON

```

{
  "active": true,
  "group_code": "DEALS",
  "tokens": {
    "loyalty_balance": "952",
    "barcode_value": "494dd709-8cd1-4dfc-8886-04c8bb5aad12"
  },
  "locations": [
    {
      "latitude": 43.6867,
      "longitude": -85.0102,
      "relevant_text": "Redeem your points @ Sub Shop on Adams St!"
    }
  ],
  "passbook": {
    "notification": "Your updated balance = 952!"
  }
}

```

This particular update would change the value visible on the wallet pass for the field "loyalty_balance" to "952", and it would also send a notification to the wallet pass holder notifying them of the updated balance value if they have an apple wallet pass (these types of notifications are not possible on Google Pay - hence why we specify "passbook").

This update would also change the language on a relevant location notification for this wallet pass.

A relevant location (for both Google Pay and Apple Wallet) allow notifications to surface on the user's lock screen when they enter within range of those lat/long coordinates listed in the payload so that the pass is easier to access. These location specific notifications can be generic for all pass holders, but can also have unique content per individual passes (perhaps singling out a particular reward a customer has earned, or reminding them about an event they had expressed interest in).

Update Dynamic Strip Image

You can also change the strip image for both Google Pay and Apple Wallet using our API. Just like updating the token field value for "loyalty_balance", simply specify the token field name "IMAGE_CODE" (all caps), and use the image file name you have provided Vibes (without the file type extension).

In the example below, we are making the same call as the example above, however we have are changing the IMAGE_CODE field value to insert a new holiday themed rewards image just for this particular pass holder. The original image file would have been named "medium_tier_rewards_holiday.png", notice that in the payload we are not including ".png".

Request JSON

```

{
  "active": true,
  "group_code": "DEALS",
  "tokens": {
    "IMAGE_CODE": "medium_tier_rewards_holiday",
    "loyalty_balance": "952"
  },
  "locations": [
    {
      "latitude": 43.6867,
      "longitude": -85.0102,
      "relevant_text": "Redeem your points @ Sub Shop on Adams St!"
    }
  ],
  "passbook": {
    "notification": "Your updated balance = 952!"
  }
}

```

Please contact your customer care representative or our help desk if you have any campaigns you would like to begin using dynamic strip images on!



Note: The only fields that will be updated on the Wallet object are: *tokens*, *locations*, and *group_code*.

Return JSON

A Wallet item entity reflecting the new state of the item.



Notes:

- An *HTTP 200* status code means that an OK result will be returned if the operation completes successfully.
- An *HTTP 404 - Not Found* error will be returned if the specified Mobile Wallet Campaign or Wallet item does not exist.