

SAP Commerce

**V3 Document**

**Contents**

[1. Defaults 1](#_Toc77948101)

[2. Connector Base Code Installation Steps 1](#_Toc77948102)

[2.1. Installation 2](#_Toc77948103)

[2.2. Steps to modify the order 2](#_Toc77948104)

[2.3. Steps to modify return flow(yacceleratorordermanagement) 3](#_Toc77948105)

[2.4. New Installation 4](#_Toc77948106)

[3. Customer Loyalty 5](#_Toc77948107)

[3.1. Loyalty Dashboard 5](#_Toc77948108)

[3.2. Custom Action 6](#_Toc77948109)

[3.3. Loyalty Enrollment Program 7](#_Toc77948110)

[3.4. Product Point 9](#_Toc77948111)

[3.5. Cart Point 11](#_Toc77948112)

[3.6. Point Redemption 12](#_Toc77948113)

[3.6.1. Coupon Redemption 12](#_Toc77948114)

[3.6.2. Generate Rewards & Upload the Coupon Codes in Annex Cloud 20](#_Toc77948115)

[4. Checkout Flow 22](#_Toc77948116)

[5. Order Flow 22](#_Toc77948117)

1. Defaults

**Client Name (Site Name):** Contact to Annex Cloud

**Site ID:** Contact to Annex Cloud

**Access Token (Site Token):** Contact to Annex Cloud

1. Connector Base Code Installation Steps

* Get Annex Cloud code base. It will consist of five extensions.
* annexcloudaddon
* annexcloudbackoffice
* annexcloudv3fullfillmentprocess
* annexcloudv3facades
* annexcloudv3services
* Copy all the five extensions inside {base\_directory}/hybris/bin/custom.
* Include all five extensions in the **localextensions.xml** file. Also, add the “addonsupport” OOTB extension which is required for Addon installation.
* Install the following addon:
* annexcloudaddon
* Now, go to {base\_directory}/hybris/bin/platform directory and execute the following addon install commands.

|  |
| --- |
| ant addoninstall -Daddonnames="annexcloudaddon" -DaddonStorefront.yacceleratorstorefront="<Name of the storefront extension>" |

* Define dependency by updating the **extensioninfo.xml** of annexcloudv3facades extension.

|  |
| --- |
| <requires-extension name="annexcloudv3services"/>  <requires-extension name="Here you need to add name of your core extension"/>  <requires-extension name="acceleratorfacades"/> |

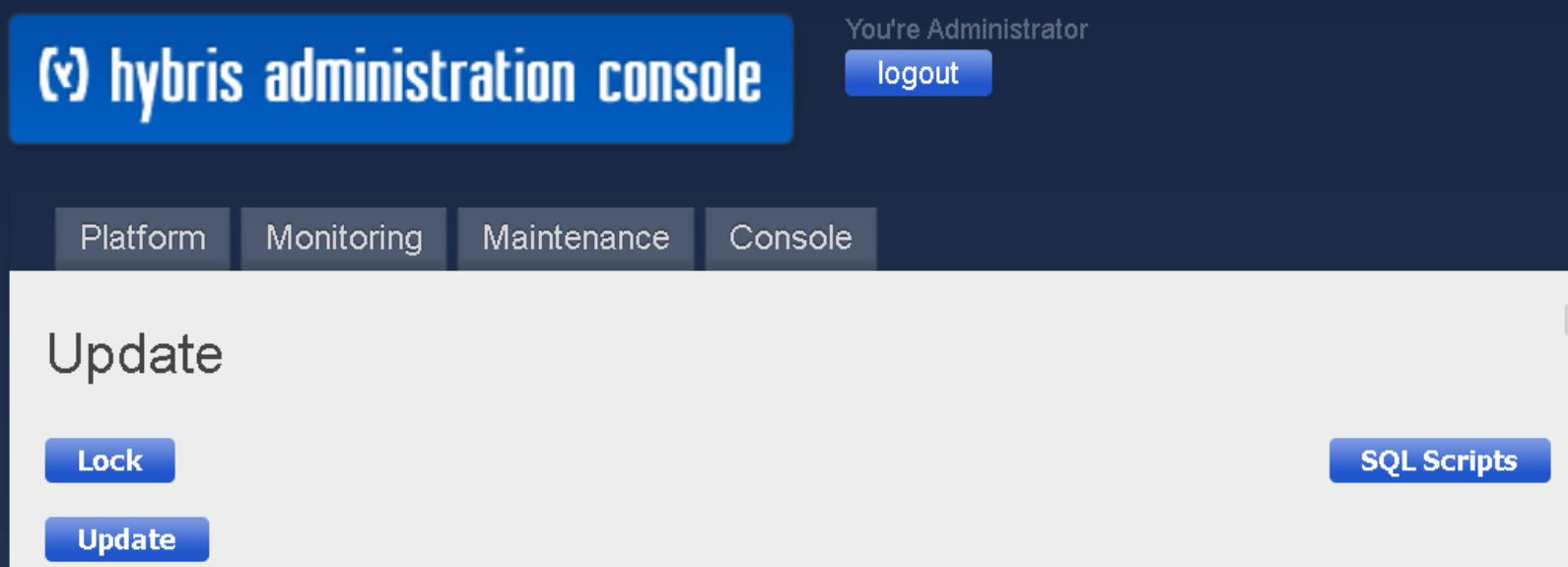
**For example,** If the name of your accelerator core is annexcloudcore, then put the same name above.

* After the addon installation, your storefront will get updated.
  1. Installation

To update system from HAC. Select only create essential data as shown below.

**Steps:**

1. Go to **HMAC/console.**
2. On **Platform** tab, click **Update**. It will display the following window.



* Execute Electronic catalog sync job so that the components get copied from staged to Online.
* After the system update from HAC, your database will get updated.
  1. Steps to modify the order

If you are using yacceleratorordermanagement then refer below steps:

* Add the following bean definitions to **order-process.xml**

|  |
| --- |
| <bean id="sendOrderDetailToAnnexCloud" class= "com.annexcloud.actions.order.SendOrderDetailToAnnexCloudAction"      parent="abstractAction">    <property name="annexCloudLoyaltyService" ref="acV3LoyaltyService"/>  </bean> |

Changes need to be done in projectdata-dynamic-business-process-order.impex

|  |
| --- |
| <action id='takePayment' bean='takePaymentAction'>  <transition name='OK' to='sendOrderDetailToAnnexCloud'/>  <transition name='NOK' to='sendPaymentFailedNotification'/>  </action>  <action id='sendOrderDetailToAnnexCloud' bean='sendOrderDetailToAnnexCloud'>  <transition name='OK' to='completeOrder'/>  <transition name='NOK' to='error'/>  </action>  <action id='completeOrder' bean='completeOrderAction'>  <transition name='OK' to='success'/>  </action> |

* Add the following dependency into **extensioninfo.xml** of **yacceleratorordermanagement** extension.

|  |
| --- |
| <requires-extension name="annexcloudv3services"/> |

* Do ant all and run the **projectdata-dynamic-business-process-order.impex in HAC**
  1. Steps to modify return flow(yacceleratorordermanagement)

If you are using yacceleratorfulfilmentprocess then refer below steps

* Add the following bean definitions to **return-process.xml**

|  |
| --- |
| <bean id="annexCloudReturnOrderAction" class="com.annexcloud.returns.order.ACReturnOrderAction"          parent="abstractAction">        <property name="annexCloudLoyaltyService" ref="acV3LoyaltyService"/>   </bean> |

If you are using yacceleratorordermanagement then refer below steps.

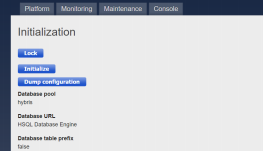
* Add the following actions to **projectdata-dynamic-business-process-return.impex** file.

|  |
| --- |
| <action id='inventoryUpdateAction' bean='inventoryUpdateAction'>  <transition name='OK' to='annexCloudReturnOrderAction'/>  </action>  <action id='annexCloudReturnOrderAction' bean='annexCloudReturnOrderAction'>  <transition name='OK' to='completeReturnAction'/>  <transition name='NOK' to='failed'/>  </action>  <action id='completeReturnAction' bean='completeReturnAction'>  <transition name='OK' to='success'/>  </action> |

* Do ant all and run the **projectdata-dynamic-business-process-order.impex.**
* After that, at the same directory, perform command **ant clean all.**
* Then, start hybris server.
  1. New Installation

1. After the system starts, go to hac (<https://ClientURL/hac>)
2. If you haven’t initialized the system, then perform step 3 else jump to step 5. 3.

Go to **HAC 🡺 Platform 🡺 Initialization** and keep all the options selected and click on the Initialize button to do system initialization as shown below.



1. Execute full solr indexing. To execute, go to **System 🡺 Search and Navigate 🡺 Facet search configurations 🡺 Select solr profile 🡺 Click on index 🡺 Select full indexing option 🡺Click start.**

* **Note:** In solr profile, annexcloud is having its own value provider for average rating.

1. To update system from HAC. Select only create essential data as shown below.



1. Execute Electronic catalog sync job so that the components get copied from staged to Online.
2. Customer Loyalty
   1. Loyalty Dashboard

* Add following in “local.properties” file:**<site\_code>.site.api.token=<Site token>**

i.e., annexsapv3.site.api.token=QWEQWE121QWE

* Import the **annexclod-cms-content.impex** file to SAP commerce.

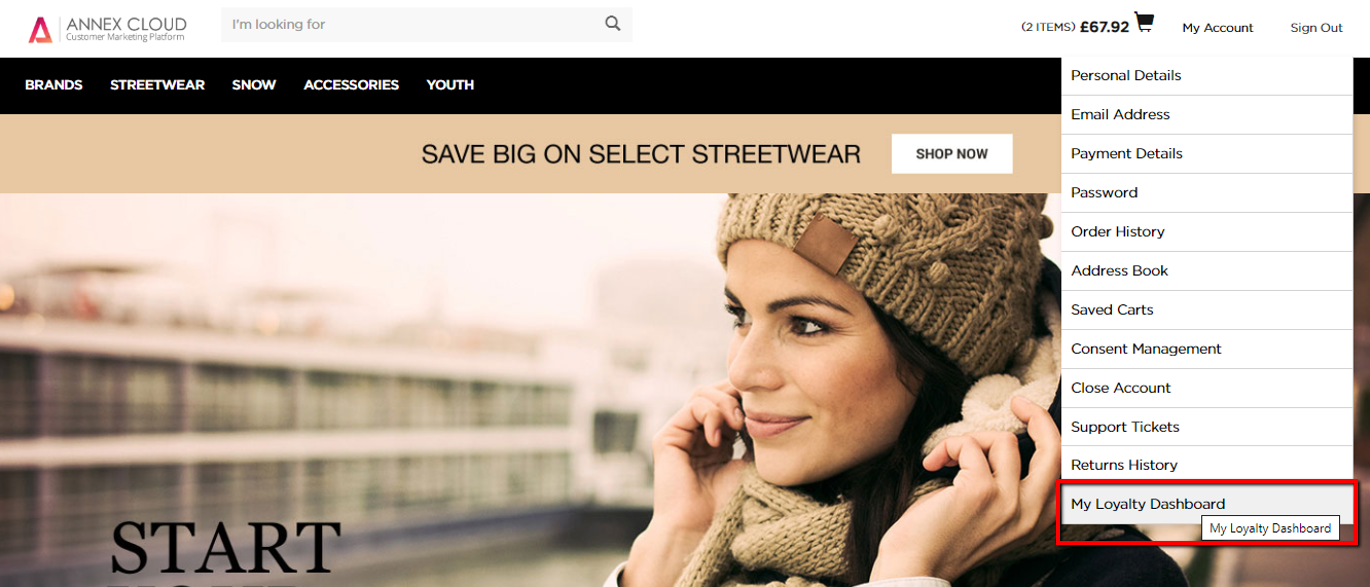
|  |
| --- |
| $annexSiteId=**ENTER\_SITE\_ID**  $annexSiteCode=**ENTER\_SITE\_CODE(SITE\_NAME)**  $annexorderProcessCode=order-process  $annexCheckoutFlowGroup=annexCloudCheckoutGroup |

**Note:** In the above code, replace the **site id, site code, access token** provided by annex cloud.

Attributes are explained below:

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| Code (Site Name) | It should be unique. |
| Site ID | Shared by Annex team to the client. |
| Base Store | Site ID associated with the base store. |
| Loyalty Program Type | Site ID can be implicit or explicit. If the site ID is explicit, the business user needs to select explicit, otherwise, by default, it is selected implicit |
| Point Redemption Type | coupon |

* The loyalty dashboard link will start displaying on my account bar once the user gets registered on the Hybris e-commerce site.
* After the loyalty dashboard setup, the user will get to see the loyalty dashboard link as shown below.



* 1. Custom Action

**To award/deduct custom points:**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Parameter Description** | **Parameter Type** |
| id (Mandatory) | Enter the user’s unique id. | string |
| actionId (Mandatory) | Enter the action id. (It is mandatory if the action id is set at the site admin) | string |
| activity (Mandatory) | Enter the activity as CREDIT or DEBIT | string |
| orderId (optional) | If the activity is DEBIT, pass the order id | string |
| credit/debit | Enter the number of credited/debited points | string |
| reason | Enter the reason for adding or removing the points (If the actionId is not passed then the reason is mandatory) | string |

* Call the following method from the class

*com.annexcloud.service.impl.DefaultAnnexCloudLoyaltyV3Service*.

**Method:**

|  |
| --- |
| public ACResponse awardDeductCustomPointsAPI(final ACCreditDebitPointsJwtRequest requestParameter,  final AnnexCloudModel annexSiteDetails) |

* 1. Loyalty Enrollment Program
* Import the following **annexcloud-cms-content.impex** to SAP commerce.

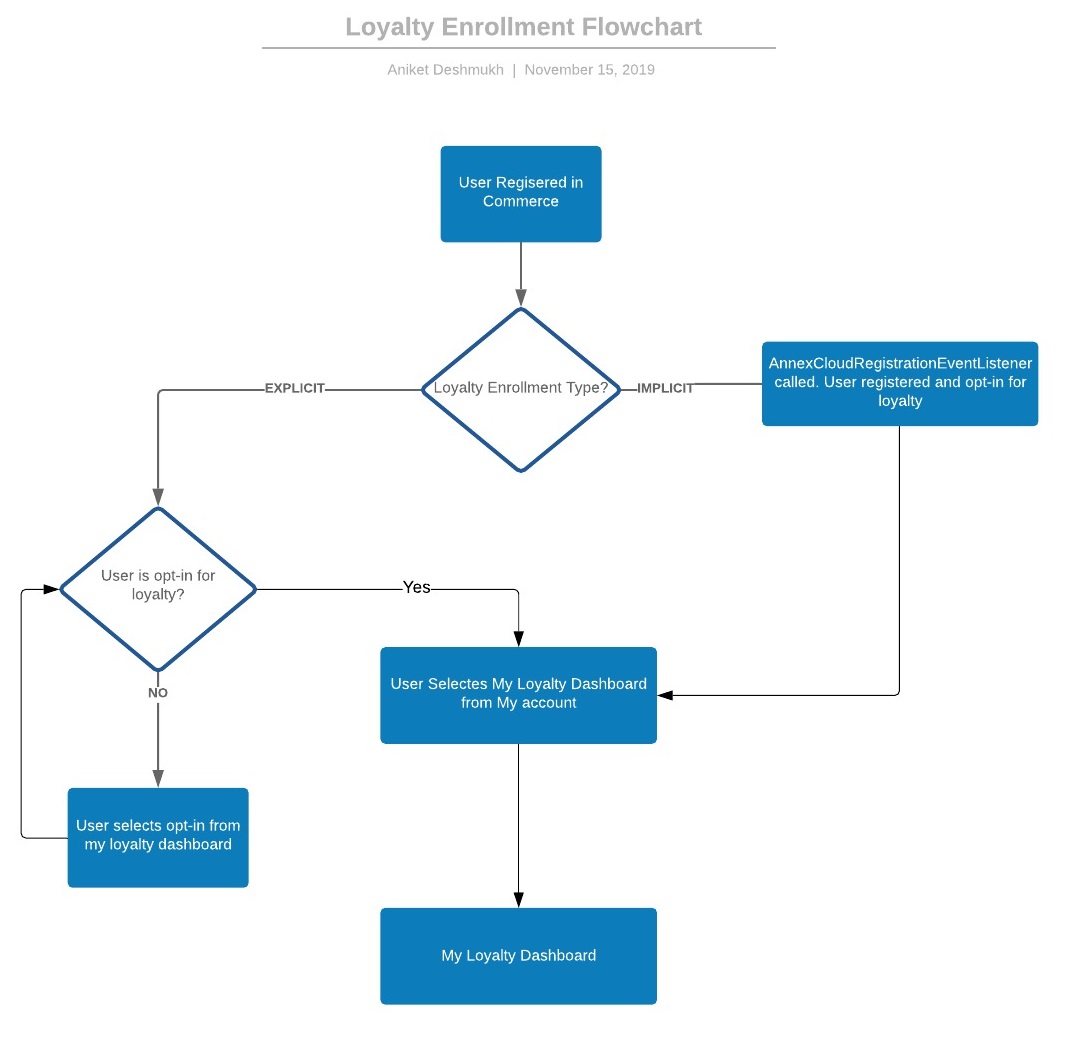
**Qualified Path:** D:\SAPHybris\hybris\bin\custom\annexcloudaddon\resources\annexcloudaddon\import\common\ annexclod-cms-content.impex

**Controller Class:** com.annexcloud.controllers.pages.**ACLoyaltyPageController**

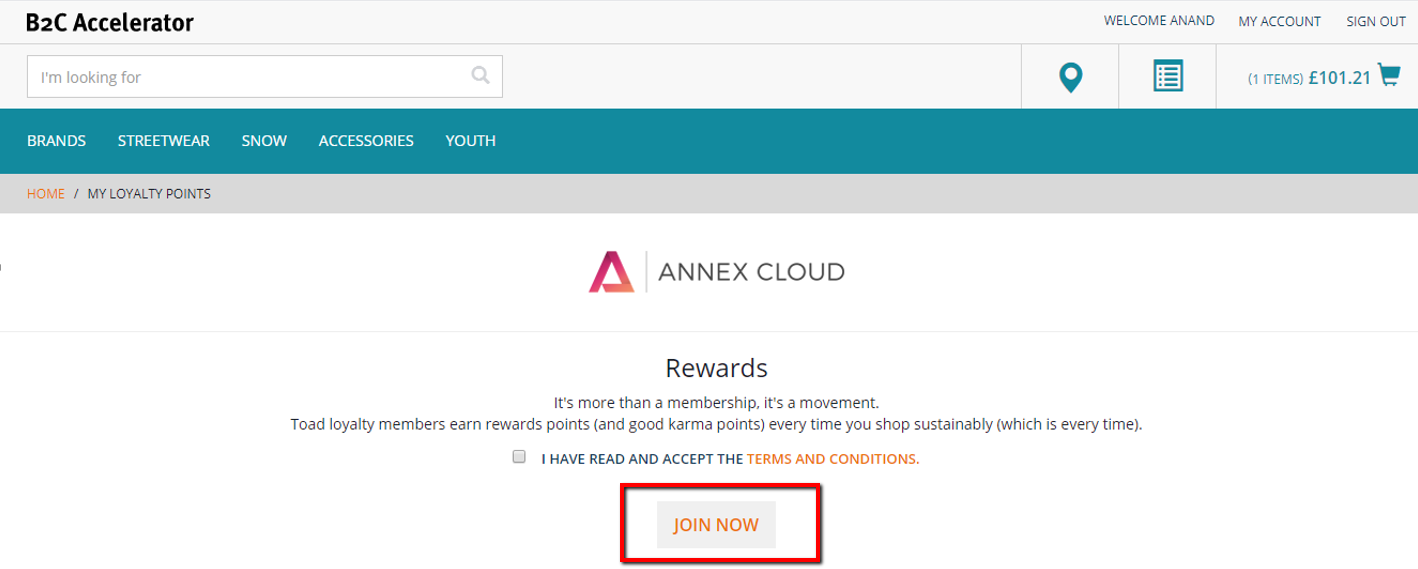
**URL:** /my-account/loyalty-point

**Jsp Page:** annexcloudaddon/responsive/pages/account/annexLoyaltyPointDashboard.jsp

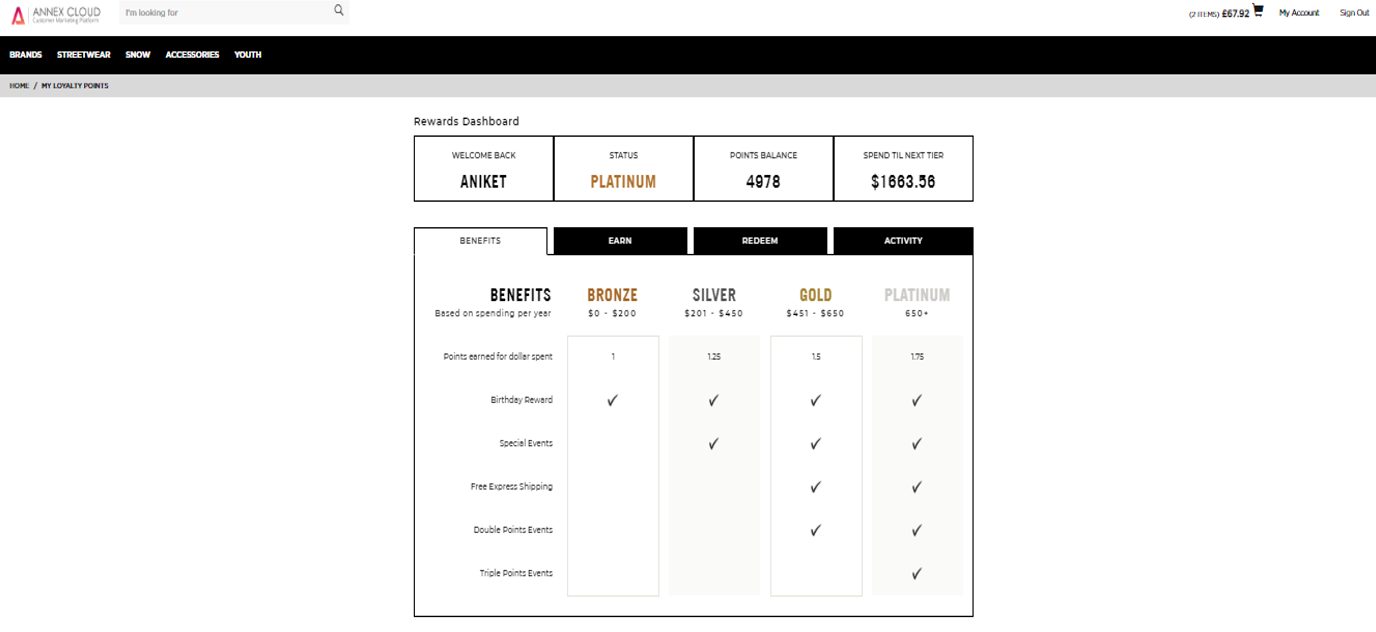
**Impex File:** annexcloud-cms-content.impex



* After the loyalty enrollment program is setup, if the program is implicit then the user will get enrolled in the loyalty program directly after creating an account.
* If the program is explicit, the user will have to join the loyalty program separately after creating an account as shown in the following image.



* After joining the loyalty program, the user will get to see the loyalty dashboard as shown in the following image.



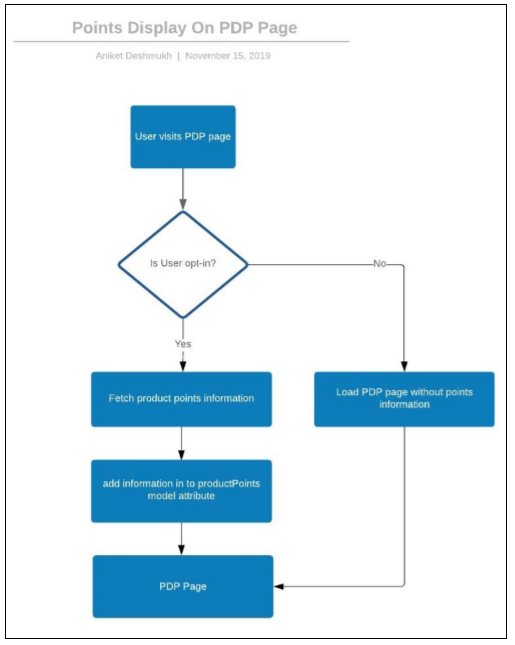
* 1. Product Point

**Precondition: Product point should be setup on Annex Cloud:**

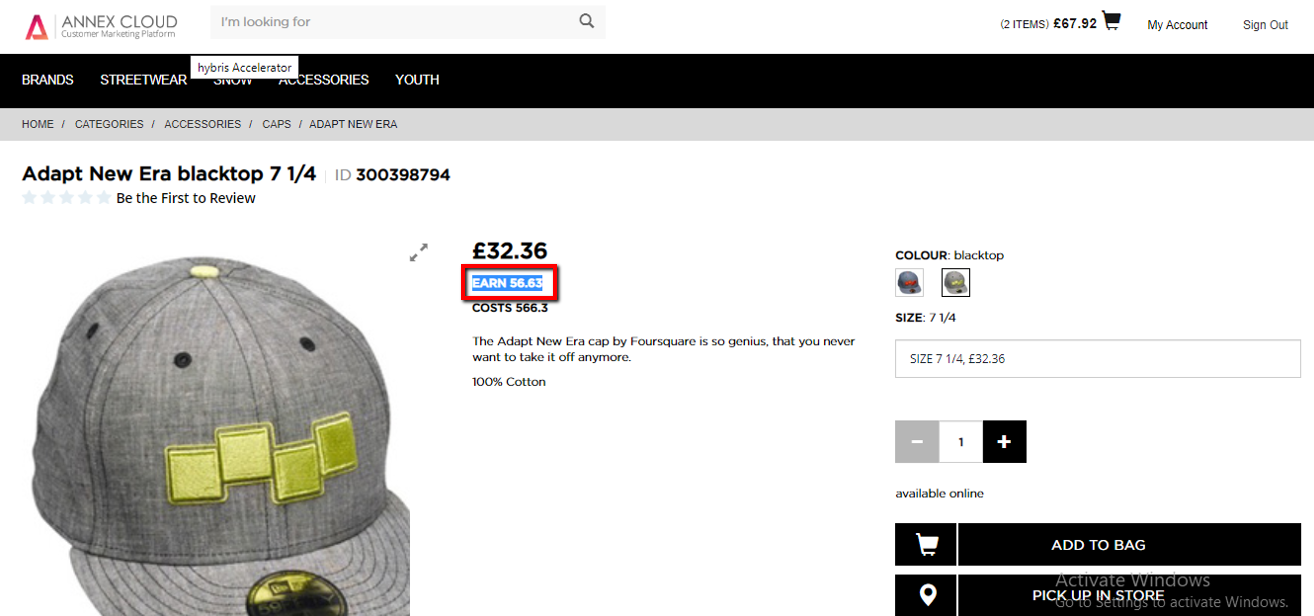
* In the Annex Cloud V3 setup, we have added **annexProductLayout1Page.jsp** page for the product details page template.
* **Qualified Path:** /annexcloudaddon/acceleratoraddon/web/webroot/WEB-INF/views/responsive/pages/product/annexProductLayout1Page.jsp

|  |  |  |
| --- | --- | --- |
| **Page** | **Out of the box file (earlier)** | **Annex files (updated)** |
| Product Detail Page | productLayout1Page.jsp | annexProductLayout1Page.jsp |
| Controller | ProductPageController.java | ACProductPageController.java |

* **JSP File:** productPricePanel.tag
* **Qualified Path:** annexcloudaddon\acceleratoraddon\web\webroot\WEB-INF\tags\responsive\product
* **Controller Class:** ACProductPageController
* **Model Attributes:** productPoints



* The user will get to see the earn points (the number of points he/she will earn after purchasing that particular product) on PDP as shown in the following image.

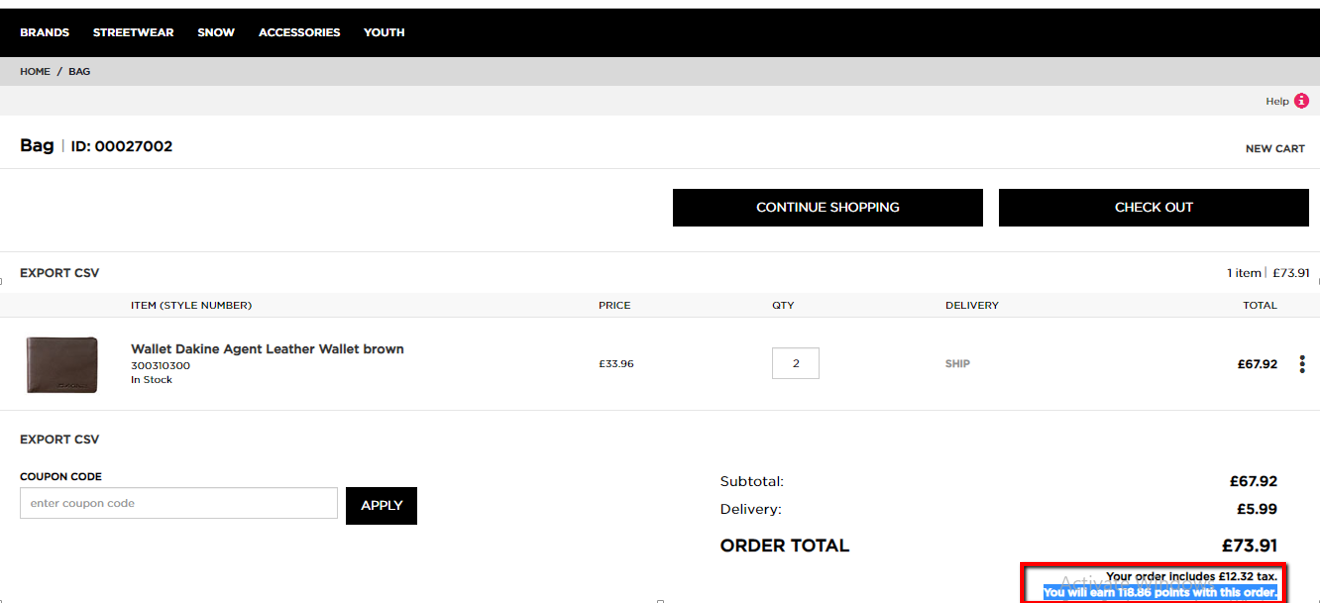


* 1. Cart Point
* In Annex Cloud V3 setup, we have added **annexCloudcartTotalsDisplay.jsp** page to display loyalty points on the cart page.

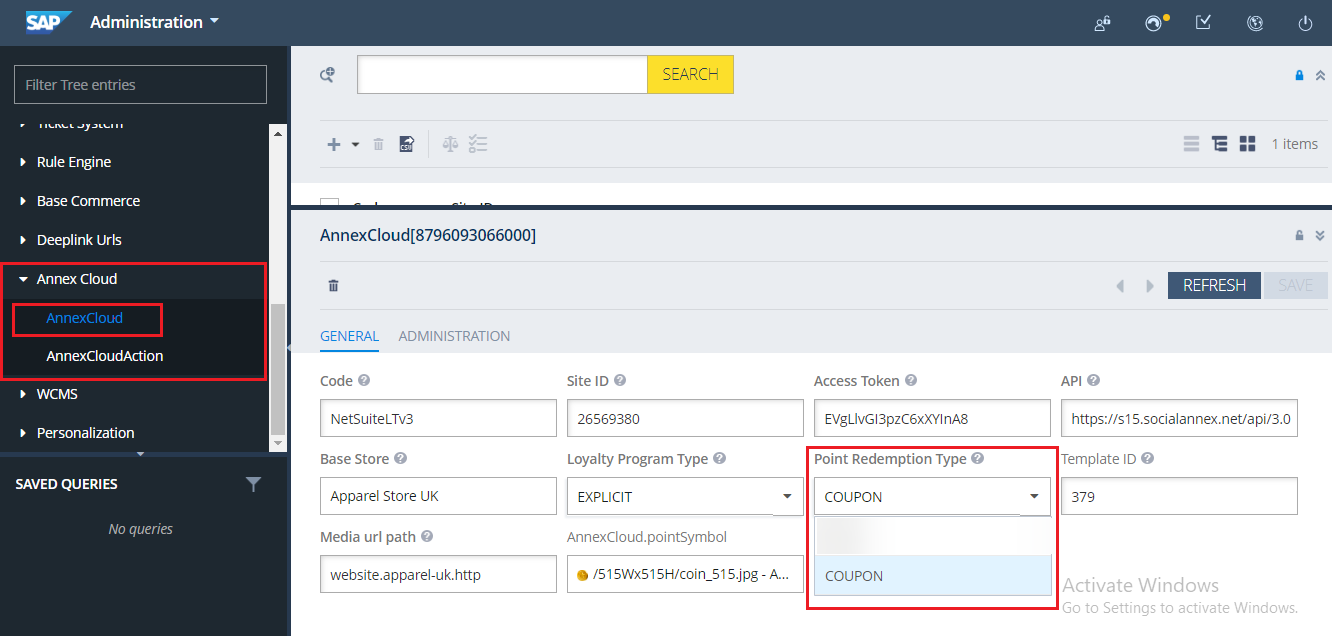
**Qualified Path:** /annexcloudaddon/acceleratoraddon/web/webroot/WEB-INF/views/responsive/pages/cart/annexCloudcartTotalsDisplay.jsp

|  |  |  |
| --- | --- | --- |
| **Page** | **Out of the box file (earlier)** | **Annex files (updated)** |
| Cart Page | cartTotalsDisplay.jsp | annexCloudcartTotalsDisplay.jsp |

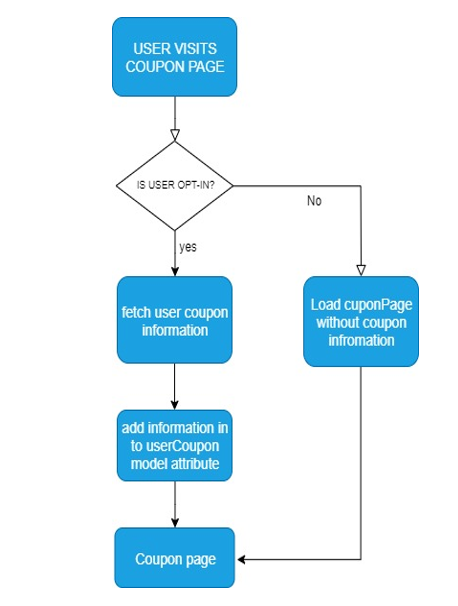
* Similarly, it can be applied on the cart page also as shown in the following image.



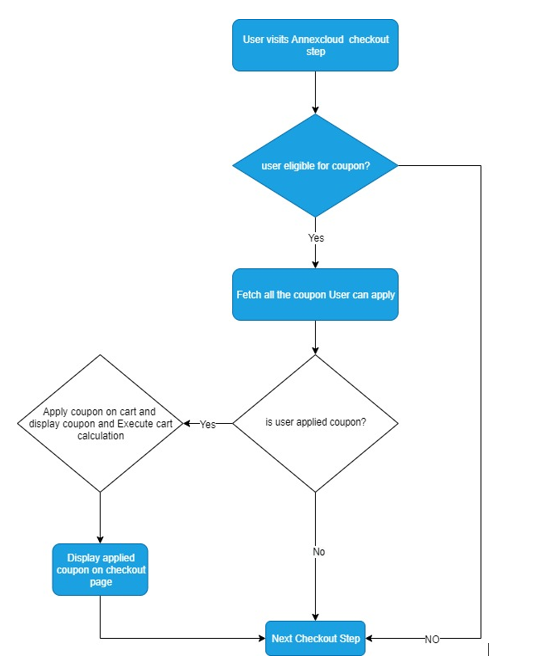
* 1. Point Redemption
* The user can redeem the loyalty points as coupon. Refer the following screenshot for this.



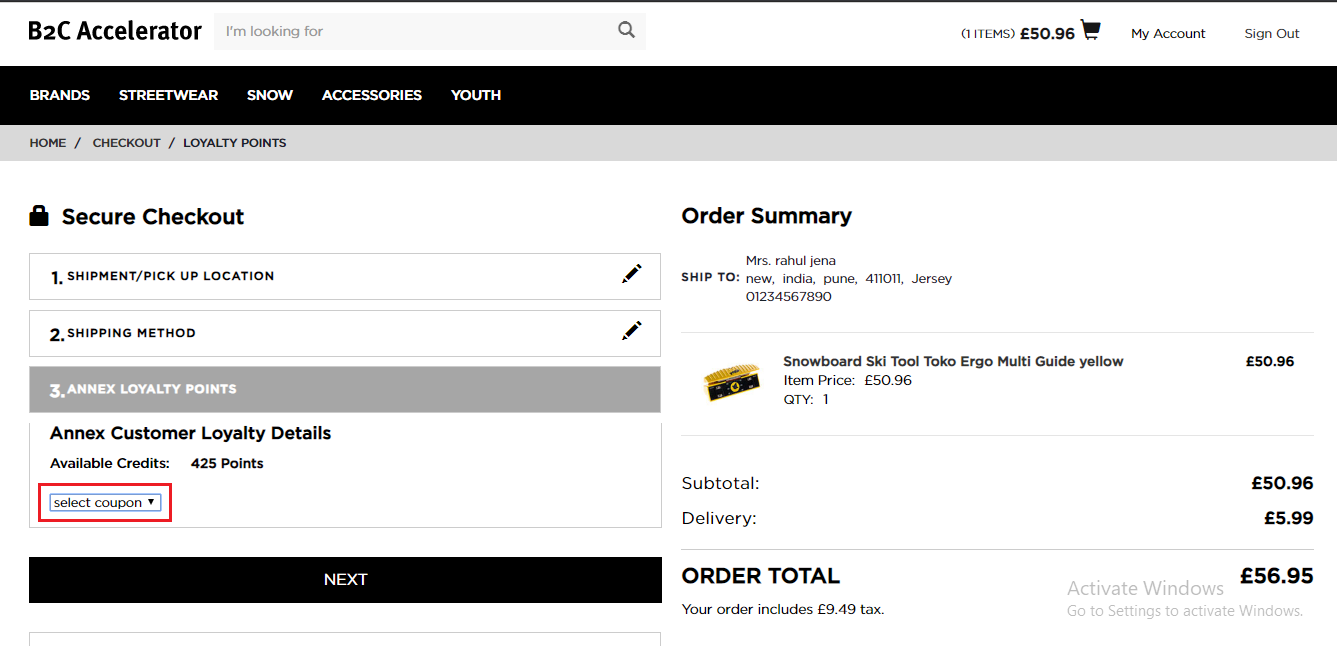
* + 1. Coupon Redemption
* To redeem the available points, we have added one checkout step.
* **JSP File:** ac acPointRedemptionPage.jsp
* **Qualified Path:** /annexcloudaddon/acceleratoraddon/web/webroot/WEB-INF/views/responsive/pages/checkout/multi/acPointRedemptionPage.jsp
* **Controller Class:** ACPointRedemptionCheckoutStepController
* **Checkout Group:** annexCloudCheckoutGroup
* **Checkout group file at line 155:** annexcloudaddon-web-spring.xml

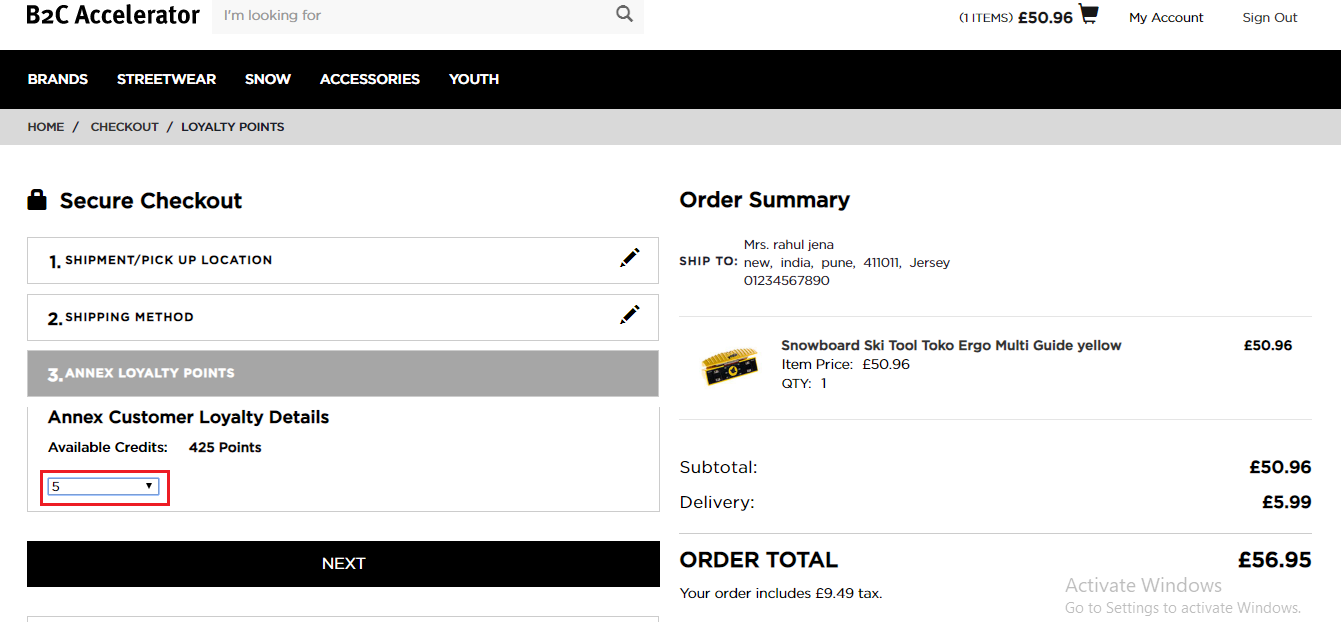


* Redeem points using coupon on checkout page.

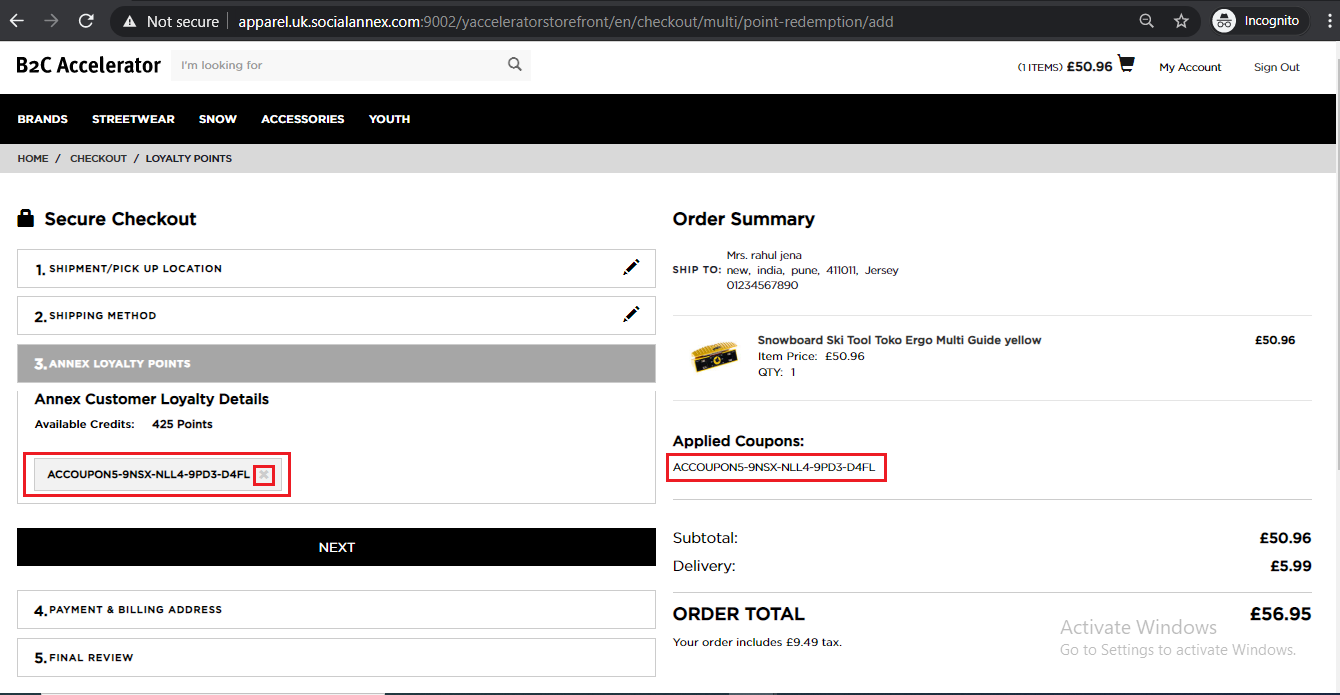


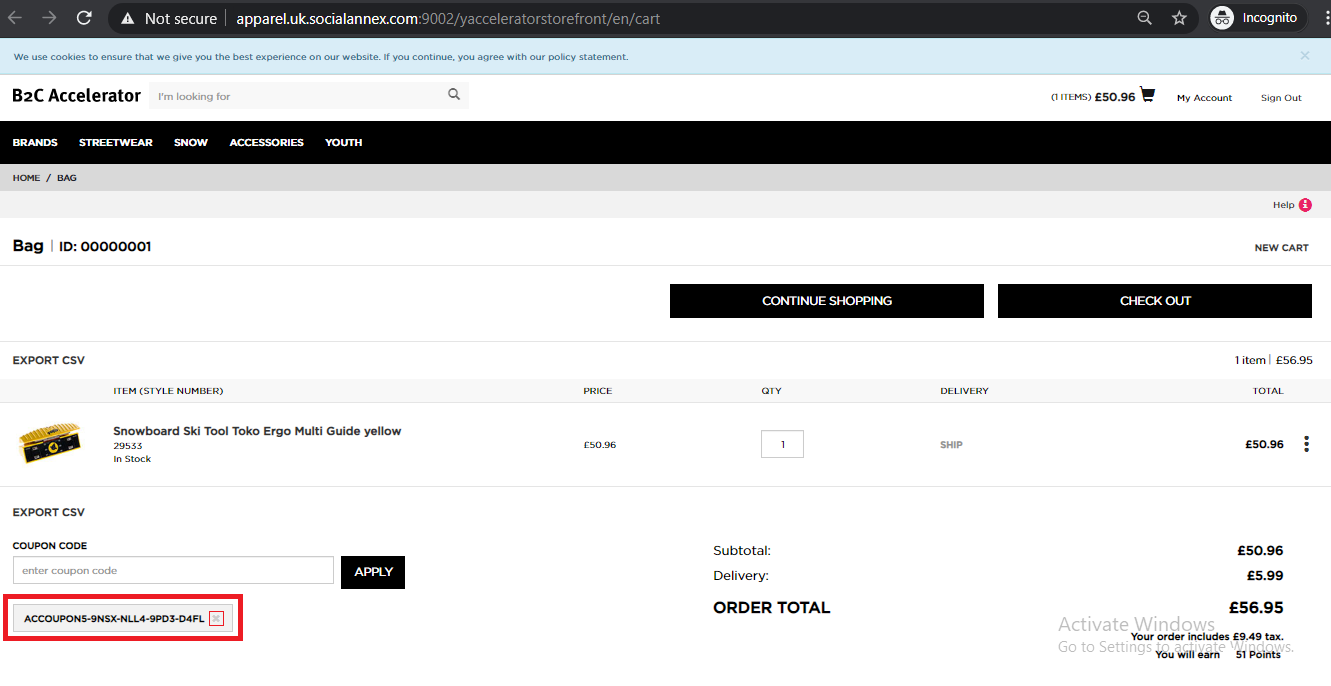
* After the coupon redemption setup, the user can redeem the coupon using the loyalty points on the checkout page using a drop-down menu as shown in the following image.





* After applying the coupon, applied coupon will be visible on the checkout as well as cart page. To remove the coupon, click on the cross button.



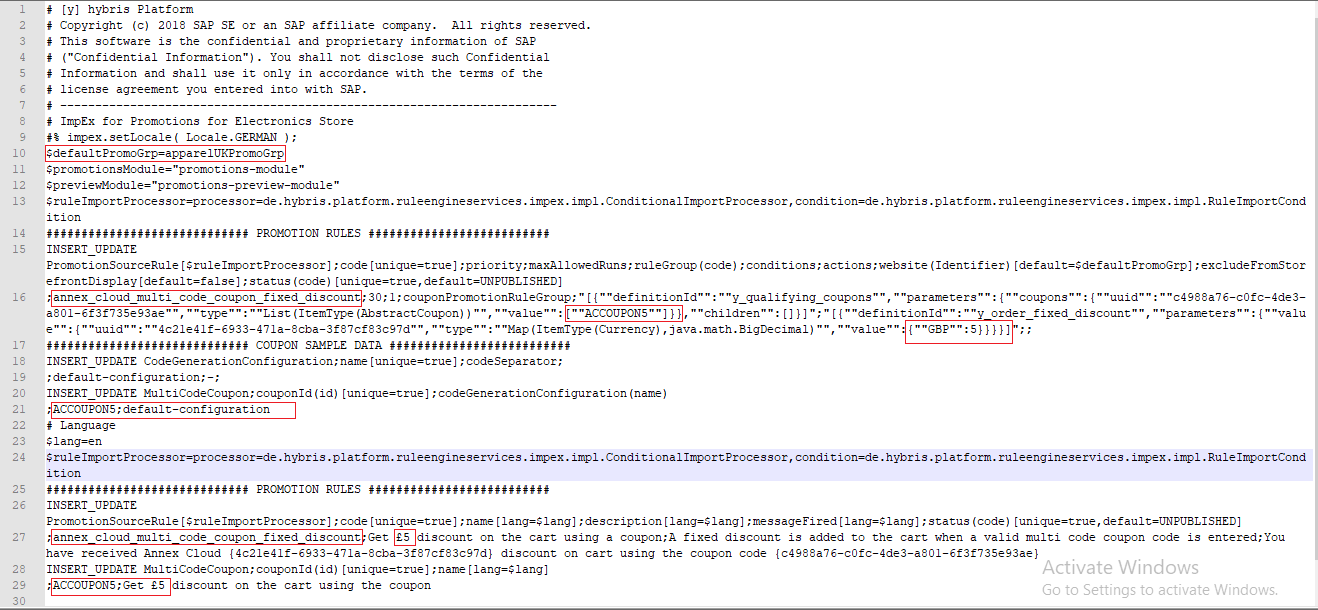


* Get the impex and then visit the following url:

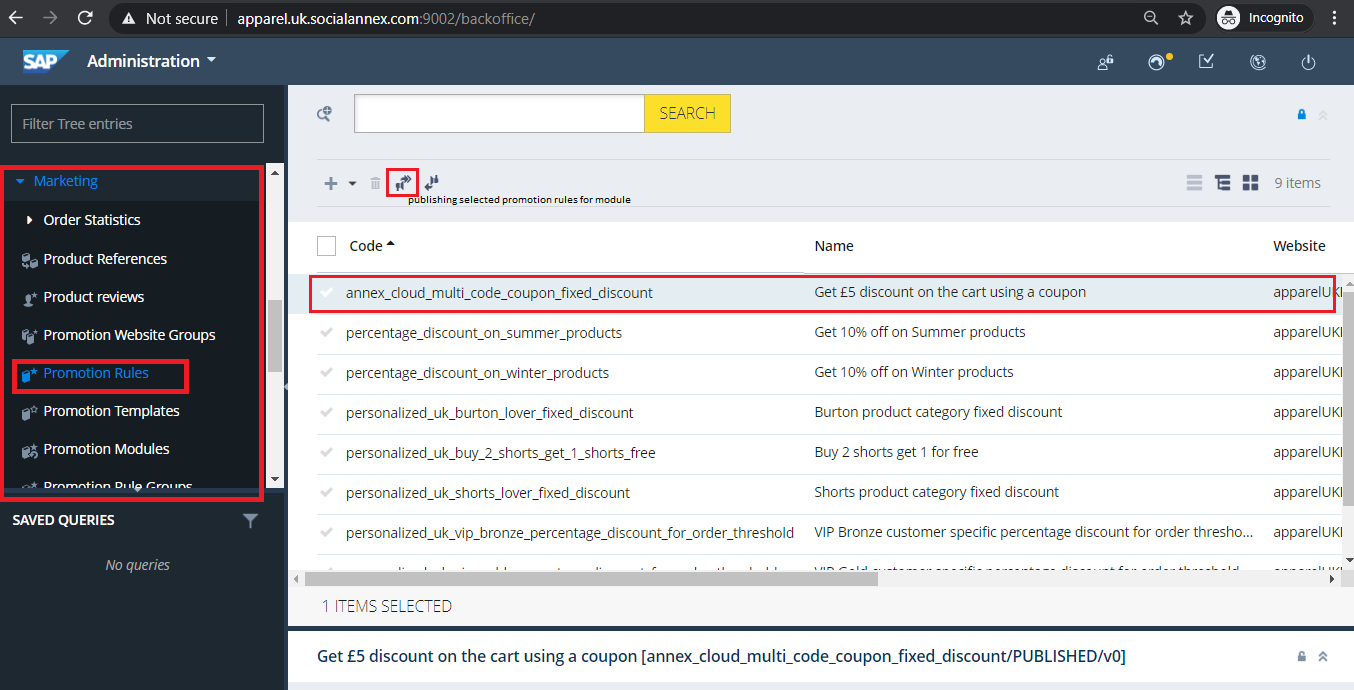
*https://localhost:9002/*

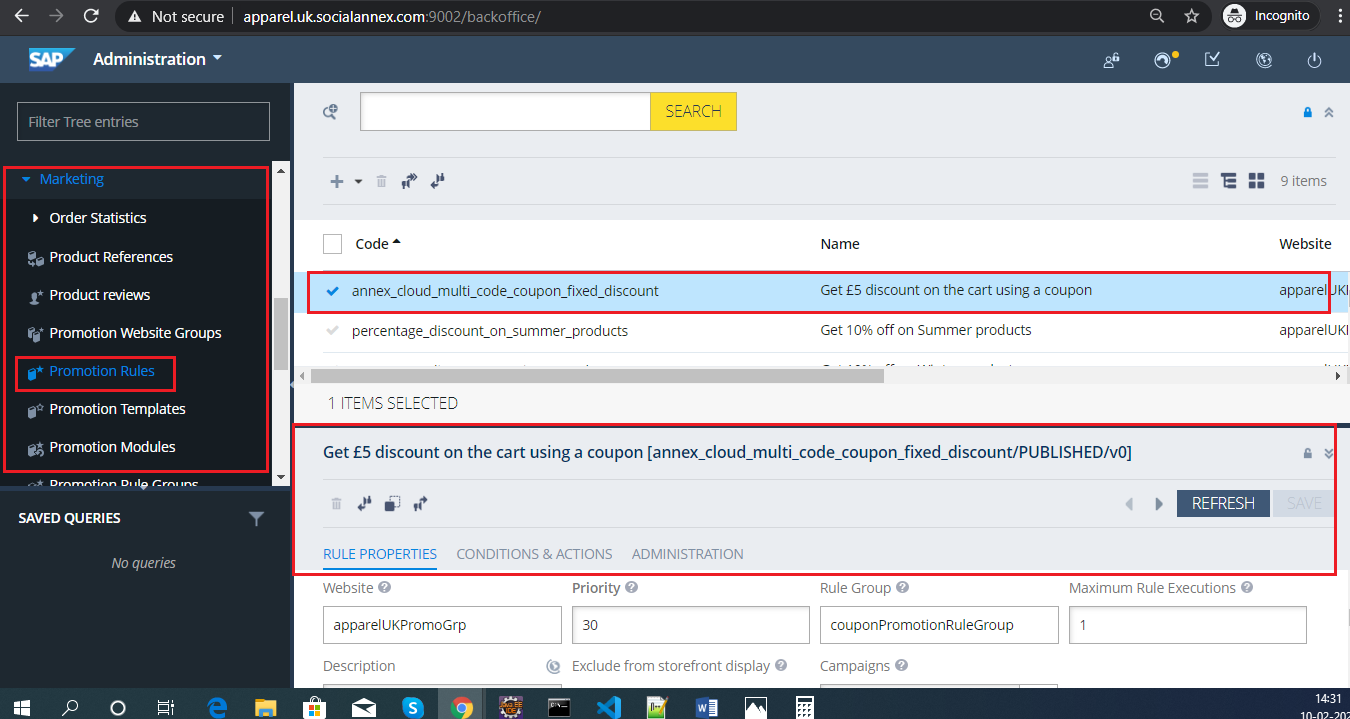
* It will display the hybris administration console.
* Go to console
* Import the impex
* The impex is given below.

|  |
| --- |
| # -----------------------------------------------------------------------  # [y] hybris Platform  #  # Copyright (c) 2018 SAP SE or an SAP affiliate company. All rights reserved.  # This software is the confidential and proprietary information of SAP  # ("Confidential Information"). You shall not disclose such Confidential  # Information and shall use it only in accordance with the terms of the  # license agreement you entered into with SAP.  # -----------------------------------------------------------------------  # ImpEx for Promotions for Electronics Store  #% impex.setLocale( Locale.GERMAN );  $defaultPromoGrp=apparelUKPromoGrp  $promotionsModule="promotions-module"  $previewModule="promotions-preview-module"  $ruleImportProcessor=processor=de.hybris.platform.ruleengineservices.impex.impl.ConditionalImportProcessor,condition=de.hybris.platform.ruleengineservices.impex.impl.RuleImportCondition  ############################# PROMOTION RULES ##########################  INSERT\_UPDATE PromotionSourceRule[$ruleImportProcessor];code[unique=true];priority;maxAllowedRuns;ruleGroup(code);conditions;actions;website(Identifier)[default=$defaultPromoGrp];excludeFromStorefrontDisplay[default=false];status(code)[unique=true,default=UNPUBLISHED]  ;annex\_cloud\_multi\_code\_coupon\_fixed\_discount;30;1;couponPromotionRuleGroup;"[{""definitionId"":""y\_qualifying\_coupons"",""parameters"":{""coupons"":{""uuid"":""c4988a76-c0fc-4de3-a801-6f3f735e93ae"",""type"":""List(ItemType(AbstractCoupon))"",""value"":[""ACCOUPON5""]}},""children"":[]}]";"[{""definitionId"":""y\_order\_fixed\_discount"",""parameters"":{""value"":{""uuid"":""4c21e41f-6933-471a-8cba-3f87cf83c97d"",""type"":""Map(ItemType(Currency),java.math.BigDecimal)"",""value"":{""GBP"":5}}}}]";;  ############################# COUPON SAMPLE DATA ##########################  INSERT\_UPDATE CodeGenerationConfiguration;name[unique=true];codeSeparator;  ;default-configuration;-;  INSERT\_UPDATE MultiCodeCoupon;couponId(id)[unique=true];codeGenerationConfiguration(name)  ;ACCOUPON5;default-configuration  # Language  $lang=en  $ruleImportProcessor=processor=de.hybris.platform.ruleengineservices.impex.impl.ConditionalImportProcessor,condition=de.hybris.platform.ruleengineservices.impex.impl.RuleImportCondition  ############################# PROMOTION RULES ##########################  INSERT\_UPDATE PromotionSourceRule[$ruleImportProcessor];code[unique=true];name[lang=$lang];description[lang=$lang];messageFired[lang=$lang];status(code)[unique=true,default=UNPUBLISHED]  ;annex\_cloud\_multi\_code\_coupon\_fixed\_discount;Get £5 discount on the cart using a coupon;A fixed discount is added to the cart when a valid multi code coupon code is entered;You have received Annex Cloud {4c21e41f-6933-471a-8cba-3f87cf83c97d} discount on cart using the coupon code {c4988a76-c0fc-4de3-a801-6f3f735e93ae}  INSERT\_UPDATE MultiCodeCoupon;couponId(id)[unique=true];name[lang=$lang]  ;ACCOUPON5;Get £5 discount on the cart using the coupon |

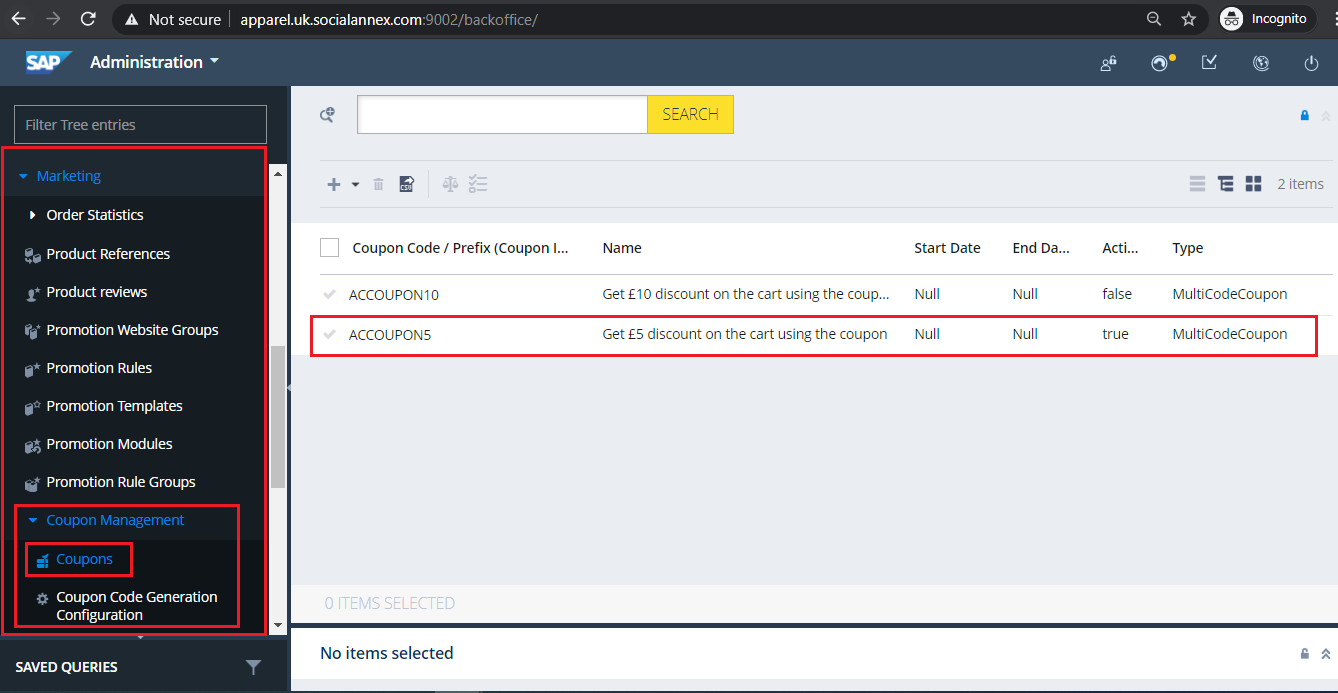


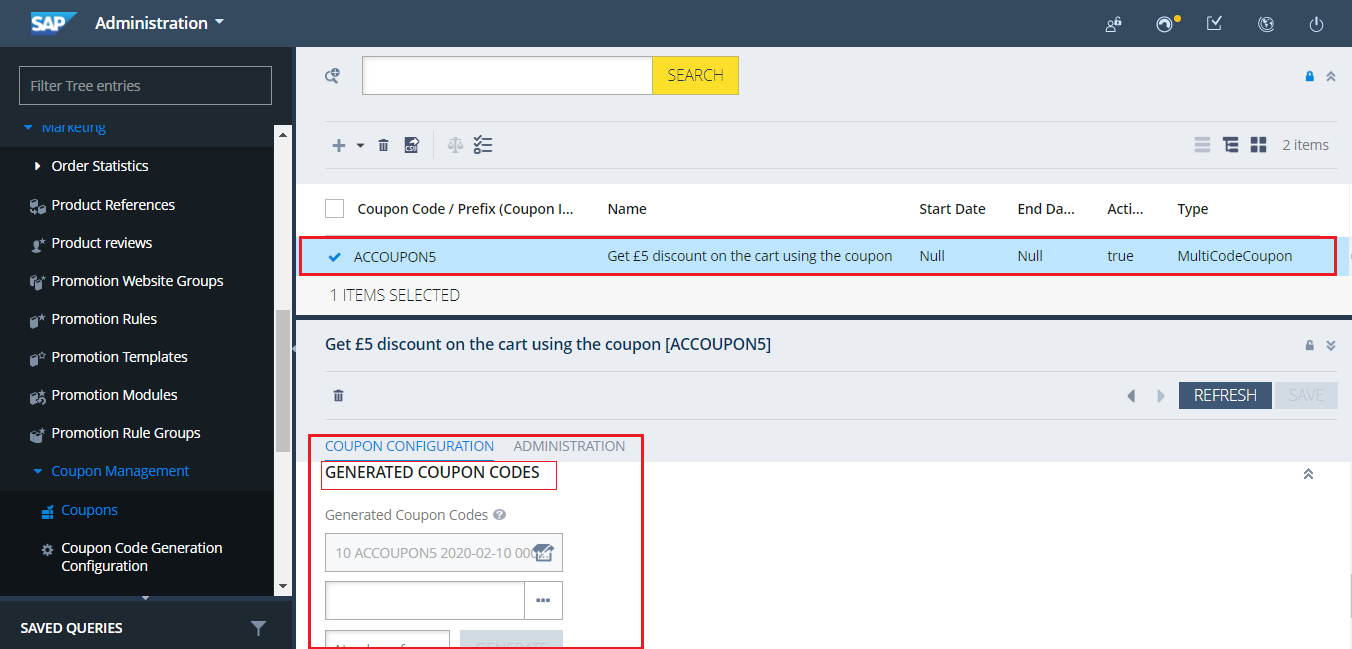
* After importing the impex, go to the back office and follow the below steps.
* Select the impex and click on publish selected promotion rule for module as shown in the following image.



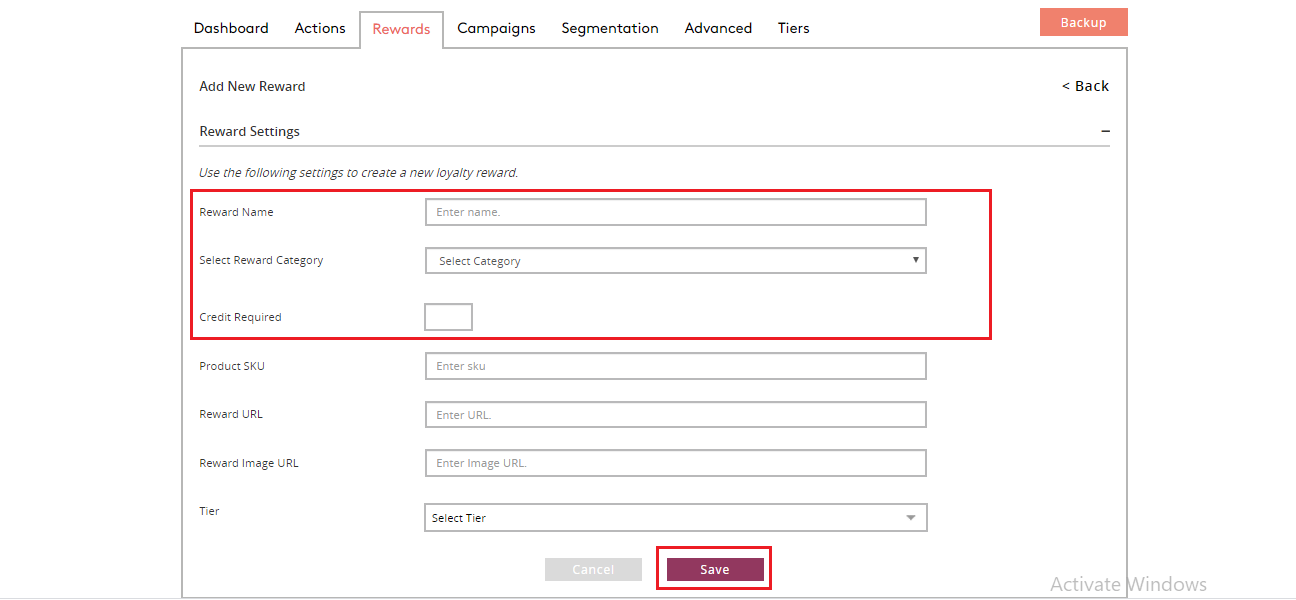


* After this, in marketing, go to coupon management and follow the below steps:
* Select the coupon and then go to General Coupon Codes
* Enter total number of coupons required.
* Click on Generate button.
* A CSV file will generate consisting the coupons.

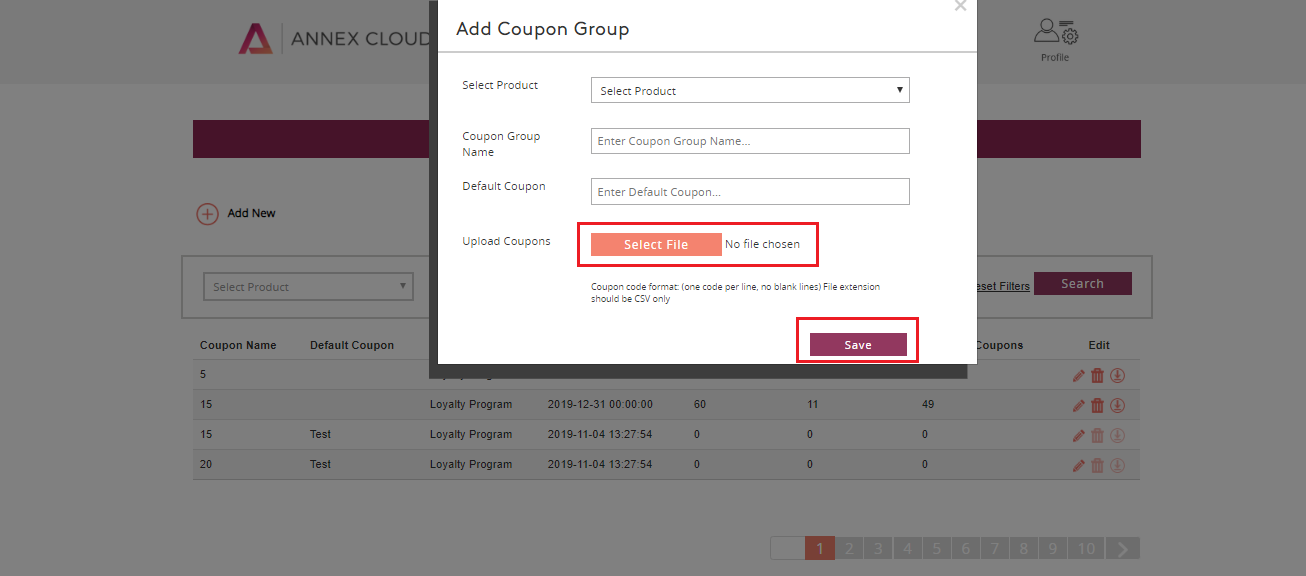




* + 1. Generate Rewards & Upload the Coupon Codes in Annex Cloud
* To create rewards, go to [annexcloud.com](http://www.annexcloud.com) and login.
* Navigate to **Products 🡺 Customer Loyalty 🡺 Rewards 🡺 Add New**. It will open the following window.



* Enter the required details and click on the Save button. It will create the reward.
* Now, to upload the csv file (coupons), navigate to **Options 🡺 Coupon Groups 🡺 Add New**. It will open the following window.



* Enter the required details and upload the csv file (generated in the above section) and click on the Save button.
* It will upload the csv file.

1. Checkout Flow

* Annex Cloud Checkout Group:
* We have created **annexCloudCheckoutGroup** into **annexcloud-multi-step checkoutconfig.xml** file.
* Add the **Transition Bean** for any checkout step to link with previous, next and current step.
* Need to add the **Redirects bean** for the checkout step.
* Then, add the key\_value pair in this Property
  + - **<property name =”checkoutStepMap”>.**
* Add the key\_value pair in this Property
  + - **<property name =”validationResultsMap”>.**
* Add the key\_value pair in this Property
  + - **<property name =”checkoutProgressBar”>.**
* All these properties come under a Checkout Group i.e. annexCloudCheckoutGroup.

1. Order Flow

* **Annex Cloud Send Order Steps:**
* Createdan action class “SendOrderDetailToAnnexCloudAction.java” in  annexcloudv3fulfilmentprocess extension
* Qualified Path:
  + - /annexcloudv3fulfilmentprocess/src/com/annexcloud/actions/order/SendOrderCompletedNotificationAction.java
* Need to add the bean for Class in the order-process-spring.xml file:
* Qualified Path:
  + - /annexcloudv3fulfilmentprocess/resources/annexcloudv3fulfilmentprocess/process/orderprocessspring.xml
* Then create action bean for the same in annexcloud-order-process.xml
* Qualified Path:
  + - /annexcloudv3fulfilmentprocess/resources/annexcloudv3fulfilmentprocess/process/annexcloudorder-process.xml.
* In action bean for Transition.OK, go to checkAuthorizedOrderPayment.
* Transition. NOK goes to error.
* **Annex Cloud Return Order Steps:**
* Created an action class “AnnexCloudReturnOrderAction.java” in
  + - annexcloudv3fulfilmentprocess extension.
* Qualified Path:
  + - /annexcloudv3fulfilmentprocess/src/com/annexcloud/actions/returns/AnnexCloudReturnOrderAction.java
* Need to add the bean for Class in order-process-spring.xml File.
* Qualified Path:
  + - /annexcloudv3fulfilmentprocess/resources/annexcloudv3fulfilmentprocess/process/orderprocessspring.xml
* Then create action bean for the same in annexcloud-order-process.xml
* Qualified Path:
  + - /annexcloudv3fulfilmentprocess/resources/annexcloudv3fulfilmentprocess/process/annexcloudorder-process.xml.
* In action bean for Transition.OK, go to transactionReviewStatus.
* Transition.NOK goes to error.