API Reference

Documentation

Version 1.0.1.

Document Version Control

Date	Version	Author	Summary of Changes
10/05/2021	1.0.0.	Adam Barwick	Initial Version of Customer Documentation (internal versions exist under separate version control)
06/01/2022	1.0.1.	Nick Howard	Amended structure of authentication request. See 2.6. Changed spelling of Authorisation to be 'Authorization' for consistency with request header.

Legal

This is not a public document and cannot be distributed freely. The intellectual property contained in this document remains the sole ownership of Ralawise Limited and is intended for use only by those in direct receipt from Ralawise Limited. Unauthorised access to the Ralawise APIs is strictly prohibited.

Table of Contents

1. Introduction

1.1.   Purpose of the Document

The purpose of this document is to provide a reference to the public interface for the Ralawise API's. The document is a technical reference guide with software developers and integration specialists as its intended audience.

1.2.   Scope of the Document

The interfaces defined in this document are those that are required by consumers that will interact with the Ralawise Authentication service, Inventory service and Order service. The examples given assume a RESTful style using JSON as the message format and HTTP as the application-level protocol.

1.3.   Access Permissions

Access to the API interface may be granted to one or more Ralawise website user accounts, and all API operations will be done in the context of the account used when requesting an access token using the Authentication API. User accounts can be setup in the My Account -> Users area of the website by users with Manager permissions to the account. To request API access for a specific user, please contact the Ralawise E-Commerce team at ecommerce@ralawise.com.

1.4.   Data Format and Protocol

The API accepts and responds to requests over HTTP 1.1, returning data in JSON format.

1.5.   API Entry Point / Base URL and Development Environment

Live URL: https://api.ralawise.com/
Dev URL: https://api.ralawise.com/test

The API is versioned, and the version will **always** be the first parameter in a URL after the base URL. Versions will be expressed as ordinal numbers prefixed with 'v'.

Example:

- https://api.ralawise.com/v1/

In order to use the Dev environment, you will need to contact Ralawise to setup a Dev user account for you.

> **NOTE:** *Orders placed through the Live URL (which is referred to throughout this document) will be submitted and processed by Ralawise as live orders which can be monitored through the Ralawise website. Whilst the Dev URL is provided for testing, there is no corresponding Dev website available to monitor orders. When testing, Ralawise can provide feedback on the success of orders placed (in addition to the API responses indicating success or failure). Ralawise recommends rigorous testing using the Dev URL before calling the Live URL.*

To aid your understanding and development process, a Postman collection in JSON format configured for the Dev environment is available on request from ecommerce@ralawise.com. This includes sample success and failure calls for each API using the test URL.

Postman is a third-party application, unsupported by Ralawise, available from www.postman.com.

1.6.   Overview

The first step is to use the Authentication API to request an access token by providing the API user login details. You can use any user accounts registered on the Ralawise website to access the API, but you must first request access for the user(s) from Ralawise.

Once an access token is received this must be set when using the Inventory API and Order API.

The access token is valid for 1200s (20mins) after which a new token must be requested.

2. Authentication API

2.1.   Reference

The API is RESTful, and provides consumers with the ability to generate a JSON Web Token (JWT).

This API is meant to be used by authorised services running on behalf of Ralawise customers, and all operations expect valid credentials to be provided in the form of a JSON Web Token or JWT (within the Authorization header of the request).

2.2.   Authorization and Authentication

The caller is expected to provide credentials within the Login request payload. Upon successful authentication, the user will receive a time-limited JSON Web Token (JWT) in return. This JWT is required to access functional APIs (such as the Inventory API and Order API).

2.3.   Request Headers

Field	Type	Length	Occurrence	Notes
Content-Type	String	-	1	application/json
Accept-Encoding	String	-	0.1	Used for HTTP compression: - Accept-Encoding: gzip, deflate

2.4.   Response Headers

Field	Type	Length	Occurrence	Notes

2.5.   Query String Parameters

Field	Type	Length	Occurrence	Notes

2.6.   Models

The structure for an authentication request is as follows.

```
{
  "user": "john.smith@customer.com",
  "password": "ewfjfb34t7cbadfyu"
}
```

2.6.1.   Credentials

Property Name	Type	Length	Occurrence	Notes
user	String	100	1	Username for the Ralawise online shop
password	String	50	1	Password for the user

2.6.2.   Error / Information Messages

Property Name	Type	Length	Occurrence	Notes
errorCode	String	100	1	A string error code
parameterName	String	500	0.1	The parameter name of the property the validation message applies to from the request
Level	String	10	1	Set as "information" for all 200 range responses. Set as "error" for all non-200 range responses"
errorMessage	String	500	1	Informational message returned to aid user support

2.7.   Resources

2.7.1. Login (POST)

Login	
Description	Authenticate on Ralawise's web platform.
URL	https://api.ralawise.com/v1/login
Verb(s)	POST
Input Headers	None
Input Route	None
Input Query String	None
Input Body	

	Name	Type	Length	Occurrence	Notes
Input Body	Credentials	Object	-	1	Login credentials for the Ralawise online shop.

Defaults	None
Outputs	Time-limited JSON Web Token
Output Headers	None
Response Codes	200 (OK) – The authentication process was performed successfully 400 (Bad Request) – The request resulted in an error or the operation was called with invalid credentials 500 (Internal Server Error) – An unexpected error occurred
Information Messages	None
Error Messages	Bad Request
Events Raised	None

Example Input	https://api.ralawise.com/v1/login
	{
	
	    "user": "john.smith@customer.com",
	    "password": "ewfjfb34t7cbadfyu"
	
	}

Example Output	{
	  "access_token":
	"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiYWRt,
	  "token_type": "bearer",
	  "expires_in": 1199
	}

Notes	"expires_in" is the time in seconds until the access token expires

2.8.   Information and Error Messages

Information messages will result in a 200 HTTP response code. The message informs the client that the response was valid, but something happened for which they should be notified.

Information Messages are returned in a response with an HTTP status code of 200. The messages array is included at the top level of an Order response. The level of the message will be set to "information". Error messages are returned in a response with an HTTP status code of a non-200 range response. The level of the message will be set to "error".

3. Inventory API

NOTE: On Order stock values are awaiting implementation and are currently not available (see 3.6.6.)

3.1. Reference

The API is RESTful, and provides consumers with the ability to query Inventory by SKU, ProductGroup-Colour or ProductGroup. This API is meant to be used by authorised services running on behalf of Ralawise customers, and all operations expect valid credentials to be provided in the form of a JSON Web Token or JWT (within the Authorization header of the request).

The Inventory API is called using the Ralawise SKU codes or part thereof, which are broken down in the following examples:

Full SKU	Product Group	Colour	Size	Notes
JH001JBLAL	JH001	JBLA	L	Full SKU (up to 20 chars), Product Group (5 chars), Colour (4 chars), Size (up to 11 chars)
GD001BLACXS	GD001	BLAC	XS	As above

A full list of Ralawise SKUs and other product information is available in the Website Data from the Marketing Hub on the Ralawise website (https://shop.ralawise.com/marketing-hub/essential-resources/web-data/).

3.2. Authorization and Authentication

The caller is expected to provide a valid JWT within the Authorization request header. All resources require this token and will return a 401 response if the API access key is not provided.

3.3. Request Headers

Field	Type	Length	Occurrence	Notes
Authorization	String	-	1	The Authorization header requires a valid bearer token (JWT). Users of this endpoint will be expected to have already authenticated via the Login endpoint. Expected format: `Bearer <JWT>`
Content-Type	String	-	1	application/json
Accept-Encoding	String	-	0.1	Used for HTTP compression: - Accept-Encoding: gzip, deflate

3.4. Response Headers

Field	Type	Length	Occurrence	Notes

3.5. Query String Parameters

Field	Type	Length	Occurrence	Notes

3.6. Models

The skeletal structure for an inventory response is as follows:

```
{
 "productGroup": {
  "id": "GD001",
  "name": "Softstyle™ adult ringspun t-shirt",
  "products": [
   {
     "productCode": "GD001BLAC",
     "colourCode": "BLAC",
     "colourDescription": "Black",
     "variants": [
       {variant},
       {variant}
     ],
   },
   {
     "productCode": "GD001GREE",
     "colourCode": "GREE",
     "colourDescription": "Green",
     "variants": [
       {variant},
       {variant}
     ],
   }

  ]
 }
}
```

The following product hierarchy is followed throughout this document and the API definition:

Level	Contains	Description
Product Group	Products	A family of products – available in a range of colours and sizes. E.g. "GD001"
Product	Variants	Product with a specified colour. E.g. "GD001GREE", which is "GD001" with a specific colour of "GREE" (green)
Variant	-	Individual SKU items – Products with a specified size and colour. E.g. "GD001BLACM", which is "GD001" with a specified colour of "BLAC" (black) in size "M" (medium)

Every response from the Get Stock endpoint will return a Product Group, containing at least one Product, itself containing at least one Variant.
If a Product Group is requested, then all child Products and their Variants will be returned.
If a Product (Product Group and Colour combination) is requested, then only that Product and its child Variants will be returned.
If a SKU is requested, then only that Variant and its parent Product will be returned in the Product Group.

3.6.1. Product Group

Property Name	Type	Length	Occurrence	Notes
id	String	20	1	Identifier for the Product Group
name	String	100	1	Friendly name for the Product Group
products	Array of Products	-	1	Individual Products within the Group. NB if the Product is specified in the request (either by providing a ProductGroup-Colour or a Variant in the

Property Name	Type	Length	Occurrence	Notes
				request) there will only be one Product returned in the array.

3.6.2. Product

Property Name	Type	Length	Occurrence	Notes
productCode	String	20	1	Identifier for the Product Group-Colour
colourCode	String	20	1	Identifier for the colour
colourDescription	String	20	0.1	Friendly name for the colour
variants	Array of Variants	-	1	Individual Variants. NB if the SKU is specified in the request (either by providing a ProductGroup-Colour or a Variant in the request) there will only be one Variant returned in the array.

3.6.3. Variant

Property Name	Type	Length	Occurrence	Notes
sku	String	20	1	Lowest level identifier for the product variant (including size and colour)
sizeCode	String	20	1	Identifier for the variant size
sizeDescription	String	100	0.1	Friendly name for the variant size
packageUnit	Package Unit object	-	1	Product item composition for single and bulk units
availableStock	Available Stock object	-	1	Current saleable stock levels, broken down by unit (carton, pack, single)
onOrderStock	On Order Stock object	-	0.1	Stock quantity currently on order, and expected availability
supplierStock	Suppler Stock object	-	0.1	Stock quantity further up the supply chain (and lead time)

Example

```
{
 "sku": "GD001BLACS",
 "sizeCode": "S",
 "sizeDescription": "Small",
 "packageUnit":
 {
  "carton": 72,
  "pack": 12,
  "single": 1
 },
 "availableStock":
 {
  "quantity": 134
 },
 "supplierStock":
 {
  "quantity": 21600,
  "leadTime": "+5 days"
 }
}
```

3.6.4. Package Unit

Property Name	Type	Length	Occurrence	Notes
carton	Int	-	1	The number of product items that constitute a carton for this variant
pack	Int	-	1	The number of product items that constitute a pack for this variant
single	Int	-	1	The number of product items that constitute a single saleable unit for this variant

Example

```
"packageUnit":
{
  "carton": 72,
  "pack": 12,
  "single": 1
}
```

3.6.5. Available Stock

Property Name	Type	Length	Occurrence	Notes
quantity	Int	-	1	The currently available quantity (total)

Example

```
"availableStock":
{
  "quantity": 134
}
```

3.6.6. On Order Stock

NOTE: On Order stock values are awaiting implementation and are currently not available

Property Name	Type	Length	Occurrence	Notes
quantity	Int	-	1	The quantity currently on back order.
expectedArrivalDate	ISO 8601 date format	-	1	The expected date that the stock will be received by the warehouse. Example: "2021-02-23"

3.6.7. Supplier Stock

Property Name	Type	Length	Occurrence	Notes
quantity	Int	-	1	The quantity of stock held by the supplier.
leadTime	String	20	1	The maximum expected lead time for stock be received from the supplier into the warehouse. Example: "+5 days"

Example

"supplierStock":
{
 "quantity": 19000,
 "leadTime": "+5 days"
}

3.6.8. Error / Information Messages

Property Name	Type	Length	Occurrence	Notes
errorCode	String	100	1	A string error code
parameterName	String	500	0.1	The parameter name of the property the validation message applies to from the request
Level	String	10	1	Set as "information" for all 200 range responses. Set as "error" for all non-200 range responses"
errorMessage	String	500	1	Informational message returned to aid user support

3.7. Resources

3.7.1. Get Stock (GET)

Get Stock					
Description	Retrieves the current stock level for a specified Product Group, Product (i.e. Product Group and Colour combination) or Variant.				
URL	https://api.ralawise.com/v1/inventory/{identifier}				
Verb(s)	GET				
Input Headers	**Name**	**Type**	**Length**	**Occurrence**	**Notes**
	Authorization	String	100	1	API access key (as provided by Ralawise)
Input Route	**Name**	**Type**	**Length**	**Occurrence**	**Notes**
	Identifier	String	20	1	Can be a Product Group identifier, a Product (Product Group & Colour) identifier, or a SKU
Input Query String	None				
Input Body	None				
Defaults	None				
Outputs	Product Group model				
Output Headers	None				
Response Codes	200 (OK) – The request inventory details are provided 401 (Not Authorized) – The operation was called with invalid credentials 404 (Bad Request) – The product group, product or variant requested was not found 500 (Internal Server Error) – An unexpected error occurred				
Bad Request Validation Messages	**Property Name**	**Condition**	**Input**	**Code**	**Message**
	identifier	Missing	Route	identifierRequired	Inventory requests require a product identifier
Information Messages	None				
Error Messages	**Code**		**HTTP Status Code**	**Explanation**	
	NotAuthorised		401	The operation was called with either no token or an invalid token	
Events Raised	None				
Example Input	https://api.ralawise.com/v1/inventory/GD001BLACL				

Example Output	{
	  "productGroup": {
	    "id":"GD001",
	    "name":"Softstyle™ adult ringspun t-shirt",
	    "products": [				

```
                    {
                      "productCode":"GD001BLAC",
                      "colourCode":"BLAC",
                      "colourDescription":"black",
                      "variants": [
                        {
                          "sku":"GD001BLACL",
                          "sizeCode":"L",
                          "sizeDescription":"Large",
                          "packageUnit": {
                            "carton":72,
                            "pack":12,
                            "single":1
                          },
                          "availableStock": {
                            "quantity":782
                          },
                          "supplierStock": {
                            "quantity":21600,
                            "leadTime":"+5 days"
                          }
                        }
                      ]
                    }
                  ]
                }
              ]
            }
          }
```

Notes	The response will always include a Product Group, containing at least one Product, itself containing at least one Variant.
	If the request is made at the SKU level, only the Variant object containing the SKU and the Product containing the Variant are returned.
	If the request is made at the Product level, all the Variants within the Product are returned.
	If the request is made at the Product Group level, all the Products and all the Variants within the Product Group and returned

3.8. Information and Error Messages

Information messages will result in a 200 HTTP response code. The message informs the client that the response was valid, but something happened for which they should be notified.

Information Messages are returned in a response with an HTTP status code of 200. The messages array is included at the top level of an Order response. The level of the message will be set to "information". Error messages are returned in a response with an HTTP status code of a non-200 range response. The level of the message will be set to "error".

The format of the message is the same for both information and error messages and is detailed below.

Request Validation Example (the request is poorly-formed, no inventory matches for identifier were found)

```
"Messages": [
  {
    "errorMessage": "Inventory data not found",
    "parameterName": "identifier",
    "level": "information",
```

```
        "errorCode": "NotFound"
      }
   ]
}
```

4. Order API

4.1.   Reference

The API is RESTful, and provides consumers with the ability to create Orders. This API is meant to be used by authorised services running on behalf of Ralawise customers, and all operations expect valid credentials to be provided in the form of Basic Authentication (within the Authorization header of the request).

4.2.   Authorization and Authentication

The caller is expected to provide a valid JWT within the Authorization request header. All resources require this token and will return a 401 response if the API access key is not provided.

4.3.   Request Headers

Field	Type	Length	Occurrence	Notes
Authorization	String	-	1	The Authorization header requires a valid bearer token (JWT). Users of this endpoint will be expected to have already authenticated via the Login endpoint. Expected format: `Bearer <JWT>`
Content-Type	String	-	1	application/json
Accept-Encoding	String	-	0.1	Used for HTTP compression: - Accept-Encoding: gzip, deflate

4.4.   Response Headers

Field	Type	Length	Occurrence	Notes

4.5.   Query String Parameters

Field	Type	Length	Occurrence	Notes

4.6. Models

The skeletal structure for an order request is as follows. Significant models in the object are highlighted in yellow and described separately:

```
{
 "order": {
  "contactName": "John Smith",
  "contactPhone": "07712 345678",
  "orderReference": "TCHS-Leavers-21",
  "acceptTerms": true,
  "lineItems": [
     {line Item},
     {line Item},
     {line Item}
  ],
  "delivery": {
   "deliveryAddress":{},
   "deliveryOption": {}
  }
 }
}
```

4.6.1. Order

Property Name	Type	Length	Occurrence	Notes
contactName	String	250	0.1	Name of the customer submitting the order
contactPhone	String	250	0.1	Contact number for the customer
orderReference	String	20	0.1	Customer's order reference / PO number
orderNotes	String	250	0.1	Customer's order notes
orderTags	String	250	0.1	Customer's tags for the order – comma-delimited list to aid searching past orders on website
specialInstructions	String	8000	0.1	Special instructions for the order
promotionalCode	String	100	0.1	Promotional code to be applied to the order – comma-delimited list for multiple promotions
acceptTerms	Boolean	-	1	Flag to indicate if terms of sale are accepted. No default – value must be set to `true`
lineItems	Array	-	1.n	Individual order lines.
Delivery	Object	-	1	Delivery address for the order, and carriage and packaging options. See 5.1

4.6.2. Line Item

Property Name	Type	Length	Occurrence	Notes
sku	String	20	1	Lowest level identifier for the product variant (including size and colour)
quantity	Int	-	1	Quantity of SKUs to be ordered
orderLineRef	String	15	0.1	Customer reference for the line item
autoBackOrder	Boolean	-	0.1	Flag to indicate if OOS items should be automatically back ordered (or if the shortage should be reported to the customer so they can source elsewhere). `true` indicates OOS items should be back ordered.

Property Name	Type	Length	Occurrence	Notes
				`false` indicates OOS items should be flagged back to the customer once the order is accepted. When autoBackOrder is to false, the "quantity" in the api response will be adjusted to match that of the allocated (available) quantity. Default is `false`

Example

```
{
    "sku": "GD057INBLS",
    "quantity": 24,
    "orderLineRef": "TCHS-Leavers-Small",
    "autoBackOrder": false
}
```

4.6.3. Delivery Address

Property Name	Type	Length	Occurrence	Notes
deliveryAccountName	String	250	1	The recipient's name (can be the company name)
addressLine1	String	80	1	
addressLine2	String	80	0.1	
townCity	String	64	1	
postcode	String	20	1	
countryName	String	30	1	Free-text country name
countryCode	ISO 3166 alpha-2 country code	2	1	e.g. "GB", "IE", "FR"

Example

```
"deliveryAddress": {
    "deliveryAccountName": "John Smith at School Leaver Hoodies Ltd ",
    "addressLine1": "Zone 3, Deeside Industrial Park",
    "addressLine2": "Unit 112 Tenth Avenue",
    "townCity": "Deeside",
    "postcode": "CH5 2UA",
    "countryName": "England",
    "countryCode": "GB"
}
```

4.6.4. Delivery Option

Property Name	Type	Length	Occurrence	Notes
plainCover	Boolean	-	0.1	Whether the order should be sent unbranded with no mention of Ralawise. Default is `false`
deliveryMethod	String	10	0.1	The carriage method for the delivery. Default is "UKMAIN-UKNBD" – UK mainland, standard delivery. Please see list of available carriage methods in Appendix 5.1 UK List of Carriage Methods.

Example

```
"deliveryOption": {
     "plainCover": false,
     "deliveryMethod": "UKMAIN-UKNBD"
 }
```

4.6.5. Totals (Response Only)

Property Name	Type	Length	Occurrence	Notes
subTotal	Decimal	-	1	Value of goods in the order (net of VAT)
vatTotal	Decimal	-	1	Value of VAT for goods in the order
deliveryTotal	Decimal	-	1	Cost of delivery for the entire order
discountTotal	Decimal	-	0.1	Value of discount applied to the order
orderTotal	Decimal	-	1	Gross value of goods in the order (subTotal + deliveryTotal + VAT - discountTotal)
orderQuantityTotal	Int	-	1	Total number of items ordered
allocatedQuantityTotal	Int	-	1	Total number of items allocated

Example

```
"totals": {
  "subTotal": 352.8,
  "vatTotal": 74.08,
  "deliveryTotal": 13.2,
  "orderTotal": 440.08,
  "orderQuantityTotal": 48,
  "allocatedQuantityTotal": 48
}
```

4.7.   Error / Information Messages

Property Name	Type	Length	Occurrence	Notes
errorCode	String	100	1	A string error code
parameterName	String	500	0.1	The parameter name of the property the validation message applies to from the request
Level	String	10	1	Set as "information" for all 200 range responses. Set as "error" for all non-200 range responses"
errorMessage	String	500	1	Informational message returned to aid user support

4.8. Resources

4.8.1. Create Order (POST)

CREATE ORDER					
Description	Places an order on Ralawise's web platform.				
URL	https://api.ralawise.com/v1/order				
Verb(s)	POST				
Input Headers	**Name** Authorization	**Type** String	**Length** 100	**Occurrence** 1	**Notes** Valid JWT token
Input Route	None				
Input Query String	None				
Input Body	**Name** Order	**Type** Object	**Length** -	**Occurrence** 1	**Notes** Order details payload
Defaults	None				
Outputs	Detailed Order details model				
Output Headers	None				
Response Codes	201 (Accepted) – The order request was accepted 400 (Bad Request) – The request resulted in an error 401 (Not Authorized) – The operation was called with invalid credentials 500 (Internal Server Error) – An unexpected error occurred				
Bad Request Validation Messages	**Property Name** contactName lineItems lineItem.sku lineItem.quantity deliveryAddress. addressLine1 deliveryAddress. townCity deliveryAddress. postcode deliveryAddress. countryCode	**Condition** Missing Missing Missing Missing Missing Missing Missing Missing	**Input** Body Body Body Body Body Body Body Body	**Code** contactNameRequired lineItemsRequired skuRequired quantityRequired addressLine1Required townCityRequired postcodeRequired countryCodeRequired	**Message** Contact Name is missing. This must be included on an order. Order requires at least one line item Line Items require a SKU Line Items require a quantity Delivery Address requires address line 1 Delivery Address requires town city Delivery Address requires postcode Delivery Address requires country code
Information Messages	**Code** SkuNotFound	**HTTP Status Code** 200		**Explanation** The SKU requested (within the Order line item) was not found in the Ralawise product catalogue	
Error Messages	**Code** NotAuthorised	**HTTP Status Code** 401		**Explanation** The operation was called with either no token or an invalid token	
Events Raised	None				

Example Input	{
	   "contactName": "John Smith",
	   "contactPhone": "07712 345678",
	   "orderReference": "MyOrderRef",
	   "acceptTerms": true,
	   "promotionalCode": "FREEDELIVERY,NEWCUSTOMER",
	   "orderNotes": "Order Notes",				

	``json
	      "orderTags": "Order Tags",
	      "specialInstructions": "Special Instructions",
	      "lineItems": [
	          {
	        "sku": "GD057INBLS",
	        "quantity": 100,
	        "orderLineRef": "TCHS-Leavers-S",
	        "autoBackOrder": false
	       }
	     ],
	      "delivery": {
	       "deliveryAddress": {
	        "deliveryAccountName": "John Smith",
	        "addressLine1": "Zone 3, Deeside Industrial Park",
	        "addressLine2": "Unit 112 Tenth Avenue",
	        "townCity": "Deeside",
	        "postcode": "CH5 2UA",
	        "countryCode": "GB",
	        "countryName": "Wales"
	       },
	       "deliveryOption": {
	        "plainCover": true,
	        "deliveryMethod": "UKMAIN-UKNBD"
	       }
	      }
	     }
	    }
	``

Example Output	```json
	{
	   "order": {
	     "orderNumber":"20368068",
	     "contactName": "John Smith",
	     "contactPhone": "07712 345678",
	     "contactEmail": "api.user@company.com",
	     "orderNotes": "Order Notes",
	     "orderTags": "Order Tags",
	     "orderReference": "MyOrderRef",
	     "specialInstructions": "Special Instructions",
	     "promotionalCode": "",
	     "acceptTerms": true,
	     "createdDate": "2021-05-05T11:23:12",
	     "lineItems": [
	       {
	         "sku": "GD057INBLS",
	         "brand": "Gildan",
	         "title": "Heavy Blend™ hooded sweatshirt",
	         "productCode": "GD057",
	         "colourCode": "INBL",
	         "colourDescription": "Indigo Blue",
	         "size": "S",
	         "orderLineRef": "TCHS-Leavers-S",
	         "quantity": 96,
	         "allocatedQuantity": 96,
	         "quantityBreak": "CTN",
	         "currency": "GBP",
	         "unitPrice": 7.35,
	         "linePrice": 705.60,
	         "vat": 162.29
	       },
	       {
	         "sku": "GD057INBLS",
	         "brand": "Gildan",
	         "title": "Heavy Blend™ hooded sweatshirt",
	         "productCode": "GD057",
	         "colourCode": "INBL",
	         "colourDescription": "Indigo Blue",
	         "size": "S",
	         "orderLineRef": "TCHS-Leavers-S",
	         "quantity": 4,
	         "allocatedQuantity": 4,
	         "quantityBreak": "SIN",
	         "currency": "GBP",
	         "unitPrice": 8.15,
	``

```
            "linePrice": 32.60,
            "vat": 7.50
          }
        ],
        "delivery": {
          "deliveryAddress": {
            "deliveryAccountName": "John Smith ",
            "addressLine1": "Zone 3, Deeside Industrial Park",
            "addressLine2": "Unit 112 Tenth Avenue",
            "townCity": "Deeside",
            "postcode": "CH5 2UA",
            "countryCode": "GB",
            "countryName": "Wales"
          },
          "deliveryOption": {
            "plainCover": true,
            "deliveryMethod": "UK Mainland Standard Delivery (1 - 2 Day Despatch)"
          }
        },
        "totals": {
          "subTotal": 738.20,
          "vat": 169.79,
          "discount": 0.00,
          "delivery": 0.00,
          "orderQuantity": 100,
          "allocatedQuantity": 100,
          "total": 907.99
        }
      },
      "Messages": []
    }
```

Notes	When successful, the response will include an enriched order object. The order will include an "orderReference" value, and a timestamp to indicate when the order was accepted ("createdDate"). The response will return each line item split out into cartons, pack and single quantities. In the example output above, a quantity of 100 was ordered. 96 were charged at the carton rate, and 4 at the single rate. Each line item will include the descriptive properties (brand, title, colour code, colour description, size) and price information (qty break, currency, unit price, line price and Vat). A totals object will include the rolled-up costs of the order as confirmed.

4.9.   Information and Error Messages

Information messages will result in a 200 HTTP response code. The message informs the client that the response was valid, but something happened for which they should be notified.

Information Messages are returned in a response with an HTTP status code of 200. The messages array is included at the top level of an Order response. The level of the message will be set to "information". Error messages are returned in a response with an HTTP status code of a non-200 range response. The level of the message will be set to "error".

The format of the message is the same for both information and error messages and is detailed below.

Information Message Example (the request is well-formed, but the line item is not available in the catalogue)

```
  "messages":[
    {
      "errorCode":"SkuNotFound",
      "errorMessage":"The SKU requested (GD057INBLXS) was not found in the Ralawise product
catalogue",
      "level":"information",
      "parameterName": "SKU"
    }
  ]
```

Allocation Failure Example (the request is well-formed, but the ordered quantity cannot be allocated)

```
  "messages":[
    {
      "errorCode":"UnableToAllocateSku",
      "errorMessage":"The SKU requested (GD057INBLS) was not found in sufficient quantity to satisfy the order. Ordered:
10, allocated: 7",
      "level":"information",
      "parameterName": "SKU"
    }
  ]
```

The allocation failure information message would occur if the customer requested the "autoBackOrder" attribute was set to `false` for the specific line item, and the number available in stock fell short of the requested amount. When "autoBackOrder" is set to false, the "quantity" on a line item will be adjusted to match that of the allocated quantity.

Request Validation Example (the request is poorly-formed, as there is a missing required field)

```
  "messages":[
    {
      "errorCode":"ContactNameRequired",
      "errorMessage":"An order requires a named customer",
      "level":"error",
      "parameterName": "contactName"
    }
  ]
```

5. Appendix

5.1.   List of UK Carriage Methods

deliveryMethod	Description
UKMAIN-COL	UK Customer Collection from Ralawise
UKMAIN-UKNBD	UK Mainland Standard Delivery
UKMAIN-12P	UK Mainland Pre-12pm Delivery
UKMAIN-PML	Northern Ireland (and ROI) Standard Delivery

For further information on delivery information and costs please see the Ralawise website (https://shop.ralawise.com/footer-pages/delivery-information/)