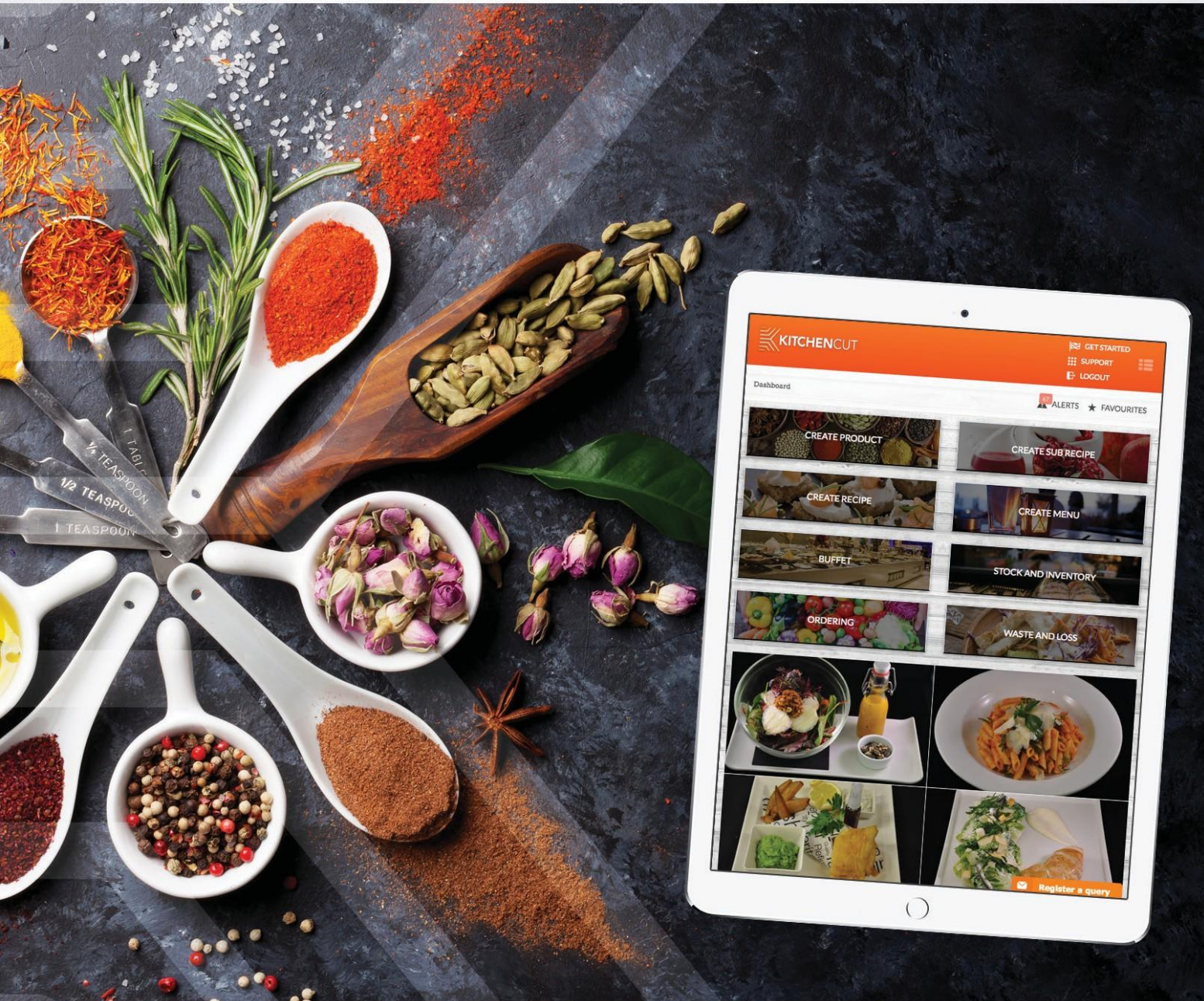




KITCHENCUT
CUTTING-EDGE F&B TECHNOLOGY



Sales API
V4.2



KITCHENCUT
CUTTING-EDGE F&B TECHNOLOGY

Contents:

1. Authorize
2. Write EPoS Data
3. GET Wastage Reasons
4. Write Wastage Data
5. API request return errors



1. Authorize

Required data:

Tenant URL	https:// + "Tenant url" + /kc-api/oauth/authorize
API key	e.g. 1V7e2IKiIK6iuXGvV8U8GzekBKW3OblQnHdKMBFP
Service key	write_epos_data

Request:

```
curl --location --request POST 'https://YOUR-DOMAIN.kitchencut.com/kc-api/oauth/authorize' \  
--header 'Content-Type: application/x-www-form-urlencoded' \  
--data-urlencode 'api_key=1V7e2IKiIK6iuXGvV8U8GzekBKW3OblQnHdKMBFP' \  
--data-urlencode 'service_key=write_epos_data'
```

Response:

```
{  
  "access_token": "CZafV0gwIydIPecik7UsldI77BuRmjrrIEPicWHx",  
  "token_type": "Bearer",  
  "expire_date": "2022-07-27 14:06:58"  
}
```

2. Write EPoS Data

Required data:

Tenant URL	https:// + "Tenant url" + /kc-api/epos/add
Access token	e.g. 3nXXE779urnRRrXSlo0X9rukkzeWlCoct9uocGDo – generated from authorize request
EPoS data	field items

EPoS data:

epos_code	(string) 1 - 255 chars	Menu item EPoS code e.g. 39437
business_desc	(string) 1 - 200 chars	Client name e.g. Brand1
outlet_name	(string) 1 - 250 chars	Revenue outlet name e.g. Restaurant1
outlet_epos_id	(string) 0 - 250 chars	RO id - supersede RO name, RO id is unique for RO.
item_desc	(string) 1 - 250 chars	Name of menu item
qty	(numeric)	Quantity sold
net	(numeric)	Net price (minus tax)
category_desc	(string) 1 - 40 chars	Reporting category e.g. Mains, White wine...
sale_date	(date) format Y-m-d H:i:s	Date/time of sales event

Request:

Max 1000 items, if over then system returns 400 Bad Request error.



KITCHENCUT
CUTTING-EDGE F&B TECHNOLOGY

Request x-www-form-urlencoded:

```
curl --location --request POST 'https://YOUR-DOMAIN.kitchencut.com/kc-api/epos/add' \  
  --header 'Authorization: Bearer ACCESS_TOKEN' \  
  --header 'Content-Type: application/x-www-form-urlencoded' \  
  --data-urlencode 'epos_data[YOUR_ID][epos_code]=1000001' \  
  --data-urlencode 'epos_data[YOUR_ID][business_desc]=Tevalis' \  
  --data-urlencode 'epos_data[YOUR_ID][outlet_name]=Bar' \  
  --data-urlencode 'epos_data[YOUR_ID][outlet_epos_id]=Xds665' \  
  --data-urlencode 'epos_data[YOUR_ID][item_desc]=Orange Juice Sml' \  
  --data-urlencode 'epos_data[YOUR_ID][qty]=20' \  
  --data-urlencode 'epos_data[YOUR_ID][net]=80' \  
  --data-urlencode 'epos_data[YOUR_ID][category_desc]=Beverage' \  
  --data-urlencode 'epos_data[YOUR_ID][sale_date]=2020-03-05 10:20:00'
```

Request JSON:

```
curl --location --request POST 'https://YOUR-DOMAIN.kitchencut.com/kc-api/epos/add' \  
  --header 'Authorization: Bearer ACCESS_TOKEN' \  
  --header 'Content-Type: application/json' \  
  --data-raw '{  
    "epos_data": {  
      "1": {  
        "epos_code": 1000001,  
        "business_desc": "Tevalis 2",  
        "outlet_name": "Bar",  
        "outlet_epos_id": "Xds665",  
        "item_desc": "Orange Juice Sml",  
        "qty": 20,  
        "net": 80,  
        "category_desc": "Beverage",  
        "sale_date": "2021-07-26 10:20:00"  
      }  
    }  
  }'
```

KITCHENCUT.COM

Copyright 2019 Kitchen CUT

All Rights Reserved



KITCHENCUT
CUTTING-EDGE F&B TECHNOLOGY

GZIP JSON request:

```
$ch = curl_init();  
curl_setopt($ch, CURLOPT_URL, $url);  
curl_setopt($ch, CURLOPT_RETURNTRANSFER, TRUE);  
curl_setopt($ch, CURLOPT_HTTPHEADER, array(  
    'Authorization: Bearer ` . $access_token,  
    'Content-Type: application/json',  
    'Content-Length: ` . strlen(gzencode(json_encode($post_data)))  
));  
curl_setopt($ch, CURLOPT_POSTFIELDS, gzencode(json_encode($post_data)));  
  
$result = json_decode(curl_exec($ch));  
$infodata = curl_getinfo($ch);  
curl_close($ch);
```

Response:

```
{  
  "2": {  
    "status": "error",  
    "fields": {  
      "business_desc": "Wrong data, string must have 1 - 200 characters.",  
      "outlet_name": "Wrong data, string must have 1 - 250 characters.",  
      "item_desc": "Wrong data, must string have 1 - 250 characters.",  
      "qty": "Wrong data, must by numeric.",  
      "net": "Wrong data, must by numeric.",  
      "category_desc": "Wrong data, string must have 1 - 40 characters.",  
      "sale_date": "Wrong data, string must be a date, format Y-m-d H:i:s"  
    }  
  },  
  "1": {  
    "status": "success"  
  }  
}
```

3. GET Wastage Reasons

Required data:

Tenant URL	https:// + "Tenant url" + /kc-api/epos/wastage_reasons
Access token	3nXXE779urnRRrXSlo0X9rukkzeWlCoct9uocGDo - generated from authorize request

Request:

```
curl --location --request GET 'https://wargrave-test2.beta2.kitchencut.com/kc-api/epos/wastage_reasons'
```

Response Example:

```
{
  "data": [
    {
      "id": 6,
      "name": "Buffet wastage"
    },
    {
      "id": 8,
      "name": "EPOS unspecified wastage"
    },
    {
      "id": 3,
      "name": "Product burnt / overcooked/ dropped"
    },
    {
      "id": 2,
      "name": "Product date expired"
    },
    {
      "id": 4,
      "name": "Product sent back - complaint"
    },
    {
      "id": 7,
      "name": "Production wastage"
    },
    {
      "id": 1,
      "name": "Quality sub standard"
    },
    {
      "id": 5,
      "name": "Waste back from restaurant"
    }
  ]
}
```

4. Write Wastage Data

Required data:

Tenant URL	https:// + "Tenant url" + /kc-api/epos/add_wastage
Access token	3nXXE779urnRRrXSlo0X9rukkzeWlCoct9uocGDo - generated from authorize request
EPoS data	field items

EPoS Data:

name	String 1 - 250 chars	(Optional) Wastage name
outlet_epos_id	String 1 - 250 chars	(Required) Revenue outlet EPOS ID
items[1][item_pos_code]	String 1 - 200 chars	(Required) POS Menu Item code
items[1][qty]	Numeric	(Required) Quantity
items[1][reason_id]	Integer	(Optional) Wastage reason ID (Default: EPOS default wastage reason ID)
items[1][employee]	String 1 - 200 chars	(Optional) Employee name
items[1][action]	String 1 - 200 chars	(Optional) Action taken

Request:

```
curl --location --request POST 'https://wargrave-test2.beta2.kitchencut.com/kc-api/epos/add_wastage' \
--data-urlencode 'name=API wastage' \
--data-urlencode 'outlet_epos_id=012' \
--data-urlencode 'items[544224][item_pos_code]=7565' \
--data-urlencode 'items[544224][qty]=1' \
--data-urlencode 'items[544224][reason_id]=6' \
--data-urlencode 'items[544224][unit_uom]=each-each-mtr' \
--data-urlencode 'items[544224][employee]=Steve' \
--data-urlencode 'items[544224][action]=Training session' \
```



KITCHENCUT
CUTTING-EDGE F&B TECHNOLOGY

Request JSON:

```
{
  "name": "API wastage",
  "outlet_epos_id": "012",
  "items": [{
    "item_pos_code": "7565",
    "qty": "1",
    "reason_id": "6",
    "unit_uom": "each-each-mtr",
    "employee": "Steve",
    "action": "Training session"
  }],
}
```

Response:

```
{
  "status": "success",
  "message": "Wastage routine has been successfully created."
}
```

KITCHENCUT.COM

Copyright 2019 Kitchen CUT

All Rights Reserved

5. API Request return errors

When a request fails, the resource server responds using the appropriate HTTP status code.

Code	Description
200	HTTP/1.1 200 OK
202	HTTP/1.1 202 Accepted
400	HTTP/1.1 400 Bad Request
401	HTTP/1.1 401 Unauthorized
403	HTTP/1.1 403 Forbidden
404	HTTP/1.1 404 Not Found
409	HTTP/1.1 409 Conflict
429	HTTP/1.1 429 Too many requests
500	HTTP/1.1 500 Internal Server Error
501	HTTP/1.1 501 Not Implemented
503	HTTP/1.1 503 Service Unavailable