

Overview

This document details the specification for implementing Custom/Bespoke shipping integration module for Linnworks.net and deploying it as Application Extension.

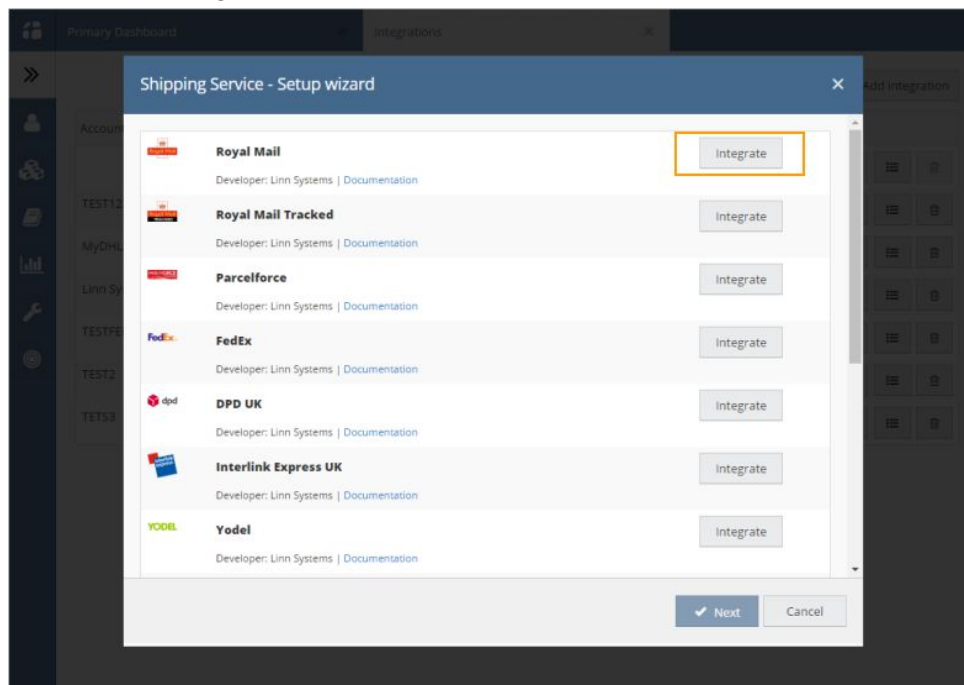
External shipping integration module allows you to integrate courier labeling solution into linnworks.net as an external application, which feels and looks like a native integration within Linnworks.net

Concept

There are basic 4 components of the integration we need to grasp in order to develop bespoke courier labeling integration in linnworks.net

Setup and configuration

This is how the user adds an integration into linnworks.net. From the user perspective, the user will navigate to Shipping Integrations, clicks Add Integration, chooses the courier integration from the list and clicks Integrate.



Linnworks will then connect to the courier integration adapter and request first stage of the wizard, and display the User Interface for entering some specific parameters required for the completing this wizard stage on the screen for user to enter.

Shipping Service - Configuration - Royal Mail - 23423

Enter your OBA account details

Please enter your account details.

Please note: once you click next, your account details will be passed to Royal Mail for them to continue processing your account integration. Until they have completed their processing you will not be able to cancel this request.

It usually takes 3 working days for your details to be processed and for your account to be activated. You will receive an email when your account is ready.

Account information

Account Number:

Address

Contact Name:

Business Name:

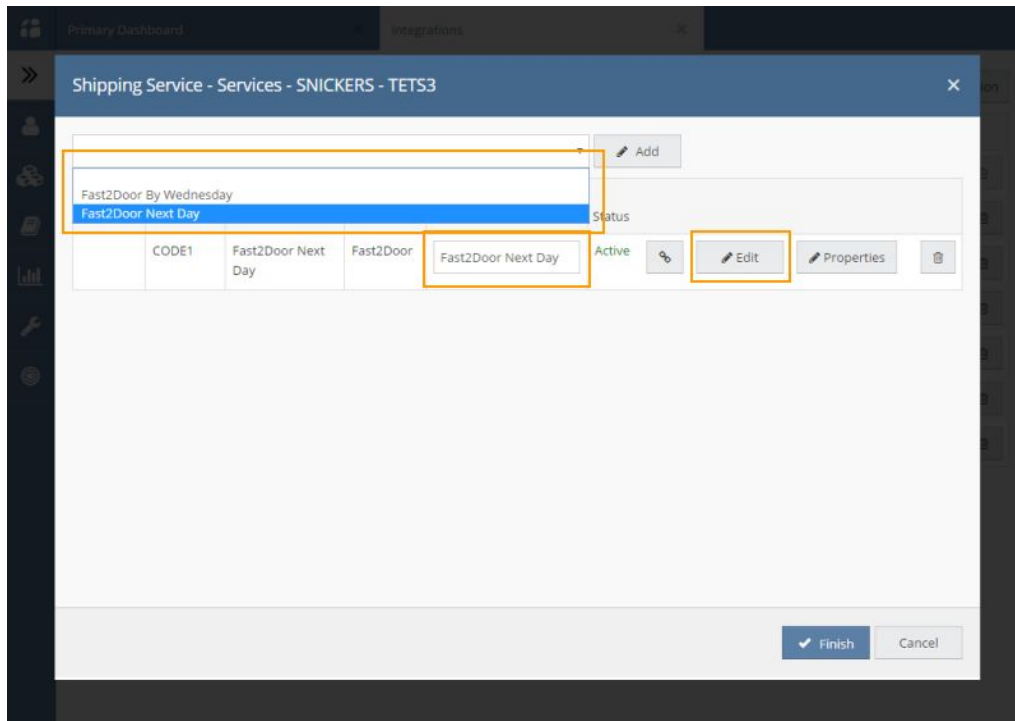
Address 1:

Next Cancel

When user clicks Next Stage, the entered values are being sent to the courier integration adapter and evaluated. The adapter then returns the next stage and so on, until the user reaches the final stage. On the final stage the integration config is being turned into Active and the user can use the integration.

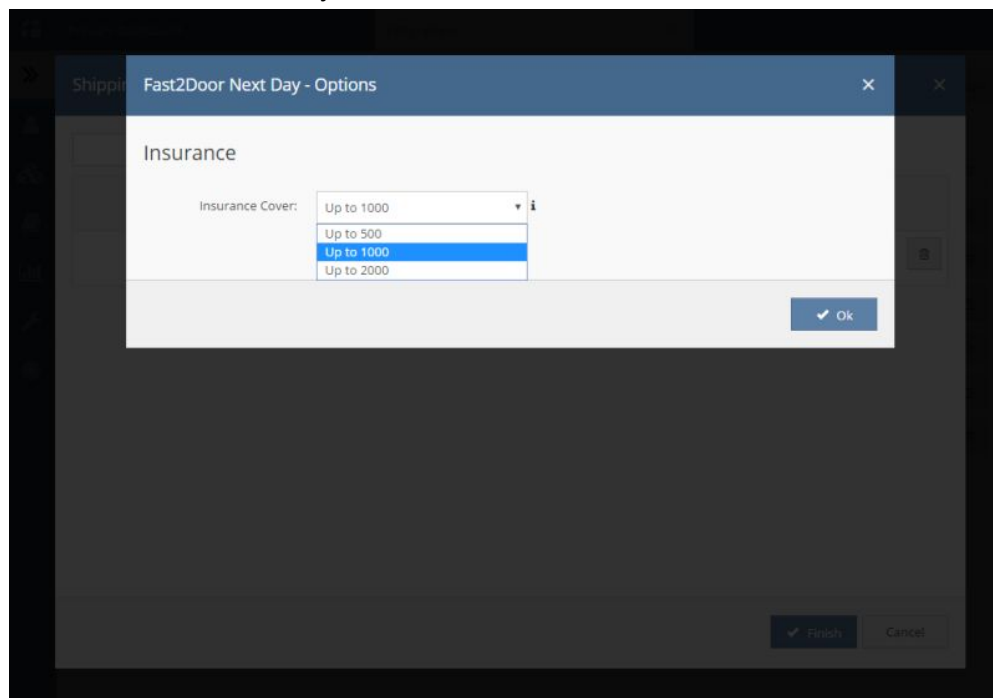
Service Configuration

Each courier integration has a list of courier services that the user can print labels for. The user has to add the services they want to use and bind them to a Linnworks.net Shipping Service name.



When an order in Linnworks.net is assigned to a shipping service which is allocated to a courier integration service, Linnworks.net will use this courier integration to print a label.

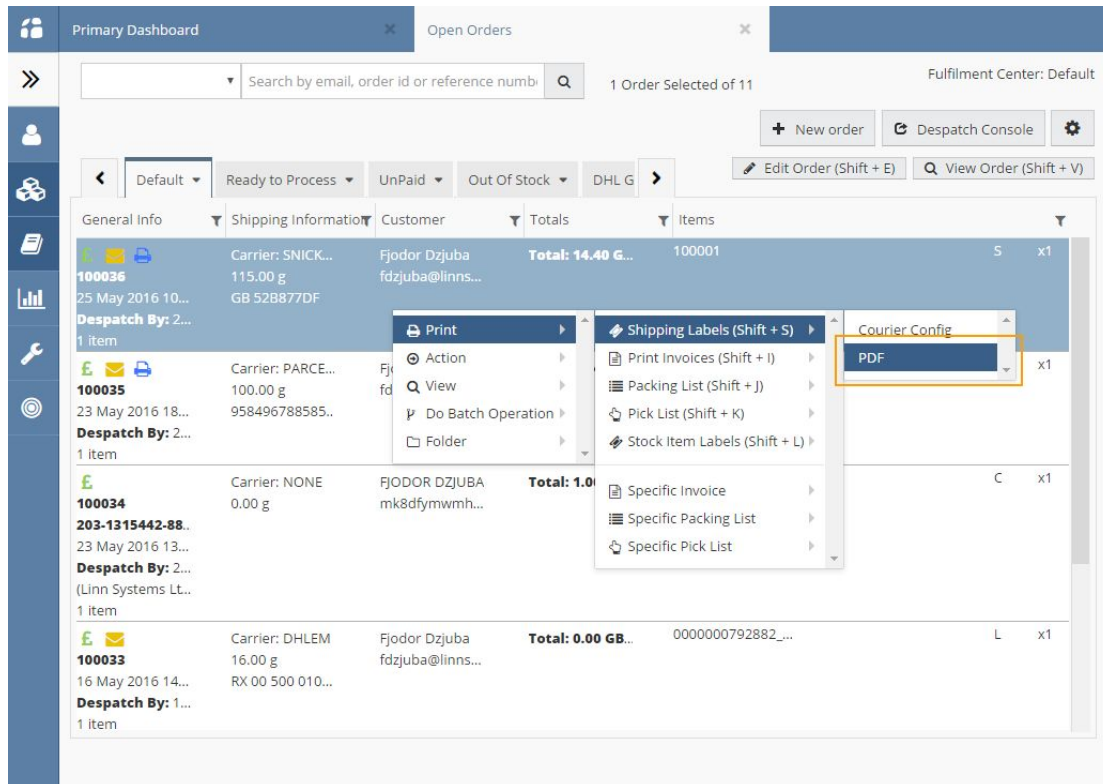
Each service has a list of configuration values. For example, whether SMS Notification enhancement is required, level of insurance etc. The user can change these configuration values for each service individually.



Each service also has a list of static property values. These are for the information only and cannot be changed.

Generating a label and documentation

When user Prints a shipping label from within linnworks. Linnworks takes the currently allocated shipping service, finds the courier integration this service is allocated for and compiles a consignment data from the order and sends this to courier integration module to produce a label.



The courier integration module returns an Image of the label and an additional documentation in PDF format. Linnworks then prints the label in PDF format or sends it directly to the Virtual Printer.

User can cancel label, in which case it sends Cancel Label request specifying linnworks.net orderid as identification of what label must be canceled.

Manifest/End-of-day

The user is required to complete consignment batch by Filing a manifest at least every 24 hours. This process simply takes all currently unmanifested consignments and puts them a

single manifest. If courier integration requires the pre-advice or manifest file to be uploaded to the courier system, the integration compiles such manifest and sends it over to the courier.

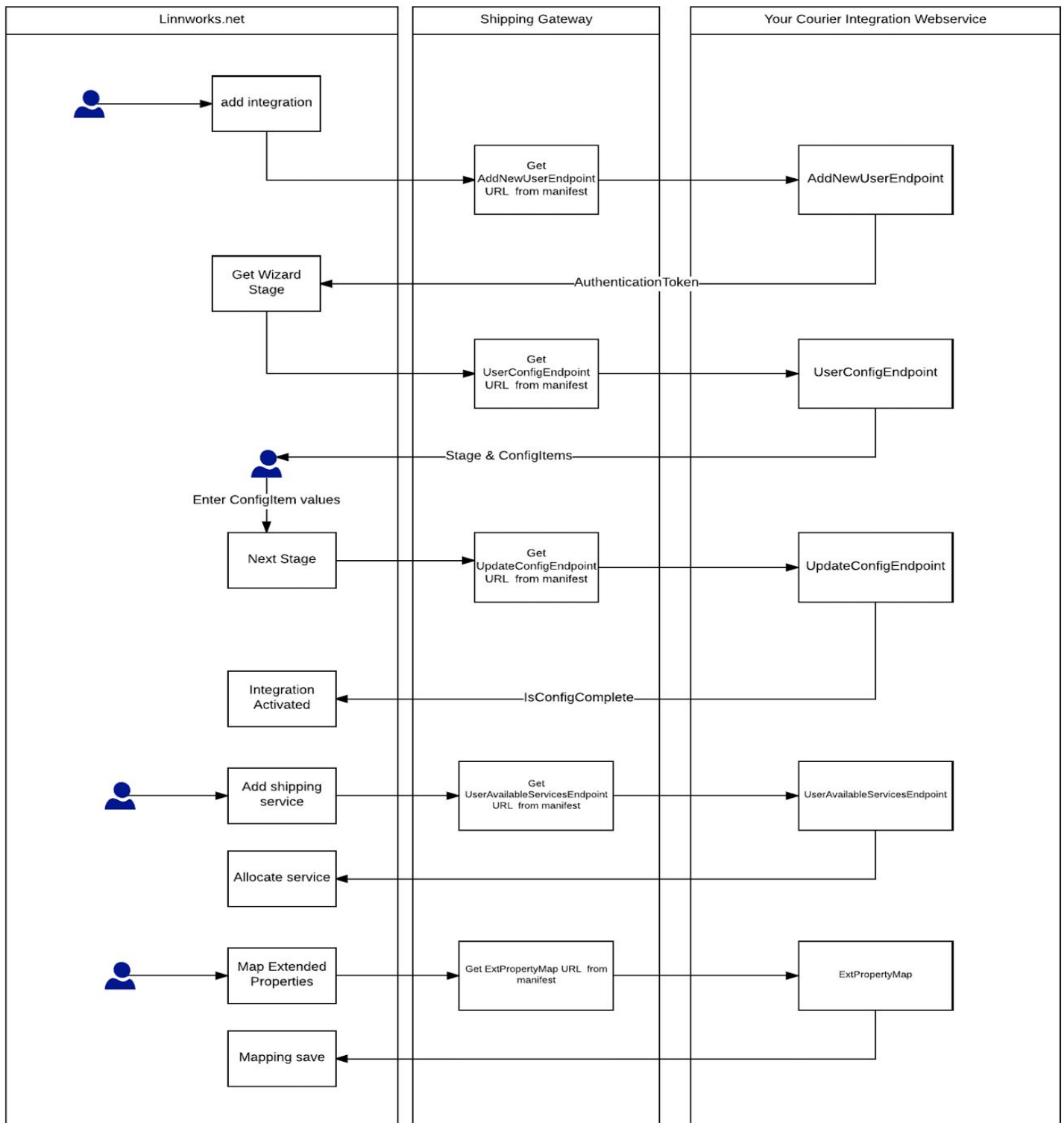
In addition if there is a specific manifest documentation the courier integration module can also provide PDF file for each manifest.

How it works

In order to create custom/ bespoke integration you have to describe to linnworks.net where to send requests for data, i.e. wizard steps, list of services available, request to generate a label and generate manifest request. We can do this by deploying Extension Manifest which contains ShippingIntegration module. Linnworks then uses this manifest to request wizard steps to create integration, get the list of available services and where to send the request to generate a label.

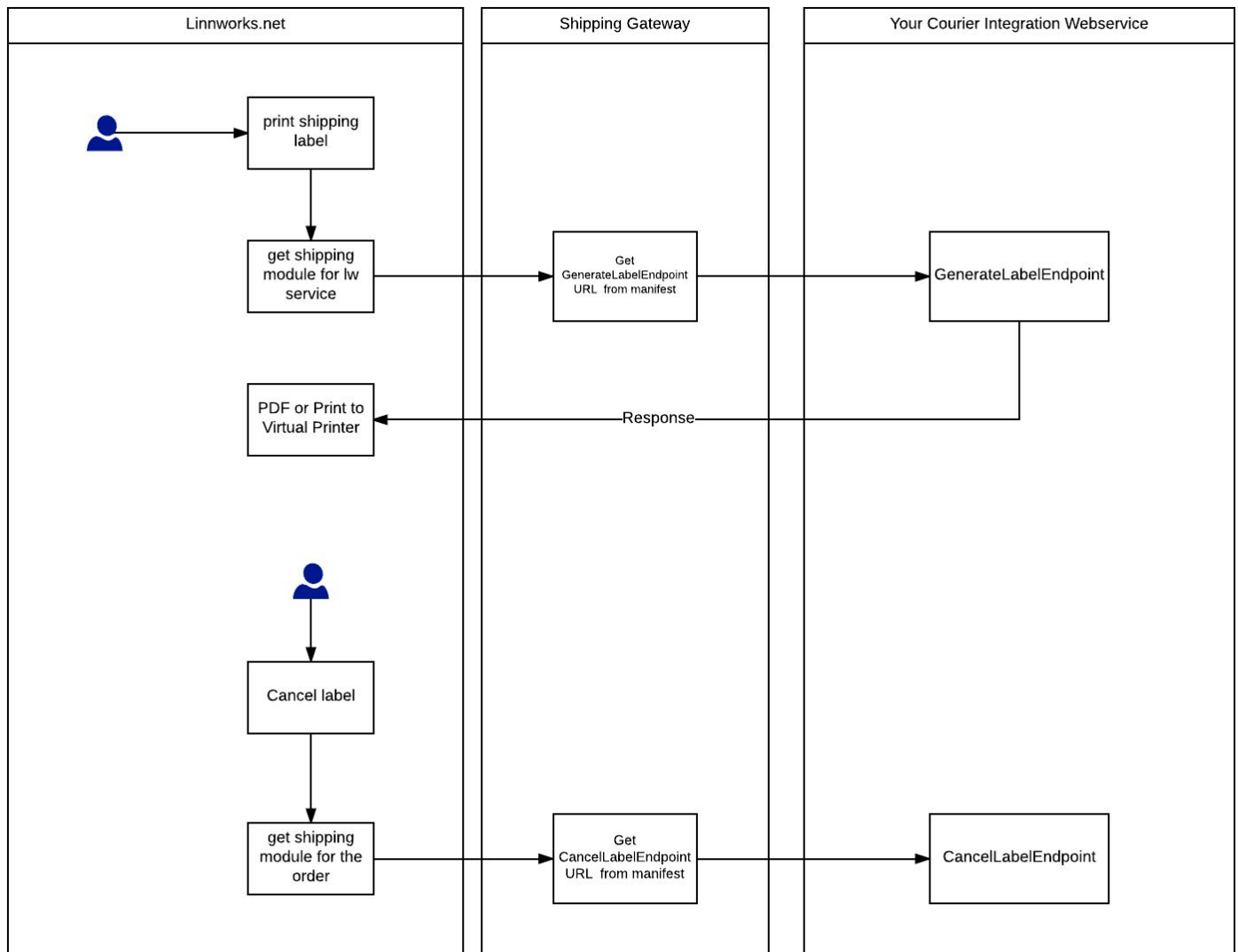
These webservice endpoints expect the data from linnworks in a specific format and returns data in a very specific format. Therefore you have developed this webservice specifically for linnworks to send data. In case of courier integration which already has an API, this webservice will simply act as a proxy middleware which translates the data from linnworks into what is expected by the courier API and back again.

Integrating new account process flow



- User Add integration
- Linnworks gets the extension manifest for the Shipping Integration and retrieves the URL for AddNewUserEndpoint
- Linnworks sends request to AddNewUserEndpoint, the Custom integration webservice will create new new account and issue a token id. From now on linnworks will be sending this authorization token (guid) with all the requests. This is how your integration identifies the user.
- Once linnworks has authorization token, it will send request to UserConfigEndpoint to get the first registration wizard stage with config items
- User enters the config item values and linnworks sends a request containing all config item values to UpdateConfigEndpoint.
- UpdateConfigEndpoint validates the config items and responds with errors if any of the config items values are incorrect.
- Linnworks will then request UserConfigEndpoint to get the new status of the registration from the custom integration webservice. If the registration status is marked as completed, linnworks will add the integration into the system. Alternatively it will proceed to the next stage of the integration wizard.
- Once the integration configuration is activated, the user can add shipping services and allocated them to shipping services defined in linnworks.net
- User clicks add shipping services, linnworks.net get the URL for UserAvailableServicesEndpoint and sends a request to the webserivce.
- Webservice responses with the list of available services.
- User adds services in linnworks.net. **The service is uniquely identified by ServiceUniqueld**
- The shipping integration may require some bespoke/specific values to be specified for each consignment. For example, SafePlace message that will appear on the label, or customs clearance information for each item in the consignment. This can be done by mapping extended property of an order or extended property of an item to properties expected by the couriers. When linnworks sends a consignment request it will automatically retriive the values for the extended property mapping from order items or order extended properties and send these values along with the consignment request.

Generate Label



- User prints a shipping label in linnworks.net
- Linnworks.net matches the shipping service assigned to the order and gets the courier integration module associated with the selected service
- Shipping module in linnworks gets the URL states in the Extension manifest for generating label
- Sends the request which contains consignment object to GenerateLabelEndpoint
- Gets the response back and prints the label using Virtual Printer or PDF

Webservice Specification

This section details every endpoint, defines structure of the request and response class.

Every endpoint is a POST method accepting JSON structure.

If your webservice on HTTPs, the certificate must be signed by minimum tier 2 certification authority. SSL 1.1, 1.2 and TLS are supported.

AddNewUserEndpoint														
Create new integration for a user and default configuration profile for the integration. Note that the same customer may have multiple integrations with the same courier.														
Request														
<table><tr><th>Field</th><th>Type</th><th>Description</th></tr><tr><td>Email</td><td>string</td><td>Email address of the customer requesting the integration</td></tr><tr><td>LinnworksUniqueIdentifier</td><td>Guid</td><td>Linnworks Unique Identifier of a user. Each LW account has a unique id.</td></tr><tr><td>AccountName</td><td>string</td><td>Name of the account the user entered in the integration</td></tr></table>			Field	Type	Description	Email	string	Email address of the customer requesting the integration	LinnworksUniqueIdentifier	Guid	Linnworks Unique Identifier of a user. Each LW account has a unique id.	AccountName	string	Name of the account the user entered in the integration
Field	Type	Description												
Email	string	Email address of the customer requesting the integration												
LinnworksUniqueIdentifier	Guid	Linnworks Unique Identifier of a user. Each LW account has a unique id.												
AccountName	string	Name of the account the user entered in the integration												
Example request														
<pre>{ "Email": "fedor_demo@linnsystems.com", "LinnworksUniqueIdentifier": "4A287F98-D1D3-434D-8D11-3159ABD094E3", "AccountName": "ACC1" }</pre>														
Response														

Field	Type	Description
AuthorizationToken	Guid	Authorization token of the newly created integration profile. All further requests associated with this integration must have this authorization token
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message
Response Example		
<pre>{ "AuthorizationToken":"c11c8f63-b362-4548-9af6-b76914afb0cb", "IsError":false, "ErrorMessage":null }</pre>		

UserConfig		
Gets the current stage of the user config. The service method returns current stage of the config, description, title and config items the user has to enter. Linnworks will then render this config items on the UI and will send the config item list to the UpdateConfig		
Request		
Field	Type	Description
AuthorizationToken	Guid	Authorization token uniquely identifying each integration profile
Example request		

```
{
  "AuthorizationToken": "c11c8f63-b362-4548-9af6-b76914afb0cb"
}
```

Response

Field	Type	Description
IsConfigActive	true/false	Identifies whether the integration profile is in Active mode. Meaning the customer completed the integration wizard and can use the integration.
ConfigStatus	string	Configuration stage name. You must provide the same name to the UpdateConfig endpoint.
ConfigStage	class	Config stage class
ConfigStage.WizardStepDescription	string	Description of the configuration stage
ConfigStage.WizardStepTitle	string	Title of the configuration stage
ConfigStage.ConfigItems	Array of ConfigItems	List of ConfigItems
ConfigStage.ConfigItems[n].ListValues	Array of ListValue	List of ListValues. The config item ValueType =5 (LIST) can have a list of available values the user can select from the drop down list
ConfigStage.ConfigItems[n].ListValues[m].Display	string	Value displayed in the drop down list
ConfigStage.ConfigItems[n].ListValues[m].Value	string	Drop down item value.
ConfigStage.ConfigItems[n].ValueType	int	Config Item type STRING = 0, INT = 1, DOUBLE = 2, BOOLEAN = 3, PASSWORD = 4, LIST = 5
ConfigStage.ConfigItems[n].ConfigItemId	string	Unique name for the config item (MAX 40 chars)
ConfigStage.ConfigItems[n].Name	string	Value displayed on the user interface
ConfigStage.ConfigItems[n].Description	string	Tooltip text for the config item
ConfigStage.ConfigItems[n].GroupName	string	Config item group name. Config item will be rendered on the UI grouped by this value.

ConfigStage.ConfigItems[n].SortOrder	int	Sort order of the config item. Config items will be rendered on the screen in this order
ConfigStage.ConfigItems[n].SelectedValue	string	Current value of the config item/or default value. (MAX 255 chars)
ConfigStage.ConfigItems[n].RegexValidation	string/null	Regex valiation
ConfigStage.ConfigItems[n].RegexError	string/null	Error message that will be displayed when Regex validation fails
ConfigStage.ConfigItems[n].MustBeSpecified	true/false	Identifies whether the config item must have value specified.
ConfigStage.ConfigItems[n].ReadOnly	true/false	Identifies whether the field is read only
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message

Response Example

```
{
  "IsConfigActive":false,
  "ConfigStatus":"StageStatusName",
  "ConfigStage":{
    "WizardStepDescription":"Description of the stage.",
    "WizardStepTitle":"Title of the wizard Stage",
    "ConfigItems":[
      {
        "ListValues":[
          {
            "ValueType":0,
            "ConfigItemId":"ITEM1",
            "Name":"Config Item 1",
            "Description":"Description Item 1",
            "GroupName":"Group item",
            "SortOrder":1,
            "SelectedValue":"","
            "RegexValidation":null,
            "RegexError":null,
            "MustBeSpecified":true,
            "ReadOnly":false
          }
        ],
        {
          "ListValues":[
            {
              "Display":"List Value 1",
              "Value":"1"
            },
            {
              "Display":"List Value 2",
              "Value":"2"
            }
          ]
        }
      ]
    }
  }
}
```

```

        {
            "Display": "France",
            "Value": "List Value 3"
        },
        "ValueType": 5,
        "ConfigItemId": "LIST1",
        "Name": "List of values",
        "Description": "User can select values from a list",
        "GroupName": "Group List",
        "SortOrder": 2,
        "SelectedValue": "1",
        "RegexValidation": null,
        "RegexError": null,
        "MustBeSpecified": true,
        "ReadOnly": false
    },
    "IsError": false,
    "ErrorMessage": null
}

```

UpdateConfig

Update integration profile config stage with values entered by the user. The ConfigStatus must match the current config stage status.

Request

Field	Type	Description
AuthorizationToken	Guid	Authorization token uniquely identifying each integration profile
ConfigStatus	string	Current config stage. UserConfig call returns a current stage status. The same status is sent back to the integration webservice.
ConfigItems	Array of ConfigItem values	List of config item values.
ConfigItems[n].ConfigItemId	string	Unique config item id (MAX 40 chars)
ConfigItems[n].SelectedValue	string	Value entered by the user (MAX 255 chars)

Example request

```

{
    "AuthorizationToken": "c11c8f63-b362-4548-9af6-b76914afb0cb",

```

```
"ConfigStatus":"StageStatusName",
"ConfigItems":[
  {
    "ConfigItemId":"ITEM1",
    "SelectedValue":"Some value"
  },
  {
    "ConfigItemId":"LIST1",
    "SelectedValue":"1"
  }
]
```

Response

Field	Type	Description
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message

Response Example

```
{
  "IsError":false,
  "ErrorMessage":null
}
```

ConfigDelete

Method is called when the user deletes the integration in linnworks.net

Request

Field	Type	Description
AuthorizationToken	Guid	Authorization token uniquely identifying each integration profile

Example request

```
{
  "AuthorizationToken":"c11c8f63-b362-4548-9af6-b76914afb0cb"
}
```

Response

Field	Type	Description
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message

Response Example

```
{
  "IsError":false,
  "ErrorMessage":null
}
```

UserAvailableServices

Returns a list of available services for a given integration profile.

Request

Field	Type	Description
AuthorizationToken	Guid	Authorization token uniquely identifying each integration profile

Example request

```
{
  "AuthorizationToken":"c11c8f63-b362-4548-9af6-b76914afb0cb"
}
```

}

Response

Field	Type	Description
Services	Array of Service class	List of available services
Services[n].ServiceName	string	Service name as appears to the user
Services[n].ServiceCode	string	Service code
Services[n].ServiceTag	string	Service tag. Arbitrary value.
Services[n].ServiceGroup	string	Service grouping name.
Services[n].ServiceUniqueId	Guid	Unique Identifier for the service. Linnworks will use this ID in the generate consignment label request. It has to be unique for each service.
Services[n].ConfigItem	Array of ConfigItems	Each service can have configuration items associated with it.
Services[n].ConfigItem[x].ListValues	Array of ListValue	List of ListValues. The config item ValueType =5 (LIST) can have a list of available values the user can select from the drop down list
Services[n].ConfigItem[x].ListValues[m].Display	string	Value displayed in the drop down list
Services[n].ConfigItem[x].ListValues[m].Value	string	Drop down item value.
Services[n].ConfigItem[x].ValueType	int	Config Item type STRING = 0, INT = 1, DOUBLE = 2, BOOLEAN = 3, PASSWORD = 4, LIST = 5
Services[n].ConfigItem[x].ConfigItemId	string	Unique name for the config item (MAX 40 chars)
Services[n].ConfigItem[x].Name	string	Value displayed on the user interface
Services[n].ConfigItem[x].Description	string	Tooltip text for the config item
Services[n].ConfigItem[x].GroupName	string	Config item group name. Config item will be rendered on the UI grouped by this value.
Services[n].ConfigItem[x].SortOrder	int	Sort order of the config item. Config items will be rendered on the screen in this order
Services[n].ConfigItem[x].SelectedValue	string	Current value of the config item/or default value. (MAX 255 chars)

ue		
Services[n].ConfigItem[x].RegexValidation	string/null	Regex validation
Services[n].ConfigItem[x].RegexError	string/null	Error message that will be displayed when Regex validation fails
Services[n].ConfigItem[x].MustBeSpecified	true/false	Identifies whether the config item must have value specified.
Services[n].ConfigItem[x].ReadOnly	true/false	Identifies whether the field is read only
Services[n].ServiceProperty	Array of ServiceProperty	Each service has a list of properties.
Services[n].ServiceProperty[m].PropertyName	string	Property Name
Services[n].ServiceProperty[m].PropertyValue	string	Property Value
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message

Response Example

```
{
  "Services":
  [
    {
      "ServiceName": "Fast2Door Next Day",
      "ServiceCode": "CODE1",
      "ServiceTag": "SOMETAG",
      "ServiceGroup": "Fast2Door",
      "ServiceUniqueId": "6a476315-04db-4d25-a25c-e6917a1bcad9",
      "ConfigItems": [
        {
          "ListValues": [
            {
              "Display": "Up to 500",
              "Value": "500"
            },
            {
              "Display": "Up to 1000",
              "Value": "1000"
            }
          ],
          "ValueType": 5,
          "ConfigItemId": "InsuranceConver",
          "Name": "Insurance Cover",
          "Description": "Additional Insurance conver",
          "GroupName": "Insurance",
          "SortOrder": 1,
          "SelectedValue": "500",
          "RegexValidation": null,
          "RegexError": null,
          "MustBeSpecified": false,

```

```

        "ReadOnly":false
      }
    ],
    "ServiceProperty":[
      {
        "PropertyName":"LABELTAG",
        "PropertyValue":"Next Day"
      }
    ]
  }
},
"IsError":false,
"ErrorMessage":null
}

```

ExtendedPropertyMapping

Get the list of extended property mapping values.

Request

Field	Type	Description
AuthorizationToken	Guid	Authorization token uniquely identifying each integration profile

Example request

```

{
  "AuthorizationToken":"c11c8f63-b362-4548-9af6-b76914afb0cb",
}

```

Response

Field	Type	Description
Items	Array of Extended Property Items	List of extended properties
Items[n].PropertyTitle	string	Property title
Items[n].PropertyName	string	Property Name (MAX 255 chars)
Items[n].PropertyDescription	string	Property Description

Items[n].PropertyType	string	Property Type ORDER ITEM
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message

Response Example

```
{
  "Items":
  [
    {
      "PropertyTitle": "Safe Place note",
      "PropertyName": "SafePlace1",
      "PropertyDescription": "Safe place note for delivery",
      "PropertyType": "ORDER"
    },
    {
      "PropertyTitle": "Extended Cover flag",
      "PropertyName": "ExtendedCover",
      "PropertyDescription": "Some description",
      "PropertyType": "ITEM"
    }
  ],
  "IsError": false,
  "ErrorMessage": null
}
```

GenerateLabel

Generate Label PNG and associated shipping label documentation. This endpoint method accept a consignment request which contains the shipping address, list of packages and shipping service configuration options. The method will generate a label for each package and return all packages with the matching sequence number.

Request

Field	Type	Description
AuthorizationToken	Guid	Authorization token uniquely identifying each integration profile

Name	string	Customer name
CompanyName	string	Company name
AddressLine1	string	Address line 1
AddressLine2	string	Address line 2
AddressLine3	string	Address line 3
Town	string	Adderss Town/City
Region	string	State/County/Region
CountryCode	string	Country code ISO 3166-2
Postalcode	string	Postal code/Zip code
DeliveryNote	string	Delivery note
Email	string	Email address of the customer
Phone	string	Phone number of the customer
OrderId	int	Linnworks order id
OrderReference	string	Linnworks order reference number
OrderCurrency	string	Currency code (3 chars)
OrderValue	decimal	Order value
PostageCharges	decimal	Shipping cost
ServiceId	Guid	Unique Service Id, as defined in UserAvailableServices response Services[n].ServiceUniqueld
Packages	Array of Package class	List of packages
Packages[n].SequenceNumber	int	Sequence number of a package. Response must contain the same sequence number for each package
Packages[n].PackageWeight	decimal	Package weight (grams)
Packages[n].PackageWidth	decimal	Package width (cm)
Packages[n].PackageHeight	decimal	Package height (cm)
Packages[n].PackageDepth	decimal	Package depth (cm)
Packages[n].PackageFormat	decimal	Packaging type BOX PARCEL PACKET LETTER

Packages[n].Items	Array of Items/products	List of order items in the package
Packages[n].Items[m].ItemName	string	Name of the product
Packages[n].Items[m].ProductCode	string	SKU
Packages[n].Items[m].Quantity	int	Item quantity
Packages[n].Items[m].UnitValue	decimal	Product unit cost (OrderCurrency)
Packages[n].Items[m].UnitWeight	decimal	Product unit weight (gram)
Packages[n].Items[m].Height	decimal	Product height (cm)
Packages[n].Items[m].Width	decimal	Product width (cm)
Packages[n].Items[m].length	decimal	Product length (cm)
Packages[n].Items[m].ExtendedProperties	Array of Extended Property classes	Extended property as mapped to linnworks item extended property
Packages[n].Items[m].ExtendedProperties[x].Name	string	Extended Property name as defined in ExtendedPropertyMapping Items[n].PropertyName
Packages[n].Items[m].ExtendedProperties[x].Value	string	Linnworks item extended property value
OrderExtendedProperties	Array of extended property	List of extended property items mapped to order extended properties
OrderExtendedProperties[m].Name	string	Extended Property name as defined in ExtendedPropertyMapping Items[n].PropertyName
OrderExtendedProperties[m].Value	string	Linnworks item extended property value
ServiceConfigItems	Array of Service configuration option values	List of config items values as defined by the user in the service configuration
ServiceConfigItems[n].ConfigItemId	string	Config item Id as defined in UserAvailableServices Services[n].ConfigItem[x].ConfigItemId
ServiceConfigItems[n].SelectedValue	string	Value specified by the user in the config

Example request

```
{
  "AuthorizationToken": "c11c8f63-b362-4548-9af6-b76914afb0cb",
  "Name": "Customer name",
  "CompanyName": "Company name",
  "AddressLine1": "Address Line 1",
  "AddressLine2": "Address Line 2",
```

```
"AddressLine3": "Address Line 3",
"Town": "Address City",
"Region": "County",
"CountryCode": "GB",
"Postalcode": "PO198DJ",
"DeliveryNote": "Some note",
"Email": "test@linnsystems.com",
"Phone": "+442036517045",
"OrderId": 100001,
"OrderReference": "ORDERREFERENCE",
"OrderCurrency": "GBP",
"OrderValue": 1.0,
"PostageCharges": 1.0,
"ServiceId": "6a476315-04db-4d25-a25c-e6917a1bcad9",
"Packages":
[
  {
    "SequenceNumber": 0,
    "Items": [
      {
        "ExtendedProperties":
        [
          {
            "Name": "ExtendedCover",
            "Value": "True"
          }
        ],
        "ItemName": "Product 1",
        "ProductCode": "SKU001",
        "Quantity": 1,
        "UnitValue": 1.0,
        "UnitWeight": 1.0,
        "Height": 1.0,
        "Width": 1.0,
        "Length": 1.0
      }
    ],
    "PackageWeight": 1.0,
    "PackageWidth": 1.0,
    "PackageHeight": 1.0,
    "PackageDepth": 1.0,
    "PackageFormat": "BOX"
  }
],
"OrderExtendedProperties":
[
  {
    "Name": "SafePlace",
    "Value": "Some message"
  }
],
"ServiceConfigItems":
[
  {
    "ConfigItemId": "InsuranceConver",
    "SelectedValue": "500"
  }
]
}
```

Response

Field	Type	Description
-------	------	-------------

LeadTrackingNumber	string	Lead tracking number for the whole consignment.
Cost	decimal	Cost of all labels
Package	Array of package response	List of package response classes
Package[n].SequenceNumber	int	Package sequence number. Must match to what is in the request Packages[n].SequenceNumber
Package[n].TrackingNumber	string	Package tracking number
Package[n].PNGLabelDataBase64	string	Base64 PNG label
Package[n].PDFBytesDocumentationBase64	Array of strings	Each element is Base64 PDF document. If no document must be an empty array. Null will result in error
LabelWidth	decimal	Label width in inches
LabelHeight	decimal	Label height in inches
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message

Response Example

```
{
  "LeadTrackingNumber": "GB1234567891",
  "Cost": 0.0,
  "Currency": "GBP",
  "Package": [
    {
      "SequenceNumber": 0,
      "TrackingNumber": "GB1234567891",
      "PNGLabelDataBase64": "DJHDD234872376478DDD....",
      "PDFBytesDocumentationBase64": [
        "DJHDD234872376478DDD....",
        "DJHDD234872376478DDD...."
      ],
      "LabelWidth": 4.0,
      "LabelHeight": 6.0
    }
  ],
  "IsError": false,
  "ErrorMessage": null
}
```

CancelLabel

Cancels all labels for the consignment by OrderId. If your integration does not support label cancellation implement an error message from this method.

Request

Field	Type	Description
AuthorizationToken	Guid	Authorization token uniquely identifying each integration profile
OrderId	int	Linnworks order id

Example request

```
{
  "AuthorizationToken" : "c11c8f63-b362-4548-9af6-b76914afb0cb",
  "OrderId":100001
}
```

Response

Field	Type	Description
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message

Response Example

```
{
  "IsError":false,
  "ErrorMessage":null
}
```

CreateManifest

The method is called by linnworks.net when a user files a manifest. The request will contain all order Id to be manifested (filed as the end of day)

Request

Field	Type	Description
AuthorizationToken	Guid	Authorization token uniquely identifying each integration profile
OrderId	Array of strings	List of OrderIds. Although OrderId is int, note that this array of strings to avoid casting issues.

Example request

```
{
  "AuthorizationToken" : "c11c8f63-b362-4548-9af6-b76914afb0cb",
  "OrderId":
  [
    "100001",
    "100002"
  ]
}
```

Response

Field	Type	Description
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message
ManifestReference	string	Manifest reference

Response Example

```
{
  "IsError":false,
  "ErrorMessage":null,
  "ManifestReference" : "RF2478937333-1"
}
```

PrintManifest

If your carrier integration requires printed version of end-of-day manifest, you can generate PDF and the user will be able to print it from linnworks.net.

Request

Field	Type	Description
AuthorizationToken	Guid	Authorization token uniquely identifying each integration profile
ManifestReference	string	Manifest reference

Example request

```
{
  "AuthorizationToken" : "c11c8f63-b362-4548-9af6-b76914afb0cb",
  "ManifestReference" : "RF2478937333-1"
}
```

Response

Field	Type	Description
IsError	true/false	Indicates if there is an error
ErrorMessage	string/null	Error message
PDFbase64	string	Base64 PDF file

Response Example

```
{
  "IsError":false,
  "ErrorMessage":null,
  "PDFbase64" : "DJHDD234872376478DDD...."
}
```

