

# Overview

All API requests are sent to talech ver HTTPS (<https://mapi.talech.com>). Request/response payload are JSON encoded. During authentication, a security token (access token) will be returned, which should be included in the HTTPS header for all subsequent requests.

## Authentication

In order to authenticate your service with our API, we must first set up an account for you. We will provide you with a *client ID*, *client secret* and *merchant token (refresh token)* which you can use to log into the merchant account without using the account's username and password.

### Request

**POST:** `o/oauth2/token`

### Sample Input

```
{  
  "grant_type": "refresh_token",  
  "client_id": "MY_CLIENT_ID",  
  "client_secret": "MY_CLIENT_SECRET",  
  "refresh_token": "MERCHANT_TOKEN"  
}
```

### Sample Output

```
{  
  "access_token": "76486/PfFMMcvQxH2x0fIW7LfBI2PR",  
  "expires_in": 3600,  
  "token_type": "Bearer"  
}
```

You will need to pass the *access\_token* in HTTPS "securityToken" header to any API calls you make that deal with merchant data, e.g.:

securityToken: 76486/PfFMMcvQxH2x0fIW7LfBI2PR

The token will automatically expire after being idle for about 30 minutes, but as long as you make an API call within that time frame, the API response will often contain a refreshed new token for you to use, without needing to authenticate again.

Here is some example PHP code on how to parse out the new security token:

```
$response = curl_exec($curl); // response from API call

list($headers, $body) = explode("\r\n\r\n", $response, 2);
while (strpos($headers, "100 Continue") !== false){
    list($headers, $body) = explode("\r\n\r\n", $body, 2);
}

if ($headers) {
    $headers = explode("\r\n", $headers);
    foreach($headers as $header) {
        $header = explode(':', $header, 2);
        if (count($header) === 2 && $header[0] === 'X-POS-SecurityToken') {
            error_log('Found new security token: ' . $header[1]);
            //Save this new token
        }
    }
}

// Response of API call is in $body
```

## Multi-location Support

If the merchant has multiple stores/locations with talech, you need to make [authentication/getAllMerchantStoreInfo](#) (Get All Merchant Store Info) API call first to get its store/location list, each of which is uniquely identified by *merchantIdentification*. In subsequent API calls, pass *merchantIdentification* in HTTPS “X-POS-MerchantId” header for store/location specific data, e.g.:

```
X-POS-MerchantId: 643987
```

# Get All Merchant Store Info

## Request

**POST:** <https://api.talech.com/authentication/getAllMerchantStoreInfo>

## Sample Input

```
{
}
```

## Sample Output

```
{
  "ResponseCode": {
    "desc": "OK",
    "statusCode": 200,
    "totalCnt": 0
  },
  "merchantStoreDetails": {
    "expiresIn": 0,
    "merchantDetails": [
      {
        "accountSetup": true,
        "activationDate": "11/11/2014 00:01:09",
        "agentInfo": "Tom Cruise",
        "applicationReceivedDate": "11/11/2014 00:01:09",
        "billingInfos": [],
        "businessDescription": "asasasasa adada",
        "contractSigned": true,
        "contractSignedDate": "07/01/2013 03:00:00",
        "defaultTimeZone": "Asia/Singapore",
        "id": 382,
        "industryTypeCode": "QuickServe",
        "isDemo": true,
        "legalBusinessTO": null,
        "merchantBoarded": false,
        "merchantIdentification": 746027,
        "merchantLogo": "",
        "premiumFeatures": [
          {
            "activationDate": "",
            "featureType": "ONLINE_ORDERING",

```

```
"id": 49,
"isActive": true,
"isDeleted": false,
"merchantId": 382,
"metaData": "",
"provider": 1,
"storeId": 1
},
{
  "activationDate": "",
  "featureType": "PO",
  "id": 62,
  "isActive": true,
  "isDeleted": false,
  "merchantId": 382,
  "metaData": "",
  "provider": 1,
  "storeId": 1
}
],
"prevStatus": null,
"salesAgent": 45,
"signupIPAddress": "",
"socialNetworkURLs": {
  "twitURL": "",
  "instURL": "",
  "fbURL": "http://example.com",
  "gURL": "",
  "ytbeURL": "",
  "yelpURL": ""
},
"statusChangeDate": "11/11/2014 00:01:09",
"stores": [
  {
    "address": {
      "address1": "dsds",
      "address2": "sdddd",
      "city": "LA",
      "country": "Ireland",
      "id": 196,
      "parentId": 0,
      "state": "CA",
      "zip": ""
    },
    "businessCloseHour": "04:00:00",
    "businessEmail": "xyz@t.com",
    "businessName": "Cafe bugs",
    "businessPhone": "55555",
    "currency": "",
    "deliveryAddress": [
      {
        "address1": "Palo3",
```

```
"address2": "Altaso",
"city": "San diekgva",
"country": "United States",
"id": 265,
"parentId": 0,
"state": "As",
"zip": "52455"
},
{
  "address1": "Palo ass",
  "address2": "Altaso",
  "city": "San diekgva",
  "country": "United States",
  "id": 267,
  "parentId": 0,
  "state": "As",
  "zip": "52455"
}
],
"deliveryHours": [
],
"equipmentShipped": true,
"gatewayInfo": {
  "active": true,
  "id": 338,
  "isDemo": false,
  "merchantAccountId": "",
  "merchantId": 382,
  "paymentsProcessorId": 3,
  "processorCode": "VM",
  "storeId": 1
},
"gatewayProvisioned": false,
"goliveDate": "",
"id": 195,
"locale": "en_IE",
"marketingMessage": "Thank you for shopping with us.",
"merchantId": 382,
"mid": "",
"note": "this is a test",
"prevStatus": "",
"setupDone": false,
"statusChangeDate": "11/11/2014 00:01:09",
"storeHours": [
  {
    "closeTimeAtAM": "03:00:00",
    "closeTimeAtPM": "11:00:00",
    "day": 1,
    "id": 1,
    "isOpen": true,
    "openTimeAtAM": "09:00:00",
    "openTimeAtPM": "04:00:00"
  }
]
```

```
    }
    ],
    "storeId": 1,
    "storeName": "my store2",
    "vatId": ""
  }
],
"tosAccepted": "Declined",
"trialEndDate": "09/01/2013 03:00:00",
"trialStartDate": "08/01/2013 03:00:00",
"website": "http://example.com",
"cashDrawerInfo": {
  "isCashDrawerMgmtEnable": true
}
}],
"securityToken": "10000003/uobgEhji4SLvUfaxtHmiEQh3",
"supportedPremiumFeatures": [
  "PO",
  "INVENTORY_ALERT",
  "INVENTORY_LOG",
  "STOCK_TAKE"
],
"user": {
  "allowedHourWeekly": "",
  "allowedOp": "",
  "changePasswordOnLogin": false,
  "displayTrackings": [],
  "email": "sxzhr12ixa@aajaia.com",
  "failedLoginAttempts": 4,
  "hPin": "D/4avRoIIvNTwjPW4AlhPpXuxCU4Mqdhryj/N6xaFQw=",
  "hasPin": 1,
  "id": 491,
  "isActive": true,
  "isClockInReqToAccess": false,
  "isClockIn": false,
  "merchantId": 382,
  "merchantStoreAccess": [
    {
      "allowedOp": "",
      "id": 932,
      "isClockInReqToAccess": false,
      "isDeleted": 0,
      "merchantId": 382,
      "storeId": 0,
      "userId": 491,
      "usrGroupCode": "Admin"
    }
  ]
},
"name": "",
"oldPassword": "",
"ownedBy": 0,
```

```
"resetDate": "",
"storeId": 1,
"superUser": false,
"userFirstName": "Jeni",
"userLastName": "milton",
"userName": "Jeni milton",
"usrGroupCode": "Admin"
}
}
}
```

## Error Codes

```
"ResponseCode": {
  "desc": "Unable to process the request",
  "statusCode": 6000
}
"ResponseCode": {
  "desc": "Records not found",
  "statusCode": 6005
}
```

## Notes

- Retrieves all merchantStoreDetails accessible by current user in merchantDetails[]

# Get All Customers

## Request

**POST:** managecustomer/getallcustomers

## Sample Input

```
{
  "offset":0,
  "limit":10
}
```

## Sample Output

```
{
  "customers":[
    {
      "addIPhoneNumber1":"","
      "addIPhoneNumber2":"","
      "address":null,
      "createdBy":129,
      "createdOn":"10/03/2012 10:02:34",
      "customerLoyaltyDetails":[
        {
          "customerId":80,
          "discountId":161,
          "id":66,
          "loyaltyAttachedDate":"10/05/2012 07:44:35",
          "loyaltyId":4,
          "loyaltyName":"Test Loyalty",
          "loyaltyRedeemedDate":"","
          "redeemed":false,
          "redemptionValue":10.5,
          "visitCount":3,
          "visitToRedeem":2
        }
      ],
      "customerVisitDetails":{"
        "averageSpend":4.92,
        "lastSpend":9,
        "lastVisit":156,
        "lifeTimeVisit":4,
      }
    }
  ]
}
```



```

        "visitThisYear":4
    },
    "deleted":false,
    "email":"testscos@test.com",
    "firstName":"Test",
    "id":80,
    "lastName":"Costumer",
    "merchantId":63,
    "modifiedBy":129,
    "modifiedOn":"10/04/2012 08:43:49",
    "phoneNumber":"9723456258",
    "userComment":"Some comment"
  },
  {
    "addIPhoneNumber1":"","
    "addIPhoneNumber2":"","
    "address":null,
    "createdBy":129,
    "createdOn":"01/14/2013 14:30:06",
    "customerLoyaltyDetails":[],
    "customerVisitDetails":{
      "averageSpend":0,
      "lastSpend":0,
      "lastVisit":0,
      "lifeTimeVisit":0,
      "visitThisYear":0
    },
    "deleted":false,
    "email":"irv.henderson@gmail.com",
    "firstName":"Irv",
    "id":136,
    "lastName":"Henderson",
    "merchantId":63,
    "modifiedBy":129,
    "modifiedOn":"04/09/2013 05:49:48",
    "phoneNumber":"9148435555",
    "userComment":"Hello!"
  },
  {
    "addIPhoneNumber1":"","
    "addIPhoneNumber2":"","
    "address":null,
    "createdBy":129,
    "createdOn":"01/14/2013 15:03:10",
    "customerLoyaltyDetails":[],
    "customerVisitDetails":{
      "averageSpend":32,
      "lastSpend":50,

```

```
        "lastVisit":85,  
        "lifeTimeVisit":2,  
        "visitThisYear":2  
    },  
    "deleted":false,  
    "email":"joy@talech.com",  
    "firstName":"Jojo",  
    "id":137,  
    "lastName":"Smith",  
    "merchantId":63,  
    "modifiedBy":129,  
    "modifiedOn":"01/14/2013 15:03:10",  
    "phoneNumber":"6505551213",  
    "userComment":""  
    },  
    ],  
    "ResponseCode":{"desc":"OK","statusCode":200}  
}
```

# Get Customer Details by ID

## Request

**POST:** managecustomer/getcustomerdetails

## Sample Input

```
{
  "id":15
}
```

## Sample Output

```
{
  "customer":{
    "addlPhoneNumber1":"","
    "addlPhoneNumber2":"","
    "address":{
      "address1":"","
      "address2":"","
      "city":"","
      "country":"United States",
      "state":"","
      "zip":""
    },
    "createdBy":129,
    "createdOn":"06/24/2013 01:23:36",
    "customerLoyaltyDetails":[],
    "customerVisitDetails":{
      "averageSpend":29.28,
      "lastSpend":36.6,
      "lastVisit":58,
      "lifeTimeVisit":10,
      "visitThisYear":10
    },
    "deleted":false,
    "email":"tom@for-each.com",
    "firstName":"Tom",
    "id":382,
    "lastName":"Winter",
    "merchantId":63,
    "modifiedBy":129,
    "modifiedOn":"06/24/2013 01:23:36",
    "phoneNumber":"","
    "userComment":""
  },
}
```

```
    "ResponseCode":{"desc":"OK","statusCode":200}  
  }
```

## Error Codes

```
"ResponseCode": {  
  "desc": "Unable to process the request",  
  "statusCode": 6000  
}  
"ResponseCode": {  
  "desc": "Records not found",  
  "statusCode": 6005  
}  
"ResponseCode": {  
  "desc": "Customer not found",  
  "statusCode": 3016  
}
```

# Add Customer

## Request

**PUT:** managecustomer/addcustomer

## Sample Input

```
{
  "customer":{
    "firstName":"Hello",
    "lastName":"world",
    "email":"asdds@gmail.com",
    "phoneNumber":"8861218546",
    "addIPhoneNumber1":"8861218548",
    "addIPhoneNumber2":"88-61218548",

    "userComment":"commentTest",
    "address":{
      "address1":"dsds",
      "address2":"sdddd",
      "city":"LA",
      "state":"CA",
      "country":"United States"
    }
  }
}
```

## Sample Output

```
{
  "customerId":18,
  "ResponseCode":{"desc":"OK","statusCode":200}
}
```

## Error Codes

```
"ResponseCode": {
  "desc": "Unable to process the request",
  "statusCode": 6000
}
"ResponseCode": {
  "desc": "Customer already exists with same email or name",
  "statusCode": 3022
}
```

# Edit Customer

## Request

**PUT:** managecustomer/editcustomer

## Sample Input

```
{
  "customer":{
    "id":15,
    "firstName":"shri",
    "lastName":"Good",
    "phoneNumber":"8861218548",
    "addlPhoneNumber1":"8861218548",
    "addlPhoneNumber2":"88-61218548",
    "userComment":"commentTest",
    "address":{
      "address1":"Dtive",
      "address2":"yu",
      "city":"PALO ALTO",
      "state":"CA"
    }
  }
}
```

## Sample Output

```
{"ResponseCode":{"desc":"OK","statusCode":200}}
```

## Error Codes

```
"ResponseCode": {
  "desc": "Unable to process the request",
  "statusCode": 6000
}
"ResponseCode": {
  "desc": "Customer does not exist",
  "statusCode": 3018
}
"ResponseCode": {
  "desc": "Customer id required",
  "statusCode": 3017
}
"ResponseCode": {
  "desc": "Invalid Email",
  "statusCode": 2002
}
```

# Fetch Order History

## Request

**POST:** order/getorderhistory

## Sample Input

```
{
  "searchCriteria":{
    "offset":0,
    "limit":100,
    "searchString": "",
    "startDate":"01/01/2013 00:00:00",
    "endDate":"01/08/2013 00:00:00",
    "paymentTypes": [ "CCRD" ],
    "orderStatus": [ "PAID", "PARTIAL_PAID" ],
    "transactionTypes" : [ "SALE" ],
    "storeIds": [1],
    "orderBy": 'firstPaymentDate',
    "userId": 344,
  }
}
```

## Sample Output

```
{
  "searchResult": {
    "orders": [
      {
        "amount": 22.1,
        "customerName": " ",
        "discount": 0,
        "discountCnt": 0,
        "listOfItems": "Bud Light",
        "loyalty": 0,
        "loyaltyCnt": 0,
        "noOfItems": 1,
        "numPayments": 2,
        "orderBusinessDate": "05/07/2013 00:00:00",
        "orderClosedBusinessDate": "05/07/2013 00:00:00",
        "orderClosedDate": "05/07/2013 16:15:23",
        "orderDate": "05/07/2013 16:14:05",
        "orderFirstPaymentBusinessDate": "05/07/2013 00:00:00",
      }
    ]
  }
}
```

```
"orderFirstPaymentDate": "05/07/2013 16:14:58",
"orderId": 6361,
"orderLastPaymentBusinessDate": "05/07/2013 00:00:00",
"orderLastPaymentDate": "05/07/2013 16:15:23",
"orderStatus": "PAID",
"payAmount": 0,
"payInfold": 6361,
"paymentInfo": [
  {
    "authCode": "514PNI",
    "errorCode": "",
    "errorMessage": "",
    "id": 6891,
    "orderId": 6361,
    "paymentAmount": 6.05,
    "paymentBusinessDate": "05/07/2013 00:00:00",
    "paymentDate": "05/07/2013 16:15:23",
    "paymentType": "Credit",
    "refundHistoryId": 0,
    "settlementStatus": "AUTHORIZED",
    "storeId": 0,
    "tipAmount": 3.4,
    "transactionCode": "",
    "transactionType": "SALE",
    "userFirstName": "",
    "userLastName": ""
  },
  {
    "authCode": "320PNI",
    "errorCode": "",
    "errorMessage": "",
    "id": 6890,
    "orderId": 6361,
    "paymentAmount": 6.05,
    "paymentBusinessDate": "05/07/2013 00:00:00",
    "paymentDate": "05/07/2013 16:14:58",
    "paymentType": "Credit",
    "refundHistoryId": 0,
    "settlementStatus": "AUTHORIZED",
    "storeId": 0,
    "tipAmount": 3.4,
    "transactionCode": "",
    "transactionType": "SALE",
    "userFirstName": "",
    "userLastName": ""
  }
],
"paymentType": "SPLITCHECK",
"refundAmount": 0,
"refundNumber": "",
"refundRevenue": 0,
"revenue": 17.000000000000004,
"subTotal": 17,
"tableNo": ""
```



```
"tax": 1.7,
"tip": 3.4,
"transactionCode": "7E6T",
"userFirstName": "Julia",
"userLastName": "Roberts"
},
{
"amount": 48.08,
"customerName": "Gina Gagnon",
"discount": 43.25,
"discountCnt": 1,
"listOfItems": "Coffee",
"loyalty": 0,
"loyaltyCnt": 0,
"noOfItems": 1,
"numPayments": 1,
"orderBusinessDate": "03/25/2013 00:00:00",
"orderClosedBusinessDate": "03/25/2013 00:00:00",
"orderClosedDate": "03/25/2013 13:41:16",
"orderDate": "03/25/2013 13:41:08",
"orderFirstPaymentBusinessDate": "03/25/2013 00:00:00",
"orderFirstPaymentDate": "03/25/2013 13:41:16",
"orderId": 5830,
"orderLastPaymentBusinessDate": "03/25/2013 00:00:00",
"orderLastPaymentDate": "03/25/2013 13:41:16",
"orderStatus": "PAID",
"payAmount": 0,
"payInfold": 5830,
"paymentInfo": [
{
"authCode": "000PNI",
"errorCode": "",
"errorMessage": "",
"id": 6621,
"orderId": 5830,
"paymentAmount": 48.08,
"paymentBusinessDate": "03/25/2013 00:00:00",
"paymentDate": "03/25/2013 13:41:16",
"paymentType": "Credit",
"refundHistoryId": 0,
"settlementStatus": "AUTHORIZED",
"storeId": 0,
"tipAmount": 0,
"transactionCode": "",
"transactionType": "SALE",
"userFirstName": "",
"userLastName": ""
}
],
"paymentType": "CCRD",
"refundAmount": 0,
"refundNumber": "",
"refundRevenue": 0,
"revenue": 43.25,
```

```
"subTotal": 116.5,
"tableNo": "",
"tax": 4.83,
"tip": 0,
"transactionCode": "N6VP",
"userFirstName": "Julia",
"userLastName": "Roberts"
},
{
  "amount": 48,
  "customerName": " ",
  "discount": 0,
  "discountCnt": 0,
  "listOfItems": "Green Salad",
  "loyalty": 0,
  "loyaltyCnt": 0,
  "noOfItems": 1,
  "numPayments": 1,
  "orderBusinessDate": "03/13/2013 00:00:00",
  "orderClosedBusinessDate": "03/13/2013 00:00:00",
  "orderClosedDate": "03/13/2013 13:09:23",
  "orderDate": "03/13/2013 13:09:13",
  "orderFirstPaymentBusinessDate": "03/13/2013 00:00:00",
  "orderFirstPaymentDate": "03/13/2013 13:09:23",
  "orderId": 5612,
  "orderLastPaymentBusinessDate": "03/13/2013 00:00:00",
  "orderLastPaymentDate": "03/13/2013 13:09:23",
  "orderStatus": "PAID",
  "payAmount": 0,
  "payInfold": 5612,
  "paymentInfo": [
    {
      "authCode": "667PNI",
      "errorCode": "",
      "errorMessage": "",
      "id": 6411,
      "orderId": 5612,
      "paymentAmount": 48,
      "paymentBusinessDate": "03/13/2013 00:00:00",
      "paymentDate": "03/13/2013 13:09:23",
      "paymentType": "Credit",
      "refundHistoryId": 0,
      "settlementStatus": "AUTHORIZED",
      "storeId": 0,
      "tipAmount": 0,
      "transactionCode": "",
      "transactionType": "SALE",
      "userFirstName": "",
      "userLastName": ""
    }
  ],
  "paymentType": "CCRD",
  "refundAmount": 8,
  "refundNumber": ""
}
```

```

    "refundRevenue": 8,
    "revenue": 48,
    "subTotal": 48,
    "tableNo": "Irv 4",
    "tax": 0,
    "tip": 0,
    "transactionCode": "9NIL",
    "userFirstName": "Julia",
    "userLastName": "Roberts"
  }
],
"totalRecords": 3
},
"ResponseCode": {
  "desc": "OK",
  "statusCode": 200
}
}

```

## Error Codes

```

"ResponseCode": {
  "desc": "Unable to process the request",
  "statusCode": 6000
}
"ResponseCode": {
  "desc": "Records not found",
  "statusCode": 6005
}

```

### Notes:

- startDate and endDate, searchString are optional. If neither startDate and endDate is given, all the order from day 1 will be returned
- supported filters
  - multiple selection is allowed
    - "paymentTypes"
      - *CASH , CCRD , DCRD , CHECK*
    - "orderStatus"
      - *ALL, OPEN, PARTIAL\_PAID, PAID*
    - "transactionTypes"
      - *SALE, VOID, REFUND, PARTIALREFUND, PARTIALVOID, UPDATETIP;*
    - "storeIds"
  - single value
    - userId
    - hideSale: boolean

- only show refund transactions
- hideRefund: boolean
  - only show sales transactions
- onlySplitCheck: boolean
  - only show SplitCheck orders

# Get Order Details by ID

## Request

**POST:** order/getorderbyid

## Sample Input

```
{  
  "orderId": 6167  
}
```

## Sample Output

```
{  
  "Order": {  
    "closedBy": 0,  
    "closedUserName": "",  
    "createdBy": 33,  
    "createdUserName": "Pos3 H.",  
    "customer": null,  
    "customerFirstName": "",  
    "customerId": 0,  
    "customerLastName": "",  
    "discountId": 0,  
    "globalDiscountAmt": 0,  
    "globalDiscountName": "",  
    "globalDiscountType": "NONE",  
    "globalDiscountValue": 0,  
    "id": 6167,  
    "merchantId": 24,  
    "modifiedBy": 33,  
    "modifiedUserName": "Pos3 H.",  
    "orderAlreadyFullPaid": false,  
    "orderAmount": 559.9000000000001,  
    "orderClosedOn": "",  
    "orderCreationTime": "04/15/2013 05:55:43",  
    "orderDetails": [  
      {  
        "addOns": [],  
        "baseProductId": 0,  
        "categoryId": 1793,  
        "comments": "",  
        "deleted": false,  
        "discountAmt": 0,  
        "discountCategory": 0,  
        "discountId": 0,  
        "discountName": "",  
        "discountType": "NONE",  
        "discountValue": 0,  
        "id": 0,  
        "productId": 0,  
        "quantity": 1,  
        "unitPrice": 559.9000000000001,  
        "userId": 33,  
        "userName": "Pos3 H.",  
        "createdTime": "04/15/2013 05:55:43",  
        "modifiedTime": "04/15/2013 05:55:43",  
        "deletedTime": ""  
      }  
    ]  
  }  
}
```

```
"discountType": "",
"discountValue": 0,
"globalDiscountAmt": 0,
"id": 16059,
"merchantId": 24,
"orderId": 6167,
"parentOrderDetailId": 0,
"price": 352,
"productId": 201278,
"productName": "AKHA DIAMOND CARRY ALL",
"quantity": 1,
"refundedPrice": 0,
"refundedQuantity": 0,
"sequence": 0,
"storeId": 1,
"tax": 0
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 2011,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16061,
  "merchantId": 24,
  "orderId": 6167,
  "parentOrderDetailId": 0,
  "price": 12.3,
  "productId": 201164,
  "productName": "Ring",
  "quantity": 1,
  "refundedPrice": 0,
  "refundedQuantity": 0,
  "sequence": 1,
  "storeId": 1,
  "tax": 0
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 2011,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
```

```
"globalDiscountAmt": 0,
"id": 16060,
"merchantId": 24,
"orderId": 6167,
"parentOrderDetailId": 0,
"price": 12.3,
"productId": 201163,
"productName": "Scarf",
"quantity": 1,
"refundedPrice": 0,
"refundedQuantity": 0,
"sequence": 2,
"storeId": 1,
"tax": 0
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 1793,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16058,
  "merchantId": 24,
  "orderId": 6167,
  "parentOrderDetailId": 0,
  "price": 189,
  "productId": 186579,
  "productName": "AKHA DIAMOND MESSENGER",
  "quantity": 1,
  "refundedPrice": 0,
  "refundedQuantity": 0,
  "sequence": 3,
  "storeId": 1,
  "tax": 18.9
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 0,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16074,
```

```
"merchantId": 24,
"orderId": 6167,
"parentOrderDetailId": 0,
"price": 6,
"productId": 191226,
"productName": "Custom Item",
"quantity": 0,
"refundedPrice": 6,
"refundedQuantity": 1,
"sequence": 4,
"storeId": 1,
"tax": 0.6
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 2011,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16064,
  "merchantId": 24,
  "orderId": 6167,
  "parentOrderDetailId": 0,
  "price": 12.3,
  "productId": 201164,
  "productName": "Ring",
  "quantity": 1,
  "refundedPrice": 0,
  "refundedQuantity": 0,
  "sequence": 5,
  "storeId": 1,
  "tax": 0
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 0,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16100,
  "merchantId": 24,
  "orderId": 6167,
```



```
"parentOrderDetailId": 0,
"price": 6,
"productId": 191226,
"productName": "Custom Item",
"quantity": 1,
"refundedPrice": 0,
"refundedQuantity": 0,
"sequence": 5,
"storeId": 1,
"tax": 0.6
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 0,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16073,
  "merchantId": 24,
  "orderId": 6167,
  "parentOrderDetailId": 0,
  "price": 123,
  "productId": 191226,
  "productName": "Custom Item",
  "quantity": 1,
  "refundedPrice": 0,
  "refundedQuantity": 0,
  "sequence": 6,
  "storeId": 1,
  "tax": 12.3
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 0,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16099,
  "merchantId": 24,
  "orderId": 6167,
  "parentOrderDetailId": 0,
  "price": 123,
```

```
"productId": 191226,
"productName": "Custom Item",
"quantity": 1,
"refundedPrice": 0,
"refundedQuantity": 0,
"sequence": 6,
"storeId": 1,
"tax": 12.3
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 0,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16087,
  "merchantId": 24,
  "orderId": 6167,
  "parentOrderDetailId": 0,
  "price": 6,
  "productId": 191226,
  "productName": "Custom Item",
  "quantity": 1,
  "refundedPrice": 0,
  "refundedQuantity": 0,
  "sequence": 7,
  "storeId": 1,
  "tax": 0.6
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 2011,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16063,
  "merchantId": 24,
  "orderId": 6167,
  "parentOrderDetailId": 0,
  "price": 12.3,
  "productId": 201163,
  "productName": "Scarf",
```

```
"quantity": 1,
"refundedPrice": 0,
"refundedQuantity": 0,
"sequence": 8,
"storeId": 1,
"tax": 0
},
{
  "addOns": [],
  "baseProductId": 0,
  "categoryId": 0,
  "comments": "",
  "deleted": false,
  "discountAmt": 0,
  "discountCategory": 0,
  "discountId": 0,
  "discountType": "",
  "discountValue": 0,
  "globalDiscountAmt": 0,
  "id": 16086,
  "merchantId": 24,
  "orderId": 6167,
  "parentOrderDetailId": 0,
  "price": 123,
  "productId": 191226,
  "productName": "Custom Item",
  "quantity": 1,
  "refundedPrice": 0,
  "refundedQuantity": 0,
  "sequence": 9,
  "storeId": 1,
  "tax": 12.3
}
],
"orderModificationTime": "04/16/2013 00:54:56",
"orderStatus": "PARTIAL_REFUNDED",
"orderSubTotal": 541,
"orderType": "tostay",
"paidAmount": 0,
"paymentDetails"[
  {
    "authCode": "",
    "cardData": "",
    "cardHolderFirstName": "",
    "cardHolderLastName": "",
    "cardNumber": "",
    "changeDue": 0,
    "createdBy": 33,
    "id": 6760,
    "merchantId": 24,
    "orderId": 6167,
    "paymentAmount": 892.9,
```

```

        "paymentDate": "04/14/2013 21:25:37",
        "paymentStatus": "SUCCESS",
        "paymentTransactionId": 0,
        "paymentType": "CASH",
        "settlementStatus": "SETTLED",
        "signature": "",
        "storeId": 1,
        "tipAmount": 0,
        "transactionCode": "RTAD",
        "transactionType": "SALE",
        "userFirstName": "Pos3",
        "userLastName": "Henderson"
    },
    {
        "authCode": "CVI928",
        "cardData": "",
        "cardHolderFirstName": "BoB",
        "cardHolderLastName": "smith",
        "cardNumber": "1111",
        "changeDue": 0,
        "createdBy": 230,
        "id": 490,
        "merchantId": 194,
        "orderId": 299,
        "paymentAmount": 100,
        "paymentDate": "03/21/2013 01:20:56",
        "paymentStatus": "SUCCESS",
        "paymentTransactionId": 245,
        "paymentType": "CCRD",
        "settlementStatus": "AUTHORIZED",
        "signature": "",
        "tipAmount": 1,
        "transactionCode": "VZC9",
        "transactionType": "SALE",
        "userFirstName": "JenifferMoses",
        "userLastName": ""
    }
],
"customer": {
    "createdBy": 230,
    "createdOn": "01/22/2013

```

14:42:56",

```

"customerLoyaltyDetails": [],
"customerVisitDetails": null,
"deleted": false, "email": "shri@gmail.com",
"firstName": "shri",
"id": 13,
"lastName": "Good",
"merchantId": 194,
"modifiedBy": 230,
"modifiedOn": "01/23/2013
16:45:57",
"phoneNumber": "8861218546",
"userComment": "commentTest"
    },
    "refundAmount": 13.3,
    "refundedDiscountAmt": 0,
    "refundedSubTotal": 12.3,
    "refundedTaxAmount": 1,
    "refundedTotalDiscountAmt": 0,
    "storeId": 1,
    "tableNo": "",
    "taxAmount": 18.9,
    "taxPercentage": 10,
    "tipsAmount": 0,
    "totalDiscount": 0,
    "totalDiscountAmt": 0,
    "transactionCode": "RTAD"
},
"ResponseCode": {
    "desc": "OK",
    "statusCode": 200
},
}

```

## Error Codes

```

"ResponseCode": {
    "desc": "Unable to process the request",
    "statusCode": 6000
}
"ResponseCode": {
    "desc": "Records not found",
    "statusCode": 6005
}

```

```
}
```

```
"ResponseCode": {  
  "desc": "Access Denied - Invalid Security Token",  
  "statusCode": 6002  
}
```

```
"ResponseCode": {  
  "desc": "Order Not Found",  
  "statusCode": 4003  
}
```

# Get All Menu Items

## Request

**POST:** `managemenu/menuitem/allmenuitems`

## Sample Input

```
{  
  "storeId": 1,  
  "offset" : 0,  
  "maxRecords": 30,  
  "searchString": "ONIONTEST",  
  "inventoryOnly": false,  
  "categoryId": 123  
}
```

## Sample Output

```
{  
  "items": [  
    {  
      "baseProductId": 0,  
      "categoryId": 1775,  
      "categoryType": "Appetizers",  
      "color": "Orange",  
      "cost": 2,  
      "dineOut": true,  
      "id": 177973,  
      "inventory": true,  
      "item": "APP105",  
      "merchantId": 167,  
      "merchantStoreProducts": [  
        {  
          "dineOut": true,  
          "discountId": 0,  
          "id": 0,  
          "merchantId": 167,  
          "partNumber": "TIN9YHEY1",  
          "positionNumber": 5,  
          "price": 6,  
          "productId": 0,  
          "quantity": 71,  
        }  
      ]  
    }  
  ]  
}
```

```
    "salesTax": 0,
    "salesTaxOn": false,
    "storeId": 1
  }
],
"modifierAssociations": [
  {
    "categoryId": 0,
    "id": 278,
    "merchantId": 167,
    "modifierId": 121,
    "pos": 1,
    "productId": 177973,
    "storeId": 0
  }
],
"modifierOptions": [],
"name": "Onion Rings",
"optionIds": "",
"partNo": "TIN9YEY1",
"positionNumber": 5,
"price": 6,
"productVariants": [
  {
    "baseProductId": 177973,
    "categoryId": 1775,
    "categoryType": "Appetizers",
    "color": "Orange",
    "cost": 2,
    "dineOut": true,
    "id": 201326,
    "inventory": true,
    "item": "ONIONTEST",
    "merchantId": 167,
    "merchantStoreProducts": [
      {
        "dineOut": true,
        "discountId": 0,
        "id": 0,
        "merchantId": 167,
        "partNumber": "",
        "positionNumber": 2,
        "price": 6,
        "productId": 0,
        "quantity": 71,
        "salesTax": 0,
        "salesTaxOn": false,
        "storeId": 1
      }
    ]
  }
],
"modifierAssociations": [],
"modifierOptions": [
```



```
{
  "baseProductId": 177973,
  "id": 40,
  "merchantId": 167,
  "modifierId": 121,
  "modifierOptionId": 296,
  "modifierOptionLabel": "",
  "productId": 201326
}
],
"name": "",
"optionIds": "296",
"partNo": "",
"positionNumber": 2,
"price": 6,
"productVariants": [],
"quantity": 71,
"salesTax": true,
"skuNumber": "ONIONTEST",
"taxPercentage": 0
}
],
"quantity": 71,
"salesTax": true,
"skuNumber": "APP105",
"taxPercentage": 0
}
],
"ResponseCode": {
  "desc": "OK",
  "statusCode": 200
},
"modifiers": [
  {
    "categoryId": 0,
    "hasPriceAttr": false,
    "id": 121,
    "maxSelection": 1,
    "merchantId": 167,
    "modifierName": "TEST SIZE Changed",
    "modifierOptions": [
      {
        "id": 295,
        "label": "XXL",
        "merchantId": 167,
        "modifierId": 121,
        "pos": 1,
        "product": null,
        "productId": 0
      },
      {
        "id": 296,
```

```
    "label": "XL",
    "merchantId": 167,
    "modifierId": 121,
    "pos": 2,
    "product": null,
    "productId": 0
  },
  {
    "id": 278,
    "label": "L",
    "merchantId": 167,
    "modifierId": 121,
    "pos": 3,
    "product": null,
    "productId": 0
  },
  {
    "id": 279,
    "label": "M",
    "merchantId": 167,
    "modifierId": 121,
    "pos": 4,
    "product": null,
    "productId": 0
  },
  {
    "id": 280,
    "label": "S",
    "merchantId": 167,
    "modifierId": 121,
    "pos": 5,
    "product": null,
    "productId": 0
  },
  {
    "id": 281,
    "label": "XS",
    "merchantId": 167,
    "modifierId": 121,
    "pos": 6,
    "product": null,
    "productId": 0
  },
  {
    "id": 297,
    "label": "XXS",
    "merchantId": 167,
    "modifierId": 121,
    "pos": 7,
    "product": null,
    "productId": 0
  }
}
```

```
    ],
    "modifierType": "VARIATION",
    "pos": 1
  },
  {
    "categoryId": 2058,
    "hasPriceAttr": false,
    "id": 123,
    "maxSelection": 3,
    "merchantId": 167,
    "modifierName": "TEST ACCESSORIES Changed",
    "modifierOptions": [
      {
        "id": 287,
        "label": "Button",
        "merchantId": 167,
        "modifierId": 123,
        "pos": 1,
        "product": {
          "baseProductId": 0,
          "categoryId": 2058,
          "categoryType": "",
          "color": "",
          "cost": 0,
          "dineOut": false,
          "id": 201296,
          "inventory": false,
          "item": "",
          "merchantId": 167,
          "merchantStoreProducts": [],
          "modifierAssociations": [],
          "modifierOptions": [],
          "name": "Button",
          "optionIds": "",
          "partNo": "",
          "positionNumber": 0,
          "price": 10,
          "productVariants": [],
          "quantity": 0,
          "salesTax": false,
          "skuNumber": "",
          "taxPercentage": 0
        },
      },
      {
        "productId": 201296
      },
    ],
  },
  {
    "id": 288,
    "label": "Lace",
    "merchantId": 167,
    "modifierId": 123,
    "pos": 2,
  },
}
```

```
"product": {
  "baseProductId": 0,
  "categoryId": 2058,
  "categoryType": "",
  "color": "",
  "cost": 0,
  "dineOut": false,
  "id": 201297,
  "inventory": false,
  "item": "",
  "merchantId": 167,
  "merchantStoreProducts": [],
  "modifierAssociations": [],
  "modifierOptions": [],
  "name": "Lace",
  "optionIds": "",
  "partNo": "",
  "positionNumber": 0,
  "price": 20,
  "productVariants": [],
  "quantity": 0,
  "salesTax": false,
  "skuNumber": "",
  "taxPercentage": 0
},
"productId": 201297
},
{
  "id": 289,
  "label": "Belt",
  "merchantId": 167,
  "modifierId": 123,
  "pos": 3,
  "product": {
    "baseProductId": 0,
    "categoryId": 2058,
    "categoryType": "",
    "color": "",
    "cost": 0,
    "dineOut": false,
    "id": 201298,
    "inventory": false,
    "item": "",
    "merchantId": 167,
    "merchantStoreProducts": [],
    "modifierAssociations": [],
    "modifierOptions": [],
    "name": "Belt",
    "optionIds": "",
    "partNo": "",
    "positionNumber": 0,
    "price": 15,
```

```
        "productVariants": [],
        "quantity": 0,
        "salesTax": false,
        "skuNumber": "",
        "taxPercentage": 0
      },
      "productId": 201298
    }
  ],
  "modifierType": "ADDON",
  "pos": 3
},
{
  "categoryId": 2060,
  "hasPriceAttr": true,
  "id": 124,
  "maxSelection": 3,
  "merchantId": 167,
  "modifierName": "TEST HAT Accessories",
  "modifierOptions": [
    {
      "id": 290,
      "label": "Feather",
      "merchantId": 167,
      "modifierId": 124,
      "pos": 1,
      "product": {
        "baseProductId": 0,
        "categoryId": 2060,
        "categoryType": "",
        "color": "",
        "cost": 3,
        "dineOut": false,
        "id": 201301,
        "inventory": true,
        "item": "",
        "merchantId": 167,
        "merchantStoreProducts": [],
        "modifierAssociations": [],
        "modifierOptions": [],
        "name": "Feather",
        "optionIds": "",
        "partNo": "",
        "positionNumber": 0,
        "price": 4,
        "productVariants": [],
        "quantity": 5,
        "salesTax": true,
        "skuNumber": "",
        "taxPercentage": 0
      },
      "productId": 201301
    }
  ]
}
```

```
    },
    {
      "id": 291,
      "label": "Lace",
      "merchantId": 167,
      "modifierId": 124,
      "pos": 2,
      "product": {
        "baseProductId": 0,
        "categoryId": 2060,
        "categoryType": "",
        "color": "",
        "cost": 3,
        "dineOut": false,
        "id": 201302,
        "inventory": false,
        "item": "",
        "merchantId": 167,
        "merchantStoreProducts": [],
        "modifierAssociations": [],
        "modifierOptions": [],
        "name": "Lace",
        "optionIds": "",
        "partNo": "",
        "positionNumber": 0,
        "price": 4,
        "productVariants": [],
        "quantity": 0,
        "salesTax": true,
        "skuNumber": "",
        "taxPercentage": 0
      },
      "productId": 201302
    },
    {
      "id": 294,
      "label": "Small Button",
      "merchantId": 167,
      "modifierId": 124,
      "pos": 3,
      "product": {
        "baseProductId": 0,
        "categoryId": 2060,
        "categoryType": "",
        "color": "",
        "cost": 3,
        "dineOut": false,
        "id": 201306,
        "inventory": true,
        "item": "",
        "merchantId": 167,
        "merchantStoreProducts": [],
```

```
    "modifierAssociations": [],
    "modifierOptions": [],
    "name": "Small Button",
    "optionIds": "",
    "partNo": "",
    "positionNumber": 0,
    "price": 4,
    "productVariants": [],
    "quantity": 5,
    "salesTax": true,
    "skuNumber": "",
    "taxPercentage": 0
  },
  "productId": 201306
}
],
"modifierType": "ADDON",
"pos": 4
},
```

```
{
  "categoryId": 2064,
  "hasPriceAttr": false,
  "id": 141,
  "maxSelection": 1,
  "merchantId": 167,
  "modifierName": "TEST ACCESSORIES2",
  "modifierOptions": [
    {
      "id": 350,
      "label": "Button",
      "merchantId": 167,
      "modifierId": 141,
      "pos": 1,
      "product": {
        "baseProductId": 0,
        "categoryId": 2064,
        "categoryType": "",
        "color": "",
        "cost": 0,
        "dineOut": false,
        "id": 201340,
        "inventory": false,
        "item": "",
        "merchantId": 167,
        "merchantStoreProducts": [],
        "modifierAssociations": [],
        "modifierOptions": [],
        "name": "Button",
        "optionIds": "",
        "partNo": "",
        "positionNumber": 0,
```

```
    "price": 10,  
    "productVariants": [],  
    "quantity": 0,  
    "salesTax": false,  
    "skuNumber": "",  
    "taxPercentage": 0  
  },  
  "productId": 201340  
}  
]  
}
```

## Error Codes

```
"ResponseCode": {  
  "desc": "Unable to process the request",  
  "statusCode": 6000  
}  
"ResponseCode": {  
  "desc": "Records not found",  
  "statusCode": 6005  
}
```



# Get Menu Item Detail

## Request

**POST:** managemenu/menuitem/fetchmenuitem

## Sample Input

```
{  
  "id": 201526  
}
```

## Sample Output

```
{  
  "item": {  
    "baseProductId": 0,  
    "categoryId": 1918,  
    "categoryType": "",  
    "color": "Blue",  
    "cost": 0,  
    "description": "",  
    "dineOut": false,  
    "id": 201526,  
    "inventory": false,  
    "isVariable": 0,  
    "item": "",  
    "merchantId": 167,  
    "merchantStoreProducts": [],  
    "modifierAssociations": [  
      {  
        "categoryId": 0,  
        "id": 760,  
        "merchantId": 167,  
        "modifierId": 149,  
        "pos": 2,  
        "productId": 201526,  
        "storeId": 0  
      },  
      {  
        "categoryId": 0,  
        "id": 506,  
        "merchantId": 167,  
        "modifierId": 121,  
      }  
    ]  
  }  
}
```

```
    "pos": 1,  
    "productId": 201526,  
    "storeId": 0  
  }  
],  
"modifierOptions": [],  
"name": "TEST10",  
"optionIds": "",  
"partNo": "",  
"positionNumber": 7,  
"price": 1,  
"productType": "REGULAR",  
"productVariants": [],  
"quantity": 1,  
"salesTax": false,  
"skuNumber": "",  
"taxPercentage": 0,  
"taxSettings": []  
},  
"ResponseCode": {  
  "desc": "OK",  
  "statusCode": 200  
}  
}
```

## Error Codes

```
"ResponseCode": {  
  "desc": "Unable to process the request",  
  "statusCode": 6000  
}  
"ResponseCode": {  
  "desc": "Records not found",  
  "statusCode": 6005  
}
```

# Get Inventory List (Tracking Quantity)

## Request

**POST:** `inventory/getallinventoryitems`

## Sample Input

```
{  
  "maxRecords":10,  
  "offset":2  
}
```

## Sample Output

```
{  
  "inventoryItems": [  
    {  
      "baseProductId":0,  
      "categoryModifierType":"NONE",  
      "categoryType": "",  
      "cost":33,  
      "dineOut":false,  
      "id":156051,  
      "isDeleted":false,  
      "item": "",  
      "merchantStoreProducts": [],  
      "name": "Skillet Beef ",  
      "optionIds": "",  
      "partNo": "",  
      "price":2454,  
      "productType": "REGULAR",  
      "quantity":141,  
      "skuNumber": "BEEF100",  
      "taxSettings": []  
    },  
    {  
      "baseProductId":0,  
      "categoryModifierType":"NONE",  
      "categoryType": "",  
      "cost":4,  
      "dineOut":false,  
      "id":156052,  
      "isDeleted":false,  
      "item": "",  
      "merchantStoreProducts": [],  
      "name": "Cumin Beef ",  
      "optionIds": "",  
      "partNo": "",  
      "price":24,  
    }  
  ]  
}
```

```
    "productType":"REGULAR",
    "quantity":143,
    "skuNumber":"BEEF101",
    "taxSettings":[]
  },
  {
    "baseProductId":0,
    "categoryModifierType":"NONE",
    "categoryType":"",
    "cost":4,
    "dineOut":false,
    "id":156053,
    "isDeleted":false,
    "item":"",
    "merchantStoreProducts":[],
    "name":"Black Pepper Beef",
    "optionIds":"",
    "partNo":"",
    "price":12,
    "productType":"REGULAR",
    "quantity":140,
    "skuNumber":"BEEF102",
    "taxSettings":[]
  },
  {
    "baseProductId":0,
    "categoryModifierType":"NONE",
    "categoryType":"",
    "cost":4,
    "dineOut":false,
    "id":156054,
    "isDeleted":false,
    "item":"",
    "merchantStoreProducts":[],
    "name":"Stir-Fried Beef and Green Peppers",
    "optionIds":"",
    "partNo":"",
    "price":12,
    "productType":"REGULAR",
    "quantity":143,
    "skuNumber":"BEEF103",
    "taxSettings":[]
  },
  {
    "baseProductId":0,
    "categoryModifierType":"NONE",
    "categoryType":"",
    "cost":99911,
    "dineOut":false,
    "id":156055,
    "isDeleted":false,
    "item":"",
    "merchantStoreProducts":[],
    "name":"Stir-Fried Beef and Potatoes ",
```

```
    "optionIds": "",
    "partNo": "",
    "price": 3322,
    "productType": "REGULAR",
    "quantity": 147,
    "skuNumber": "BEEF104",
    "taxSettings": []
  },
  {
    "baseProductId": 0,
    "categoryModifierType": "NONE",
    "categoryType": "",
    "cost": 4,
    "dineOut": false,
    "id": 156056,
    "isDeleted": false,
    "item": "",
    "merchantStoreProducts": [],
    "name": "Water Boiled Beef ",
    "optionIds": "",
    "partNo": "",
    "price": 12,
    "productType": "REGULAR",
    "quantity": 136,
    "skuNumber": "BEEF105",
    "taxSettings": []
  },
  {
    "baseProductId": 0,
    "categoryModifierType": "NONE",
    "categoryType": "",
    "cost": 4,
    "dineOut": false,
    "id": 156057,
    "isDeleted": false,
    "item": "",
    "merchantStoreProducts": [],
    "name": "Dry-Fried Cumin Beef (Hunan Style)",
    "optionIds": "",
    "partNo": "",
    "price": 12,
    "productType": "REGULAR",
    "quantity": 120,
    "skuNumber": "BEEF106",
    "taxSettings": []
  },
  {
    "baseProductId": 0,
    "categoryModifierType": "NONE",
    "categoryType": "",
    "cost": 4,
    "dineOut": false,
    "id": 156058,
```

```

        "isDeleted":false,
        "item": "",
        "merchantStoreProducts": [],
        "name": "Beef Curry with Rice ",
        "optionIds": "",
        "partNo": "",
        "price": 12,
        "productType": "REGULAR",
        "quantity": 141,
        "skuNumber": "BEEF107",
        "taxSettings": []
    },
    {
        "baseProductId": 0,
        "categoryModifierType": "NONE",
        "categoryType": "",
        "cost": 4,
        "dineOut": false,
        "id": 156059,
        "isDeleted": false,
        "item": "",
        "merchantStoreProducts": [],
        "name": "Slow-Cooked Beef with Potatoes ",
        "optionIds": "",
        "partNo": "",
        "price": 12,
        "productType": "REGULAR",
        "quantity": 144,
        "skuNumber": "BEEF108",
        "taxSettings": []
    }
],
"ResponseCode": {"desc": "OK", "statusCode": 200}
}

```

## Error Codes

```

"ResponseCode": {
  "desc": "Unable to process the request",
  "statusCode": 6000
}
"ResponseCode": {
  "desc": "Records not found",
  "statusCode": 6005
}

```

# Edit Inventory Item Quantity

## Request

**PUT:** inventory/editinventoryitem

## Sample Input

```
{
  "storeId":1,
  "inventoryItems":{
    "id":3188,
    "quantity":"10"
  }
}
```

## Sample Output

```
{
  "ResponseCode": {
    "desc": "OK",
    "statusCode": 200
  }
}
```

## Error Codes

```
"ResponseCode": {
  "desc": "Unable to process the request",
  "statusCode": 6000
}
"ResponseCode": {
  "desc": "Records not found",
  "statusCode": 6005
}
"ResponseCode": {
  "desc": "Menu Item does not exist",
  "statusCode": 3055
}
```

# Batch Edit Inventory Item

## Request

**PUT:** inventory/batch/editinventoryitem

## Sample Input

```
{
  "inventoryItems": [
    {"id": 3188, "name": "CAPBELT", "cost": 30, "quantity": "10", "partNo": "", "skuNumber": "DD3", "price": 3.00},
    {"id": 3189, "name": "Skirt", "cost": 30, "quantity": "10", "partNo": "", "skuNumber": "DD31", "price": 12.00}
  ]
}
```

## Sample Output

```
{
  "ResponseCode": {
    "desc": "OK",
    "statusCode": 200
  }
}
```

## Error Codes

```
"ResponseCode": {
  "desc": "Unable to process the request",
  "statusCode": 6000
}
"ResponseCode": {
  "desc": "Records not found",
  "statusCode": 6005
}
```



# Get All Employees

## Request

**POST:** manageemployee/getallemployees

## Sample Input To fetch all active/inactive employees

```
{  
  "searchCriteria":{"activeFilter":1}  
}
```

## Sample Input to fetch only active employees

```
{  
  "searchCriteria":{"activeFilter":0}  
}
```

## Sample Input To fetch only inactive employees

```
{  
  "searchCriteria":{"activeFilter":2}  
}
```

## Sample Output

```
{  
  "users": [  
    {  
      "allowedHourWeekly": "",  
      "allowedOp": "1110",  
      "deleted": false,  
      "email": "shria1QAA@gmail.com",  
      "failedLoginAttempts": 0,  
      "hasPin": 1,  
      "id": 397,  
      "isActive": true,  
      "isClockInReqToAccess": true,  
      "merchantId": 291,  
    }  
  ]  
}
```

```
"merchantStoreAccess": [],
"name": "",
"oldPassword": "",
"phoneNumber": "9920289311",
"resetDate": "",
"superUser": false,
"userFirstName": "BspTest",
"userLastName": "Chand",
"userName": "BspTest Chand",
"usrGroupCode": "User"
},
{
  "allowedHourWeekly": "",
  "allowedOp": "111111",
  "deleted": false,
  "email": "shria@arunji.com",
  "failedLoginAttempts": 0,
  "hasPin": 1,
  "id": 334,
  "isActive": true,
  "isClockInReqToAccess": false,
  "merchantId": 291,
  "merchantStoreAccess": [
    {
      "allowedOp": "111111",
      "id": 548,
      "isClockInReqToAccess": false,
      "isDeleted": 0,
      "merchantId": 291,
      "userId": 334,
      "usrGroupCode": "Admin"
    }
  ],
  "name": "",
  "oldPassword": "",
  "phoneNumber": "9920289311",
  "resetDate": "",
  "superUser": false,
  "userFirstName": "BspTest",
  "userLastName": "Chand",
  "userName": "BspTest Chand",
  "usrGroupCode": "Admin"
},
{
  "allowedHourWeekly": "",
  "allowedOp": "111111",
  "deleted": false,
  "email": "shria@arunjia.com",
  "failedLoginAttempts": 0,
  "hasPin": 1,
  "id": 335,
  "isActive": true,
```

```
"isClockInReqToAccess": false,
"merchantId": 291,
"merchantStoreAccess": [
  {
    "allowedOp": "111111",
    "id": 549,
    "isClockInReqToAccess": false,
    "isDeleted": 0,
    "merchantId": 291,
    "userId": 335,
    "usrGroupCode": "User"
  }
],
"name": "",
"oldPassword": "",
"phoneNumber": "9920289311",
"resetDate": "",
"superUser": false,
"userFirstName": "Jeni",
"userLastName": "milton",
"userName": "Jeni milton",
"usrGroupCode": "User"
},
{
  "allowedHourWeekly": "",
  "allowedOp": "1000110",
  "deleted": false,
  "email": "shria@arunjia.com",
  "failedLoginAttempts": 0,
  "hasPin": 1,
  "id": 373,
  "isActive": true,
  "isClockInReqToAccess": false,
  "merchantId": 291,
  "merchantStoreAccess": [
    {
      "allowedOp": "1000110",
      "id": 561,
      "isClockInReqToAccess": false,
      "isDeleted": 0,
      "merchantId": 291,
      "userId": 373,
      "usrGroupCode": "User"
    }
  ]
},
"name": "",
"oldPassword": "",
"phoneNumber": "9920289311",
"resetDate": "",
"superUser": false,
"userFirstName": "Jeni",
"userLastName": "milton",
```

```
"userName": "Jeni milton",
"usrGroupCode": "User"
},
{
  "allowedHourWeekly": "",
  "allowedOp": "1000110",
  "deleted": false,
  "email": "shria@arunjia.com",
  "failedLoginAttempts": 0,
  "hasPin": 1,
  "id": 374,
  "isActive": true,
  "isClockInReqToAccess": false,
  "merchantId": 291,
  "merchantStoreAccess": [
    {
      "allowedOp": "1000110",
      "id": 562,
      "isClockInReqToAccess": false,
      "isDeleted": 0,
      "merchantId": 291,
      "userId": 374,
      "usrGroupCode": "User"
    }
  ],
  "name": "",
  "oldPassword": "",
  "phoneNumber": "9920289311",
  "resetDate": "",
  "superUser": false,
  "userFirstName": "Jeni",
  "userLastName": "milton",
  "userName": "Jeni milton",
  "usrGroupCode": "User"
},
{
  "allowedHourWeekly": "",
  "allowedOp": "1000110",
  "deleted": false,
  "email": "shria@arunjia.com",
  "failedLoginAttempts": 0,
  "hasPin": 1,
  "id": 375,
  "isActive": false,
  "isClockInReqToAccess": false,
  "merchantId": 291,
  "merchantStoreAccess": [
    {
      "allowedOp": "1000110",
      "id": 563,
      "isClockInReqToAccess": false,
      "isDeleted": 0,
```

```
    "merchantId": 291,  
    "userId": 375,  
    "usrGroupCode": "User"  
  }  
],  
  "name": "",  
  "oldPassword": "",  
  "phoneNumber": "9920289311",  
  "resetDate": "",  
  "superUser": false,  
  "userFirstName": "Jeni",  
  "userLastName": "milton",  
  "userName": "Jeni milton",  
  "usrGroupCode": "User"  
},  
{  
  "allowedHourWeekly": "",  
  "allowedOp": "1000110",  
  "deleted": false,  
  "email": "shria@arunjia.com",  
  "failedLoginAttempts": 0,  
  "hasPin": 1,  
  "id": 376,  
  "isActive": true,  
  "isClockInReqToAccess": true,  
  "merchantId": 291,  
  "merchantStoreAccess": [  
    {  
      "allowedOp": "1000110",  
      "id": 564,  
      "isClockInReqToAccess": true,  
      "isDeleted": 0,  
      "merchantId": 291,  
      "userId": 376,  
      "usrGroupCode": "User"  
    }  
  ],  
  "name": "",  
  "oldPassword": "",  
  "phoneNumber": "9920289311",  
  "resetDate": "",  
  "superUser": false,  
  "userFirstName": "Jeni",  
  "userLastName": "milton",  
  "userName": "Jeni milton",  
  "usrGroupCode": "User"  
},  
{  
  "allowedHourWeekly": "",  
  "allowedOp": "1000110",  
  "deleted": false,  
  "email": "shria@arunjia.com",
```

```
"failedLoginAttempts": 0,
"hasPin": 1,
"id": 377,
"isActive": true,
"isClockInReqToAccess": false,
"merchantId": 291,
"merchantStoreAccess": [
  {
    "allowedOp": "1000110",
    "id": 565,
    "isClockInReqToAccess": false,
    "isDeleted": 0,
    "merchantId": 291,
    "userId": 377,
    "usrGroupCode": "User"
  }
],
"name": "",
"oldPassword": "",
"phoneNumber": "9920289311",
"resetDate": "",
"superUser": false,
"userFirstName": "Jeni",
"userLastName": "milton",
"userName": "Jeni milton",
"usrGroupCode": "User"
},
{
  "allowedHourWeekly": "",
  "allowedOp": "1000110",
  "deleted": false,
  "email": "ol@arunjia.com",
  "failedLoginAttempts": 0,
  "hasPin": 1,
  "id": 379,
  "isActive": true,
  "isClockInReqToAccess": false,
  "merchantId": 291,
  "merchantStoreAccess": [
    {
      "allowedOp": "1000110",
      "id": 567,
      "isClockInReqToAccess": false,
      "isDeleted": 0,
      "merchantId": 291,
      "userId": 379,
      "usrGroupCode": "Admin"
    }
  ]
},
"name": "",
"oldPassword": "",
"phoneNumber": "9920289311",
```

```
"resetDate": "",
"superUser": false,
"userFirstName": "Jeni",
"userLastName": "milton",
"userName": "Jeni milton",
"usrGroupCode": "Admin"
},
{
  "allowedHourWeekly": "",
  "allowedOp": "1000110",
  "deleted": false,
  "email": "olk@arunjia.com",
  "failedLoginAttempts": 0,
  "hasPin": 1,
  "id": 378,
  "isActive": true,
  "isClockInReqToAccess": true,
  "merchantId": 291,
  "merchantStoreAccess": [
    {
      "allowedOp": "1000110",
      "id": 811,
      "isClockInReqToAccess": true,
      "isDeleted": 0,
      "merchantId": 291,
      "userId": 378,
      "usrGroupCode": "Admin"
    }
  ],
  "name": "",
  "oldPassword": "",
  "phoneNumber": "9920289311",
  "resetDate": "",
  "superUser": false,
  "userFirstName": "Jeni",
  "userLastName": "milton",
  "userName": "Jeni milton",
  "usrGroupCode": "Admin"
},
{
  "allowedHourWeekly": "",
  "allowedOp": "1000110",
  "deleted": false,
  "email": "olasx@sarunjia.com",
  "failedLoginAttempts": 0,
  "hasPin": 1,
  "id": 394,
  "isActive": true,
  "isClockInReqToAccess": false,
  "merchantId": 291,
  "merchantStoreAccess": [
    {
```

```
        "allowedOp": "1000110",
        "id": 820,
        "isClockInReqToAccess": false,
        "isDeleted": 0,
        "merchantId": 291,
        "userId": 394,
        "usrGroupCode": "Admin"
    }
],
"name": "",
"oldPassword": "",
"phoneNumber": "9920289311",
"resetDate": "",
"superUser": false,
"userFirstName": "Jeni",
"userLastName": "milton",
"userName": "Jeni milton",
"usrGroupCode": "Admin"
},
{
    "allowedHourWeekly": "",
    "allowedOp": "1000110",
    "deleted": false,
    "email": "olasx@sarunjia.com",
    "failedLoginAttempts": 0,
    "hasPin": 1,
    "id": 395,
    "isActive": true,
    "isClockInReqToAccess": false,
    "merchantId": 291,
    "merchantStoreAccess": [
        {
            "allowedOp": "1000110",
            "id": 821,
            "isClockInReqToAccess": false,
            "isDeleted": 0,
            "merchantId": 291,
            "userId": 395,
            "usrGroupCode": "User"
        }
    ],
    "name": "",
    "oldPassword": "",
    "phoneNumber": "9920289311",
    "resetDate": "",
    "superUser": false,
    "userFirstName": "Jeni",
    "userLastName": "milton",
    "userName": "Jeni milton",
    "usrGroupCode": "User"
},
{
```



```

    "allowedHourWeekly": "",
    "allowedOp": "1000110",
    "deleted": false,
    "email": "olak@arunjia.com",
    "failedLoginAttempts": 0,
    "hasPin": 1,
    "id": 396,
    "isActive": true,
    "isClockInReqToAccess": true,
    "merchantId": 291,
    "merchantStoreAccess": [
      {
        "allowedOp": "1000110",
        "id": 822,
        "isClockInReqToAccess": true,
        "isDeleted": 0,
        "merchantId": 291,
        "userId": 396,
        "usrGroupCode": "Admin"
      }
    ],
    "name": "",
    "oldPassword": "",
    "phoneNumber": "9920289311",
    "resetDate": "",
    "superUser": false,
    "userFirstName": "Jeni",
    "userLastName": "milton",
    "userName": "Jeni milton",
    "usrGroupCode": "Admin"
  }
],
"ResponseCode": {
  "desc": "OK",
  "statusCode": 200
}
}

```

## Error Codes

```

"ResponseCode": {
  "desc": "Unable to process the request",
  "statusCode": 6000
}
"ResponseCode": {
  "desc": "Records not found",
  "statusCode": 6005
}

```

## Notes

- For backward compatibility if the input does not have search "criteria" condition, the API by default returns only **Active** employees
- The flag "deleted" in the response tells whether an employee is active/inactive
  - true -> inactive employee
  - false->active employee
- Added an array merchantStoreAccess[] in the user data model , which holds the set of merchant/stores the user has access to

# Logout

## Request

**POST:** <https://api.talech.com/authentication/logout>

## Sample Input

```
{
  "user":{
    "id":635299
  },
  "storeId":1,
  "merchantId":635299
}
```

## Sample Output

```
{"ResponseCode":{"desc":"OK","statusCode":200}}
```

## Error Codes

```
"ResponseCode": {
  "desc": "Unable to process the request",
  "statusCode": 6000
}
"ResponseCode": {
  "desc": "Records not found",
  "statusCode": 6005
}
```

# Sample PHP code

```
<?php

/******
 *
 * This class handles making API calls and storing the security token for reuse
 *
 *****/

class TalechApi {
    public $base_url = "";
    public $security_token = "";

    public function __construct($base_url = 'https://mapi.talech.com/', $security_token = "") {
        $this->base_url = $base_url;
        $this->security_token = $security_token;
    }

    /**
     * Initialize curl and make an api call. Determine if a new security token
     * is being returned as part of the response, and if so, parse and store that
     * token.
     */
    public function call($uri, $method = 'POST', $data = false, $headers = false, $use_token =
true){

        $url = $this->base_url . $uri;

        $default_headers = array('Content-Type: application/json');
        if ($use_token && $this->security_token) {
            $default_headers[] = 'securityToken: ' . $this->security_token;
        }
        $headers = $headers ? array_merge($default_headers, $headers) :
$default_headers;

        $curl = curl_init();

        switch ($method) {
            case "POST":
                curl_setopt($curl, CURLOPT_POST, 1);
```

```

        if ($data) {
            curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
        }
        break;
    case "PUT":
        curl_setopt($curl, CURLOPT_CUSTOMREQUEST, 'PUT');
        if ($data) {
            curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
        }
        break;
    }

    curl_setopt($curl, CURLOPT_HTTPHEADER, $headers);

    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    curl_setopt($curl, CURLOPT_HEADER, 1);

    $response = curl_exec($curl);

    //Parse for new security token
    list($headers, $body) = explode("\r\n\r\n", $response, 2);
    while (strpos($headers, " 100 Continue") !== false){
        list($headers, $body) = explode( "\r\n\r\n", $body , 2);
    }

    if ($headers) {
        $headers = explode("\r\n", $headers);
        foreach($headers as $header) {
            $header = explode(':', $header, 2);
            if (count($header) === 2 && $header[0] === 'X-POS-SecurityToken') {
                //error_log('Renewing security token: ' . $header[1]);
                $this->security_token = $header[1];
            }
        }
    }
}

//Return api response
return $body;
}

/*****
*

```

```

* Helper functions to simplify common api calls
*
*****/
public function post($uri, $data = false, $headers = false, $use_token = true) {
    return $this->call($uri, 'POST', is_array($data) ? json_encode($data) : $data,
$headers, $use_token);
}

public static function get($uri, $data = false, $headers = false, $use_token = true) {
    return $this->call($uri, 'GET', is_array($data) ? json_encode($data) : $data, $headers,
$use_token);
}

public static function put($uri, $data = false, $headers = false, $use_token = true) {
    return $this->call($uri, 'PUT', is_array($data) ? json_encode($data) : $data, $headers,
$use_token);
}
}

/*****
*
* Instantiate a TalechApi object, retrieve an access token using a merchant
* token and then get the list of the merchant's customers.
*
*****/
$api = new TalechApi();

//Using the client id, client secret and merchant token, retrieve an access token
$client_id    = "example_application";
$client_secret = "abcd1234";
$merchant_token = "16940471/0GqQfTC25I5f1rNtFjAz0tTN";

$input = array(
    "grant_type" => "refresh_token",
    "client_id"  => $client_id,
    "refresh_token" => $merchant_token,
    "client_secret" => $client_secret
);

$response = json_decode($api->post('o/oauth2/token', $input), true);
print_r($response);
/* Sample output:
{

```

```
        "access_token": "16851849/uW0f3H24SL65FXo6Ci6jxSq0",
        "expires_in": 1800,
        "token_type": "Bearer"
    }
*/
if (isset($response['access_token'])) {
    //Set the security/access token
    $api->security_token = $response['access_token'];
}

/*
 * start making API calls
 */

//Get MerchantStoreInfo
$resp = json_decode($api->post('authentication/getAllMerchantStoreInfo', '{}'), true);
print_r($resp);

//Get a list of customers
$input = array(
    'offset' => 0,
    'limit' => 100
);
$resp = json_decode($api->post('managecustomer/getallcustomers', $input), true);
print_r($resp);
```