

# Integration - Export API - implementation steps

Clarifruit supports REST API, to allow our customers to integrate with our system.

Integration use cases:

- Barcode scanning - to get inspection details filled automatically.
- Export Hooks - like inspection creation, approval, OR deletion.
- Import inspections into Clarifruit.

## Export Integration - Implementation Steps:

- Mappings - in order to store & use the customer's ID's for each parameter of information - we map all the relevant information - with Clarifruit ID & Customer ID (externalID).

You can easily export a comma-separated values (CSV) mappings file, which would be nice for you to be completed prior to using the API.

You don't have to map Clarifruit fields with yours - but it will be much easier for you to recognize the data this way. Otherwise, you will get the information we are sending with our ID's.

For example, the CSV mappings file might contain the following information, requiring you to fill out the last column:

Field	Field Type	Clarifruit ID	Your ID
Apple	Produce Family	212	apple
Common Green Apple	Produce Variety	222	CGA
High Season Apple Standard	Custom Standard	8405	123
Intake Inspection Process	Custom Inspection Process	5456	intake
Grower	Custom Property used in Inspection Process	Xnbz	grower

Adding your ID to the mapping file will allow you as a customer to recognize your data by translating our ID for each field exported into your system that you will be storing Clarifruit inspection.

- **Customer tasks:**
  - Provide Clarifruit with an **endpoint URL & token** for authentication.
  - Fill in the mappings CSV file with his ID's & send it back to Clarifruit.
- **Clarifruit tasks:**
  - Get the customer endpoint & token.
    - Try to send a request outside of the system - using Postman / Insomnia or any other API platform.
  - Store the customer token - in our internal Git - Private services.
  - Configure the relevant use cases using the customer endpoint URL.
  - For Barcode:
    - **Produce** - leave blank if you want the integration to be used for all produce categories.
      - You can set a different configuration per produce if needed - to be used in case the information will be totally different per produce. Or the endpoint will be different per produce.
    - **Barcode Field** - Will be the inspection name OR a barcode property - to trigger the API.
    - **NameKey** - means the expected inspection name key within the JSON response. this will use for the inspection name.
      - Do not use it if the name is the field that would trigger the API. Use it only if the API uses a barcode property to trigger the API.
      -

## View Configuration

Action Type	Callback
Hook Type	Barcode Lookup
Produce	Grapes
Barcode Field	barcode
Name Key	inspectionName
Produce Key	produceKey
Variety Key	codeVariety
Process Key	processKey
Standard Key	standardKey
Package Key	packageKey
Url	https://private-services.clarifruit.com/barcode?url=https://test-bar.onrender.com/test?barcode={0}
Http Headers	X-API-KEY: <input type="text"/>
Http Method	GET

Close

- **Produce key, Variety Key, Process Key, Standard Key, Package Key** - set the key that would be in the JSON response - to be associated & recognized by Clarifruit.
- **URL** - That would be the customer endpoint URL - the API will send get requests from.
- **HTTP Headers** - This is used for Clarifruit internal token - to be handled internally.
- Clarifruit to send a list with all relevant mappings to the customer.
- Clarifruit - to import the CSV mappings file into the customer's company within Clarifruit Dashboard configurations.
- Testing - using the barcode examples.
- **Tip - use the simulator to test communications & results.**
- The simulator is available within the edit pop-up of the hook.
- Here you can trigger a valid value & test it.
  - In this case - this is a barcode value, and we can see the JSON response from the customer.

## Edit Configuration

Simulate Event

Produce Type  
All × ∨

Barcode  ▶

Endpoint  
<https://private-services.clarifruit.com/barcode?ur...>

Response After Conversion Endpoint Response

Advanced ∨

```
{
  "codeVariety": "THOMPSON",
  "customer": "Amazon",
  "ggnumber": 31254236,
  "grower": "Grower C",
  "inspectionName": "Grapes - PS: 755297",
  "processKey": "p_grape_ps",
  "standardKey": "stand_ps_graes"
}
```

[Cancel](#) Save

- And here we can see the mappings that Clarifruit system performs - to test that all mappings are correct, and the system recognizes all relevant parameters from the customer's JSON.

Barcode  ▶

Response After Conversion Endpoint Response

```
Name: Grapes - PS: 755297
Variety: Thompson
Process: Grapes - Pre Shipment
Standard: Pre-Shipment
Properties:
  GG Number CSD: 31254236
  Grower Name: Grower C
  Customer: Amazon
```

- This is it 😊 We are all set & good to go.