E-Gift Certificate

Version 22.1.0
SFRA version 6.0.0
Compatibility Mode: 21.7

Developed by: Aqsa Shahzad & Noor Ul Ain

Table of Contents

1. Summary

This custom cartridge is an implementation of the E-Gift Certificate functionality which handles the purchase of Gift Certificate via the storefront. The gift certificate is sent to the recipient via email which will include the unique gift certificate code that can be used to redeem it. It also includes a feature to check the remaining balance on the Gift Certificate and redeem it as a payment method.

This document details the integration process for this custom cartridge. This document provides technical instructions for installing E-Gift Certificate custom cartridge and user guide for using this cartridge to purchase gift certificates directly from the website, check its balance and redeem it as a payment method, all from the site.

All custom features are described in detail in this document below.

2. Component Overview

Functional Overview

The cartridge is developed to enable the customer to purchase gift certificates from the site, the customer can simply fill out a form and add the gift certificate to their cart and pay for it in the checkout flow as they would for any product. The gift certificate is sent via email to the recipient specified by the customer. They can set the amount of the gift certificate to their liking (up to a limit) and can also add a message to send to the recipient along with the gift certificate.

Once the form is filled and validated, Gift Certificate object is made at the back end as well as a Gift Certificate Line Item. This Gift Certificate Line Item is added to cart, just as a Product Line Item. And the checkout flow is completed as usual.

Once the order is confirmed, an email is sent to the Gift Certificate recipient, containing the gift certificate code, the expiry date and a link to the store where they can redeem the Gift Certificate. It will also tell the customer from what date they can start using this Gift Certificate because the Gift Certificate is not enabled at the back end from the get-go, but it is enabled after some days (configurable by the merchant) as a precautionary measure to ensure that the gift certificate is only used after the payment is captured.

Similarly, the expiry date is also configurable and the merchant can decide for how long they want the gift certificate be available to the customer. Also, the customer is also notified by the store when the expiry date is nearing.

The gift certificates can be redeemed on the site and used as a payment method. It also enables the customers to check the balance on their gift certificate on the site using their gift certificate code. So, if they have redeemed it, they can check how much amount is left.

Use Cases

The use cases for this custom cartridge are listed below:

- On the storefront footer, a storefront customer should be able to see a “Gift Cards” option.
- As a Registered/Guest customer, storefront user should be able to purchase a gift card by adding information on the “Purchase a Gift Card” form, such as:
    - Amount
    - To (Recipient’s name)
    - Email (Recipient’s email ID)
    - Confirm Email (email validation)
    - Message (to send along with the gift certificate)
- As a Registered/Guest customer, storefront user should see the purchased gift card automatically added in his/her Cart, having same price as the amount that was added by the user while purchasing a gift card.
- As a registered/Guest customer, storefront user should be able to remove the gift card from their cart, as any other product.
- As a registered/Guest customer, storefront user should be able to see the gift card in the mini cart as well.
- As a Registered/Guest customer, storefront user should be able to pay for the gift card successfully via the payment method of their liking.
- The storefront user should not be able to see “Gift Card” as a payment method as they cannot pay for Gift Certificates with Gift Certificates.
- Gift card shall have an expiry date (configurable in Business Manager) which will be set automatically.
- A few days before the expiry (number of days configurable in the Business Manager), the customer shall receive emails to remind them of the expiry.
- Once the order is successfully placed by the Registered/Guest customer, an email should be sent by the store to the recipient of the Gift Card, giving them the following information:
    - Gift Card code

    - Sender's name
    - Sender's message
    - Gift Card expiry date
- Gift Certificate status in the system should be set to "Pending" and it is not enabled while the order is being placed, and its status is set to "Issued" after the order is confirmed but only after some time (days configurable in Business Manager) after the order is placed, it is enabled in the Business Manager to ensure that the payment is captured before the Gift Certificate is redeemed on the site.
- As a Registered/Guest customer, storefront user should be able to check their gift card remaining balance by entering their Gift Card Number into the "Check Balance" form.
- As a Registered/Guest customer, the storefront user should be able to redeem the gift certificate via payment method in checkout flow.
- The "Check Balance" functionality should also be able available on the checkout page, to see the remaining available balance during the checkout flow.

Limitations, Constraints

- The current version supports for the default locale en_US.

Compatibility

The custom cartridge is designed for Salesforce Commerce Cloud API Version 21.9 (Compatibility Mode: 21.7) with controllers for extended functionality. The cartridge is tested for SFRA 6.0.0.

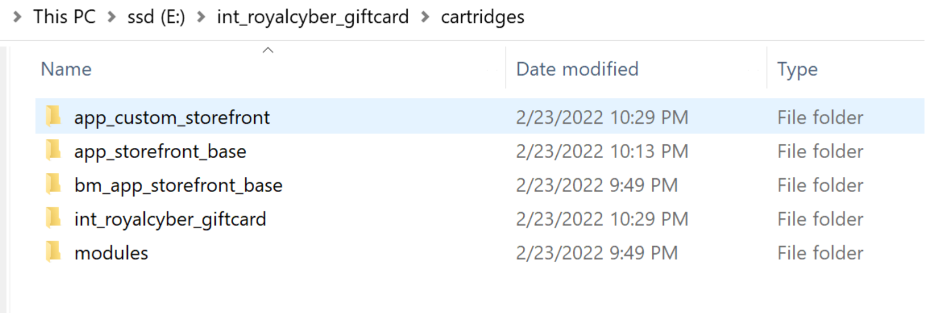
Privacy, Payment

The customer's information is saved in a custom object and accessed when sending expiry reminder emails.

3. Implementation Guide

Setup of Business Manager

1. Extract the zip file.
2. Place the folder "app_custom_storefront" and "int_royalcyber_giftcard" inside the folder "cartridges" of int_royalcyber_giftcard.



> This PC > ssd (E:) > int_royalcyber_giftcard > cartridges

Name	Date modified	Type
app_custom_storefront	2/23/2022 10:29 PM	File folder
app_storefront_base	2/23/2022 10:13 PM	File folder
bm_app_storefront_base	2/23/2022 9:49 PM	File folder
int_royalcyber_giftcard	2/23/2022 10:29 PM	File folder
modules	2/23/2022 9:49 PM	File folder

3. Open the file "webpack.config.js" present in the root folder of storefront-reference-architecture and replace "app_storefront_base" with "int_royalcyber_giftcard" in js file.

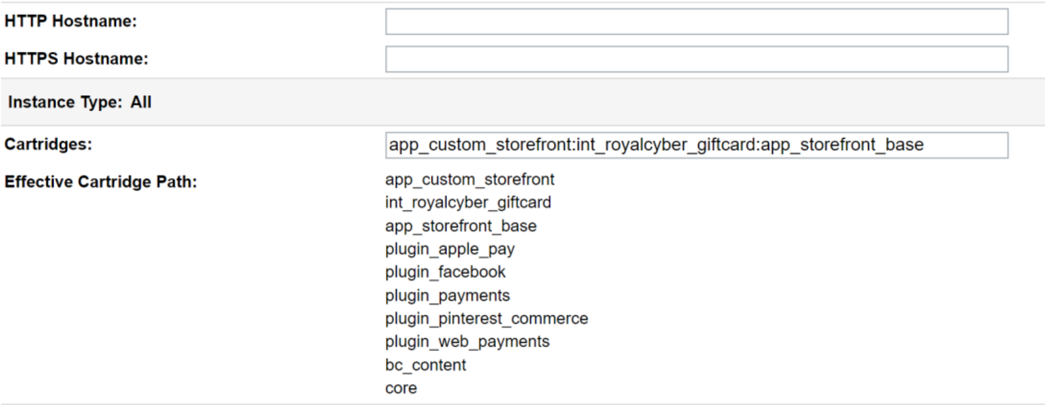
4. Open the file “package.json” present in the root folder of storefront-reference-architecture and set the following two variables:

```
"packageName": " int_royalcyber_giftcard ",
"paths": {
  "base": "./cartridges/app_storefront_base/"
 }
```

5. Compile js and scss for this custom cartridge, execute the following at root folder of storefront-reference-architecture:

```
npm run compile:js && npm run compile:scss
```

6. Upload cartridge to the Salesforce Commerce Cloud instance.
7. Login to Business Manager and activate the code version with the uploaded cartridges.
8. Add app_custom_storefront and int_royalcyber_giftcard in the cartridge path for the target site as show below in the screenshot.



HTTP Hostname:	
HTTPS Hostname:	
Instance Type: All	
Cartridges:	app_custom_storefront:int_royalcyber_giftcard:app_storefront_base
Effective Cartridge Path:	app_custom_storefront int_royalcyber_giftcard app_storefront_base plugin_apple_pay plugin_facebook plugin_payments plugin_pinterest_commerce plugin_web_payments bc_content core

9. Apply changes.
10. In the Business Manager, select the target site and then navigate to Merchant Tools> Content> Content Assets.
11. In the search field enter the ID “footer-account” and click “Find”.
12. Select the “footer-account” content asset.
13. Click the “lock” because you will be editing the content asset.
14. In the “General” tab, scroll down to the “Content” section and copy the highlighted code in the “Body” field after Gift Registry as shown below

```
<div><button class="title btn text-left btn-block d-sm-none" aria-expanded="false" aria-
controls="collapsible-account">Account</button><h2 class="title d-none d-sm-
block">Account</h2></div>
        <ul id="collapsible-account"class="menu-footer content">
        <li><a href="$httpsURL('Account-Show')$" title="Go to My Account">My Account</a></li>
        <li><a href="$httpsURL('Order-History')$" title="Go to Check Order">Check Order</a></li>
        <!--<li><a href="$httpsURL('Wishlist-Search')$" title="Go to Wishlist">Find a
Wishlist</a></li>-->
        <!--<li><a href="$httpsURL('GiftRegistry-Landing')$" title="Go to Gift Registry">Gift
Registry</a></li>-->
        <li><a href="$httpsURL('Giftcard-Show')$" title="Go to Giftcard">Gift Card</a></li>
        </ul><!-- END: footer-account -->
```

15. Click “Apply”.
16. Now scroll up and click “unlock”.

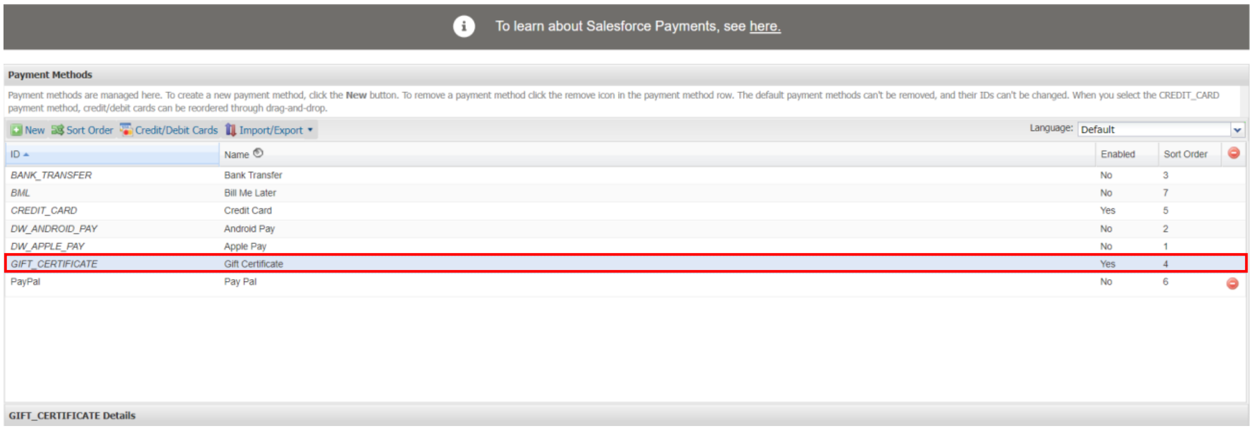
Configuration

1. Navigate to Administration -> Site Development -> Site Import & Export.
2. Upload system_custom_objects_jobs_metadata.zip from meta directory.
3. Import system_custom_objects_jobs_metadata.zip.

Enable Payment Method

1. Navigate to Merchant Tools > Ordering > Payment Methods.
2. Locate *GIFT_CERTIFICATE* payment method.
3. Change Enabled value to “Yes” if it is “No” by clicking on it and selecting from drop down.
4. Refer to the screen shot below:

Payment Methods



To learn about Salesforce Payments, see here.

Payment Methods

Payment methods are managed here. To create a new payment method, click the **New** button. To remove a payment method click the remove icon in the payment method row. The default payment methods can't be removed, and their IDs can't be changed. When you select the CREDIT_CARD payment method, credit/debit cards can be reordered through drag-and-drop.

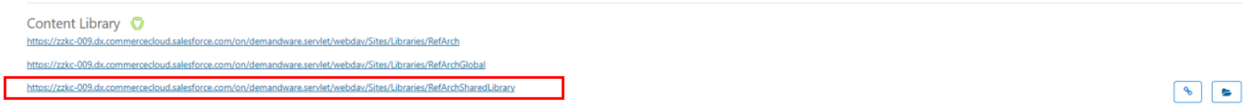
Language: Default

ID	Name	Enabled	Sort Order
BANK_TRANSFER	Bank Transfer	No	3
BML	Bill Me Later	No	7
CREDIT_CARD	Credit Card	Yes	5
DW_ANDROID_PAY	Android Pay	No	2
DW_APPLE_PAY	Apple Pay	No	1
GIFT_CERTIFICATE	Gift Certificate	Yes	4
PayPal	Pay Pal	No	6

GIFT_CERTIFICATE Details

Upload Images

1. Navigate to Administration > Site Development > Development Setup.
2. Scroll down to Content Library section and click on the link highlighted in the screen shot below.



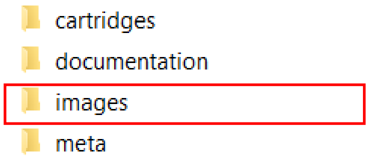
Content Library

https://zzkc-009.dx.commercecloud.salesforce.com/on/demandware.servlet/webdav/Sites/Libraries/RefArch

https://zzkc-009.dx.commercecloud.salesforce.com/on/demandware.servlet/webdav/Sites/Libraries/RefArchGlobal

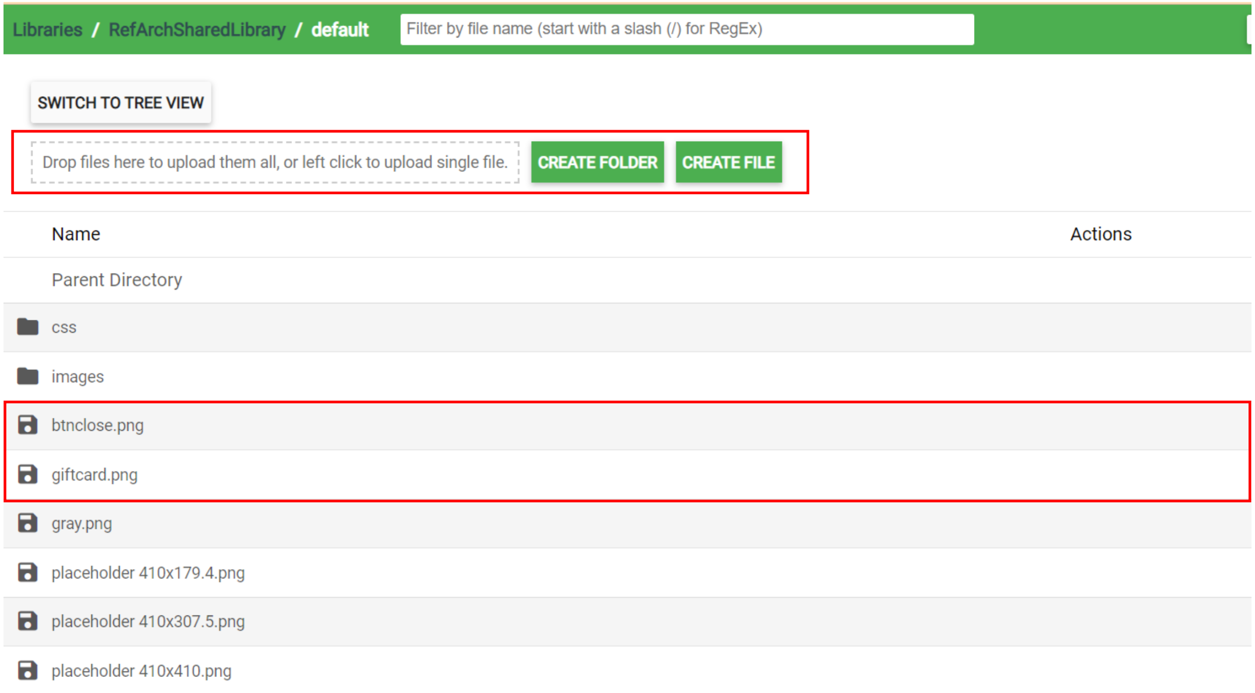
https://zzkc-009.dx.commercecloud.salesforce.com/on/demandware.servlet/webdav/Sites/Libraries/RefArchSharedLibrary

3. After login (if required) go to “default” folder.
4. Note there is an “images” folder in “link_royalcyber_gc” root directory.



cartridges
documentation
images
meta

5. Drag the pictures from “images” folder to the “default” folder of “RefArchSharedLibrary”.


Libraries / RefArchSharedLibrary / default
Filter by file name (start with a slash (/) for RegEx)
SWITCH TO TREE VIEW
Drop files here to upload them all, or left click to upload single file.
CREATE FOLDER
CREATE FILE

Name	Actions
Parent Directory	
css	
images	
btnclose.png	
giftcard.png	
gray.png	
placeholder 410x179.4.png	
placeholder 410x307.5.png	
placeholder 410x410.png	

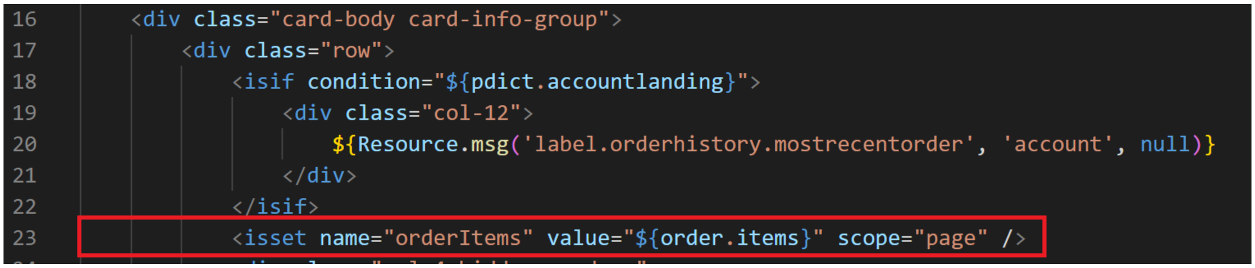
Custom Code

Changes in some templates of the base cartridge were required the templates that have been customized are:

- account\order\orderHistoryCard.isml
- account\order\orderList.isml
- account\order\shippingTile.isml
- cart\cart.isml
- checkout\billing\creditCardForm.isml
- checkout\billing\paymentOptions\paymentOptionsContent.isml
- checkout\billing\paymentOptions\ paymentOptionsSummary.isml
- checkout\billing\paymentOptions\ paymentOptionsTabs.isml
- checkout\cart\miniCart.isml
- checkout\confirmation\confirmationDetails.isml
- checkout\confirmation\confimationEmail.isml
- checkout\shipping\shipping.isml
- checkout\shipping\shippingSummary.isml
- checkout\checkout.isml
- checkout\orderProductSummary.isml
- common\layout\page.isml

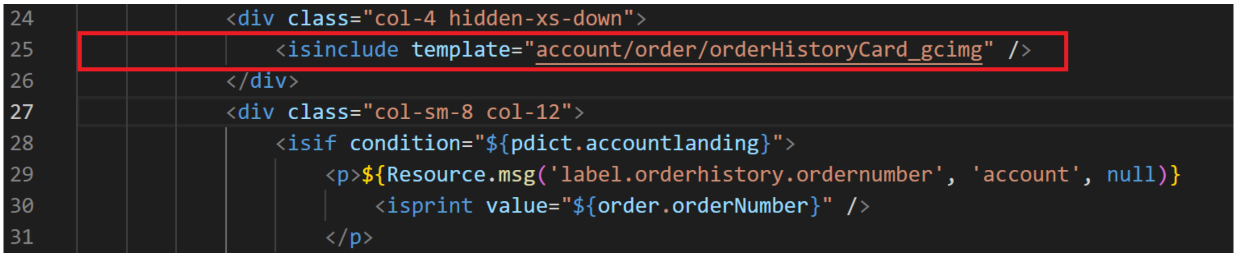
Below changes should be made in your cartridge above int_royalcyber_giftcard for e-gift certification.

1. Changes for account\order\orderHistoryCard.isml
    - Insert given snippet `<isset name="orderItems" value="${order.items}" scope="page" />` in orderHistoryCard.isml below the closing tag of `<isif condition = "${pdict.accountlanding}">.` Refer to the below screenshot.



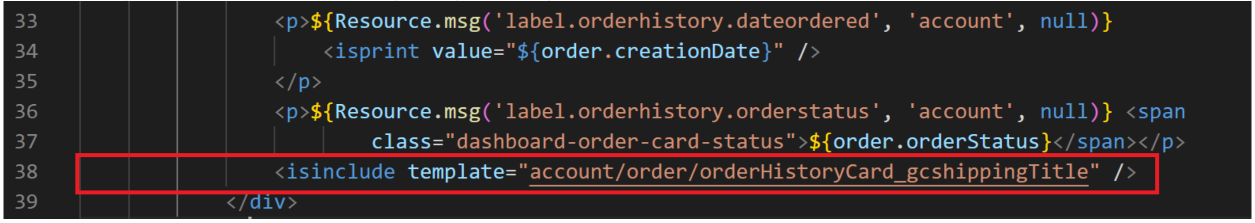
```
<div class="card-body card-info-group">
    <div class="row">
        <isif condition="${pdict.accountlanding}">
            <div class="col-12">
                ${Resource.msg('label.orderhistory.mostrecentorder', 'account', null)}
            </div>
        </isif>
        <isset name="orderItems" value="${order.items}" scope="page" />
```

- Insert given snippet `<isinclude template ="account/order/orderHistoryCard_gcimg" />` in orderHistoryCard.isml replacing the content of `<div class="col-4 hidden-xs-down">`. Refer to the below screenshot.



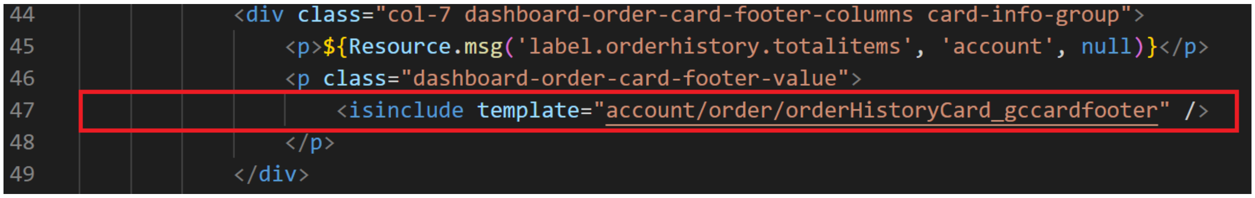
```
<div class="col-4 hidden-xs-down">
    <isinclude template="account/order/orderHistoryCard_gcimg" />
</div>
<div class="col-sm-8 col-12">
    <isif condition="${pdict.accountlanding}">
        <p>${Resource.msg('label.orderhistory.ordernumber', 'account', null)}
            <isprint value="${order.orderNumber}" />
        </p>
```

- Insert given snippet `<isinclude template ="account/order/orderHistoryCard_gcshippingTitle" />` in orderHistoryCard.isml below the line of `<p>${Resource.msg('label.orderhistory.orderstatus', 'account', null)} <span class="dashboard-order-card status"> ${order.orderStatus}</span></p>`. Refer to the below screenshot.



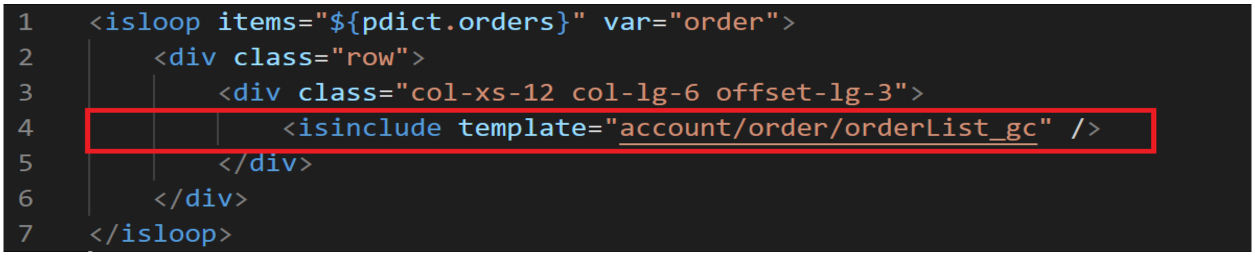
```
    <p>${Resource.msg('label.orderhistory.dateordered', 'account', null)}
        <isprint value="${order.creationDate}" />
    </p>
    <p>${Resource.msg('label.orderhistory.orderstatus', 'account', null)} <span
            class="dashboard-order-card-status">${order.orderStatus}</span></p>
    <isinclude template="account/order/orderHistoryCard_gcshippingTitle" />
</div>
```

- Insert given snippet `<isinclude template ="account/order/orderHistoryCard_gccardfooter" />` in orderHistoryCard.isml replacing the content of `<p class="dashboard-order-card-footer-value">`. Refer to the below screenshot.



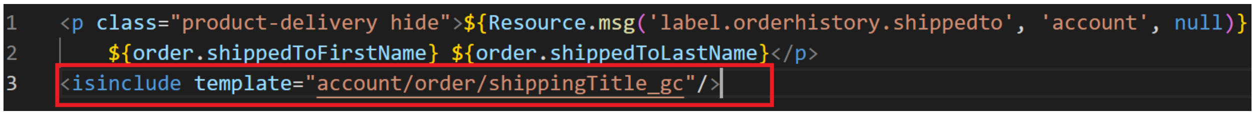
```
<div class="col-7 dashboard-order-card-footer-columns card-info-group">
    <p>${Resource.msg('label.orderhistory.totalitems', 'account', null)}</p>
    <p class="dashboard-order-card-footer-value">
        <isinclude template="account/order/orderHistoryCard_gccardfooter" />
    </p>
</div>
```

2. Changes for account\order\orderList.isml
   - Insert given snippet `<isinclude template="account/order/orderList_gc" />` in orderList.isml replacing the content of `<div class="col-xs-12 col-lg-6 offset-lg-3">`. Refer to the below screenshot.



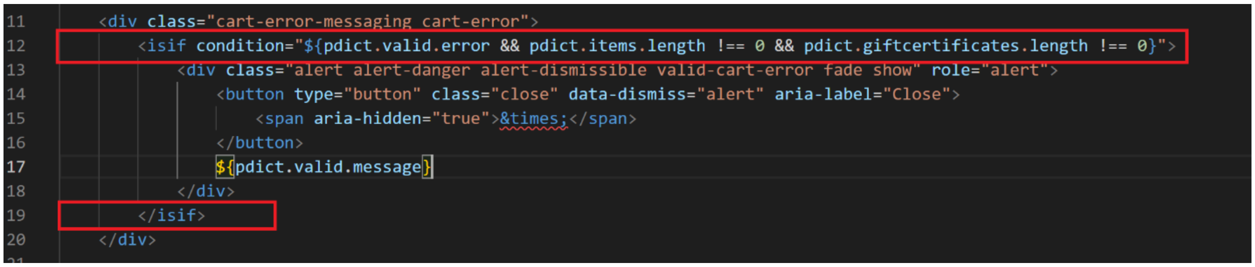
```
<isloop items="${pdict.orders}" var="order">
    <div class="row">
        <div class="col-xs-12 col-lg-6 offset-lg-3">
            <isinclude template="account/order/orderList_gc" />
        </div>
    </div>
</isloop>
```

3. Change for account\order\shippingTile.isml
    - Insert given snippet `<isinclude template="account/order/shippingTitle_gc"/>` in shippingTile.isml below the line of `<p class="product-delivery hide">${Resource.msg('label.orderhistory.shippedto', 'account', null)} ${order.shippedToFirstName} ${order.shippedToLastName}</p>`. Refer to the below screenshot.



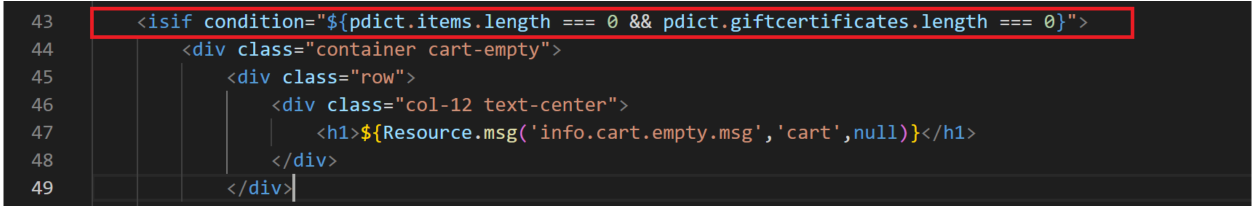
```
<p class="product-delivery hide">${Resource.msg('label.orderhistory.shippedto', 'account', null)}
    ${order.shippedToFirstName} ${order.shippedToLastName}</p>
<isinclude template="account/order/shippingTitle_gc"/>
```

4. Changes for cart\cart.isml
    - Modify the given snippet `<isif condition="${pdict.valid.error && pdict.items.length !== 0 && pdict.giftcertificates.length !== 0}">` in cart.isml below the line of `<div class="cart-error-messaging cart-error">`. Refer to the below screenshot.



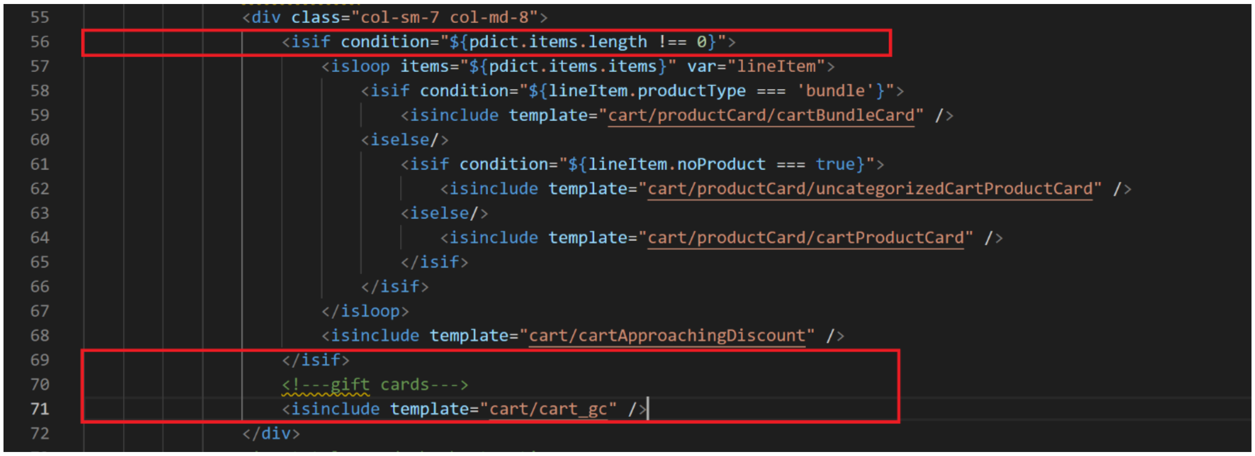
```
<div class="cart-error-messaging cart-error">
    <isif condition="${pdict.valid.error && pdict.items.length !== 0 && pdict.giftcertificates.length !== 0}">
        <div class="alert alert-danger alert-dismissible valid-cart-error fade show" role="alert">
            <button type="button" class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">&times;</span>
            </button>
            ${pdict.valid.message}
        </div>
    </isif>
</div>
```

- Modify the given snippet `<isif condition="${pdict.items.length === 0 && pdict.giftcertificates.length === 0}">` in cart.isml above the line of `<div class="container cart-empty">`. Refer to the below screenshot.



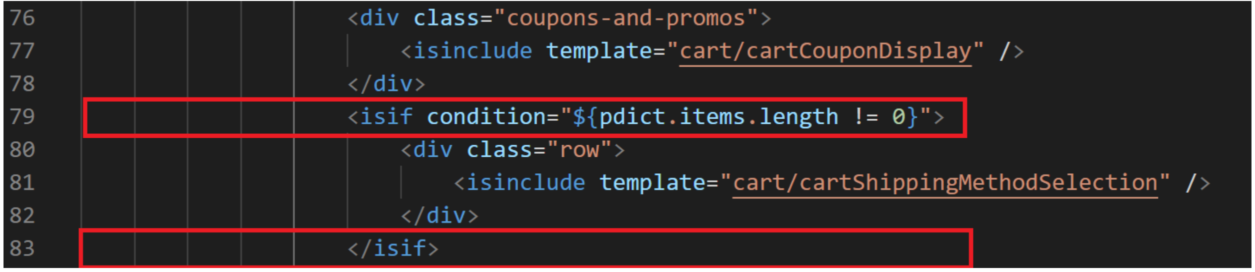
```
<isif condition="${pdict.items.length === 0 && pdict.giftcertificates.length === 0}">
    <div class="container cart-empty">
        <div class="row">
            <div class="col-12 text-center">
                <h1>${Resource.msg('info.cart.empty.msg','cart',null)}</h1>
            </div>
        </div>
```

- Insert given snippet `<isif condition="${pdict.items.length !== 0}">` in cart.isml below the line of `<div class="col-sm-7 col-md-8">` and insert the given snippet `<isinclude template="cart/cart_gc" />` after the closing tag `</isif>` above the div closing tag. Refer to the below screenshot. Also notice line 57 in the screen shot where `<isloop items="${pdict.items.items}" var="lineItem">`



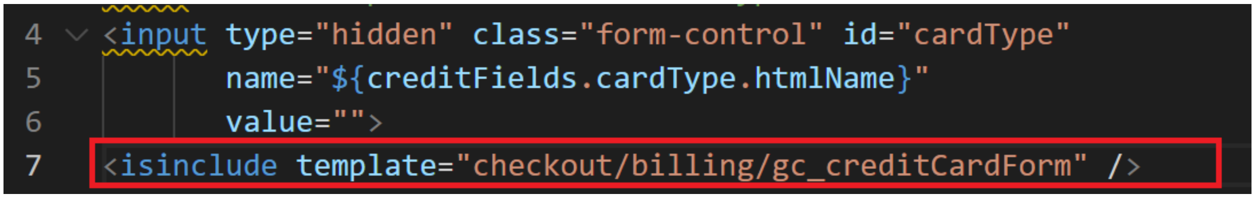
```
<div class="col-sm-7 col-md-8">
    <isif condition="${pdict.items.length !== 0}">
        <isloop items="${pdict.items.items}" var="lineItem">
            <isif condition="${lineItem.productType === 'bundle'}">
                <isinclude template="cart/productCard/cartBundleCard" />
            <iselse/>
                <isif condition="${lineItem.noProduct === true}">
                    <isinclude template="cart/productCard/uncategorizedCartProductCard" />
                <iselse/>
                    <isinclude template="cart/productCard/cartProductCard" />
                </isif>
            </isif>
        </isloop>
        <isinclude template="cart/cartApproachingDiscount" />
    </isif>
    <!---gift cards--->
    <isinclude template="cart/cart_gc" />
</div>
```

- Insert given snippet `<isif condition="${pdict.items.length !== 0}">` in cart.isml below the line of `<div class="coupons-and-promos"> <isinclude template="cart/cartCouponDisplay" /></div>`. Refer to the below screenshot.



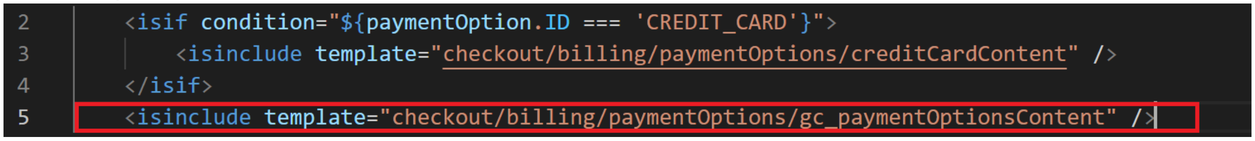
```
<div class="coupons-and-promos">
    <isinclude template="cart/cartCouponDisplay" />
</div>
<isif condition="${pdict.items.length != 0}">
    <div class="row">
        <isinclude template="cart/cartShippingMethodSelection" />
    </div>
</isif>
```

5. Changes for checkout\billing\creditCardForm.isml
    - Insert given snippet `<isinclude template= "checkout/billing/gc_creditCardForm" />` in creditCardForm.isml below the line of `<input type="hidden" class="form-control" id="cardType" name ="${creditFields.cardType.htmlName}" value="">`. Refer to the below screenshot.



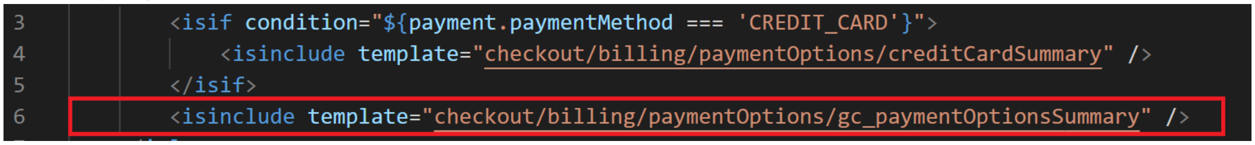
```
<input type="hidden" class="form-control" id="cardType"
        name="${creditFields.cardType.htmlName}"
        value="">
<isinclude template="checkout/billing/gc_creditCardForm" />
```

6. Changes for checkout\billing\paymentOptions\paymentOptionsContent.isml
    - Insert given snippet `<isinclude template="checkout/billing/paymentOptions/gc_paymentOptionsContent" />` in paymentOptionsContent.isml below the line `<isif condition="${paymentOption.ID === 'CREDIT_CARD'}"> <isinclude template="checkout/billing/paymentOptions/creditCardContent" /> </isif>`. Refer to the below screenshot.



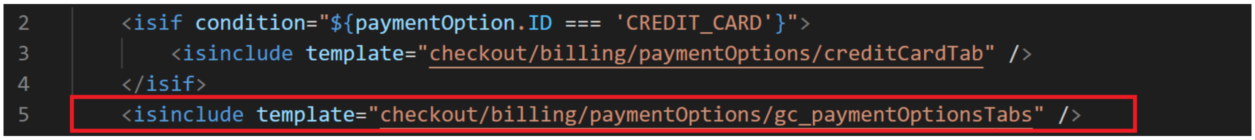
```
<isif condition="${paymentOption.ID === 'CREDIT_CARD'}">
    <isinclude template="checkout/billing/paymentOptions/creditCardContent" />
</isif>
<isinclude template="checkout/billing/paymentOptions/gc_paymentOptionsContent" />
```

7. Changes for checkout\billing\paymentOptions\paymentOptionsSummary.isml
   - Insert given snippet `<isinclude template="checkout/billing/paymentOptions/gc_paymentOptionsSummary" />` in paymentOptionsSummary.isml below the line `<isif condition="${payment.paymentMethod === 'CREDIT_CARD'}"><isinclude template="checkout/billing/paymentOptions/creditCardSummary" /></isif>`. Refer to the below screenshot.



```
    <isif condition="${payment.paymentMethod === 'CREDIT_CARD'}">
        <isinclude template="checkout/billing/paymentOptions/creditCardSummary" />
    </isif>
    <isinclude template="checkout/billing/paymentOptions/gc_paymentOptionsSummary" />
```

8. Changes for checkout\billing\paymentOptions\paymentOptionsTabs.isml
   - Insert given snippet `<isinclude template="checkout/billing/paymentOptions/gc_paymentOptionsTabs" />` in paymentOptionsTabs.isml below the line `<isif condition="${paymentOption.ID === 'CREDIT_CARD'}"> <isinclude template="checkout/billing/paymentOptions/creditCardTab" /></isif>`. Refer to the below screenshot.

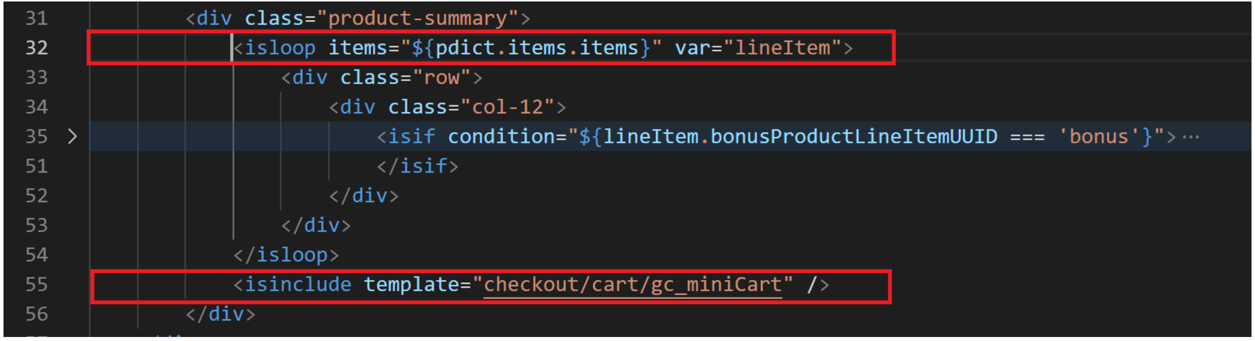


```
    <isif condition="${paymentOption.ID === 'CREDIT_CARD'}">
        <isinclude template="checkout/billing/paymentOptions/creditCardTab" />
    </isif>
    <isinclude template="checkout/billing/paymentOptions/gc_paymentOptionsTabs" />
```

9. Changes for checkout\cart\miniCart.isml
   - Insert the following script in top of the miniCart.isml.

```
<isscript>
        var assets = require('*/cartridge/scripts/assets.js');
       assets.addCss('/css/cart.css');
</isscript>
```

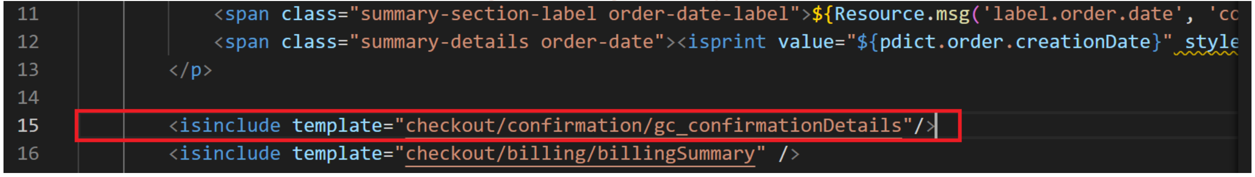
   - Modify the condition `<isloop items="${pdict.items.items}" var="lineItem">` below the line of `<div class="product-summary">` and insert the given snippet `<isinclude template="checkout/cart/gc_miniCart" />` in miniCart.isml after the isloop closing tag. Refer to the below screenshot.



```
        <div class="product-summary">
            <isloop items="${pdict.items.items}" var="lineItem">
                <div class="row">
                    <div class="col-12">
                        <isif condition="${lineItem.bonusProductLineItemUUID === 'bonus'}">…
                        </isif>
                    </div>
                </div>
            </isloop>
            <isinclude template="checkout/cart/gc_miniCart" />
        </div>
```

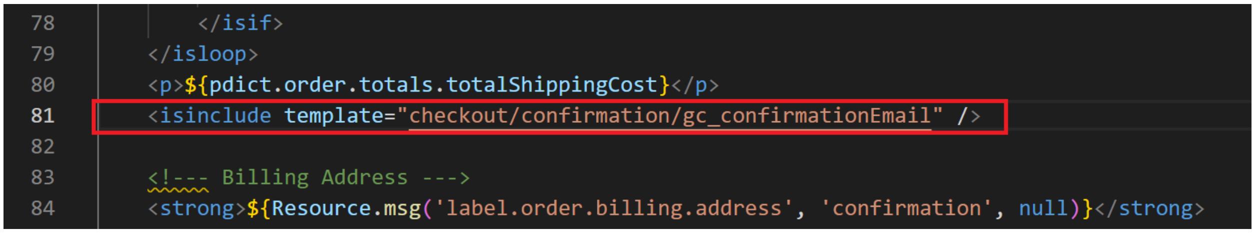
10. Changes for checkout\confirmation\confirmationDetails.isml
   - Insert the given snippet `<isinclude template ="checkout/confirmation/gc_confirmationDetails"/>` in

confirmationDetails.isml above the line of <isinclude template="checkout/billing/billingSummary" />. Refer to the screenshot.



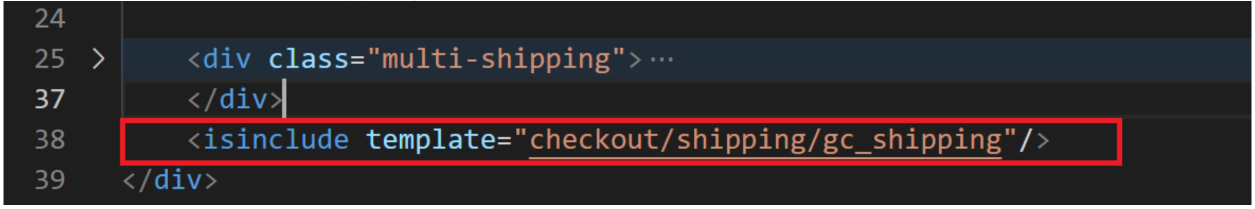
```
        <span class="summary-section-label order-date-label">${Resource.msg('label.order.date', 'cc
        <span class="summary-details order-date"><isprint value="${pdict.order.creationDate}" style
    </p>

    <isinclude template="checkout/confirmation/gc_confirmationDetails"/>
    <isinclude template="checkout/billing/billingSummary" />
```

11. Changes for checkout\confirmation\confirmationEmail.isml
    - Insert the given snippet <isinclude template="checkout/confirmation/gc_confirmationEmail"/> in confirmationEmail.isml below the line of <p>${pdict.order.totals.totalShippingCost}</p>. Refer to the screenshot.



```
        </isif>
    </isloop>
    <p>${pdict.order.totals.totalShippingCost}</p>
    <isinclude template="checkout/confirmation/gc_confirmationEmail" />

    <!--- Billing Address --->
    <strong>${Resource.msg('label.order.billing.address', 'confirmation', null)}</strong>
```

12. Changes for checkout\shipping\shipping.isml
    - Insert the given snippet <isinclude template="checkout/shipping/gc_shipping"/> in shipping.isml below the div of <div class="multi-shipping">. Refer to the screenshot.

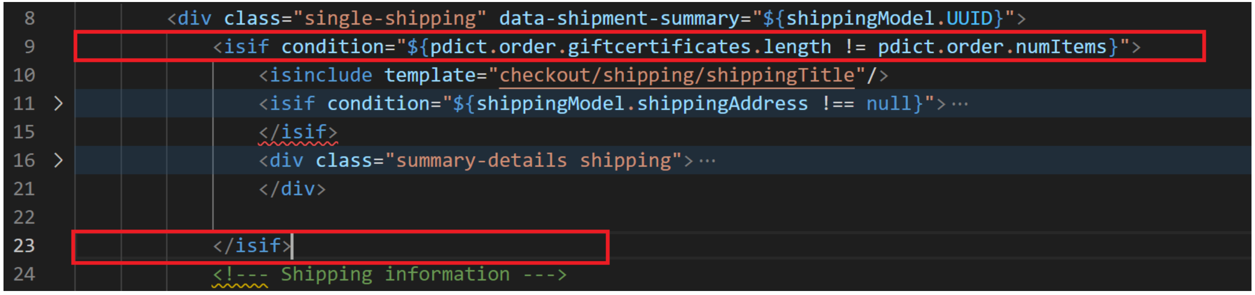


```

    <div class="multi-shipping">···
    </div>
    <isinclude template="checkout/shipping/gc_shipping"/>
</div>
```

13. Changes for checkout\shipping\shippingSummary.isml
    - Insert the following script in top of the shippingSummary.isml.

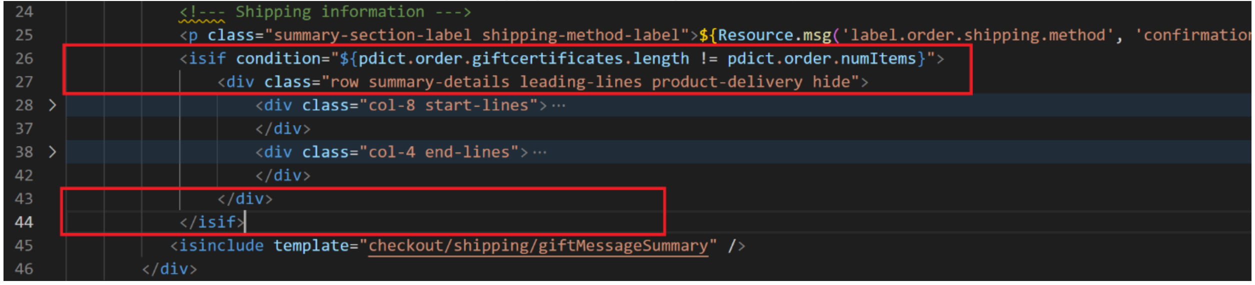
```
<isscript>
    var assets = require('*/cartridge/scripts/assets.js');
</isscript>
```

- Insert the given snippet <isif condition="${pdict.order.giftcertificates.length != pdict.order.numItems}"> below the line of <div class="single-shipping" data-shipment-summary="${shippingModel.UUID}"> and closing tag </isif> above shipping information in shippingSummary.isml. Refer to the screenshot.



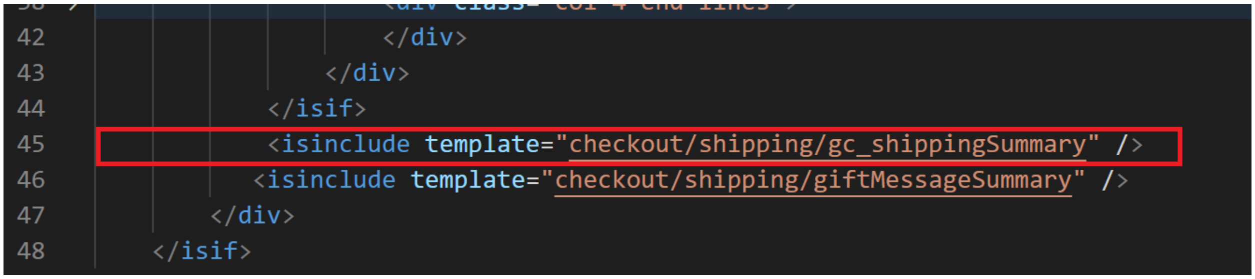
```
    <div class="single-shipping" data-shipment-summary="${shippingModel.UUID}">
        <isif condition="${pdict.order.giftcertificates.length != pdict.order.numItems}">
            <isinclude template="checkout/shipping/shippingTitle"/>
            <isif condition="${shippingModel.shippingAddress !== null}">···
            </isif>
            <div class="summary-details shipping">···
            </div>

        </isif>
        <!--- Shipping information --->
```

- Insert the given snippet `<isif condition ="${pdict.order.giftcertificates.length != pdict.order.numItems}"><div class="row summary-details leading-lines product-delivery hide">` in shippingSummary.isml below the line of `<p class="summary-section-label shipping-method-label">${Resource.msg('label.order.shipping.method', 'confirmation', null)}</p>` and closing tags above the line of `<isinclude template="checkout/shipping/giftMessageSummary" />`. Refer to the screenshot



```
            <!--- Shipping information --->
            <p class="summary-section-label shipping-method-label">${Resource.msg('label.order.shipping.method', 'confirmatio
            <isif condition="${pdict.order.giftcertificates.length != pdict.order.numItems}">
                <div class="row summary-details leading-lines product-delivery hide">
                    <div class="col-8 start-lines">⋯
                    </div>
                    <div class="col-4 end-lines">⋯
                    </div>
                </div>
            </isif>
           <isinclude template="checkout/shipping/giftMessageSummary" />
        </div>
```

- Insert given code snippet `<isinclude template ="checkout/shipping/gc_shippingSummary" />` in shippingSummary.isml above the line of `<isinclude template ="checkout/shipping/giftMessageSummary" />` . Refer to the screenshot below.

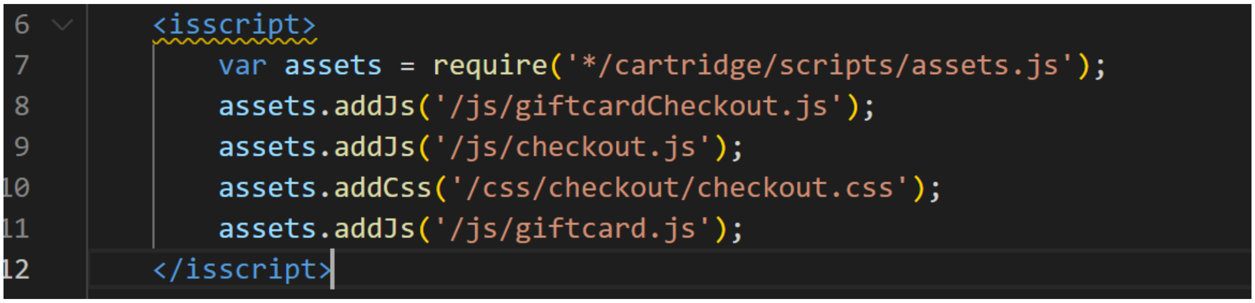


```
                        </div>
                    </div>
                </isif>
                <isinclude template="checkout/shipping/gc_shippingSummary" />
               <isinclude template="checkout/shipping/giftMessageSummary" />
            </div>
        </isif>
```

14. Changes for checkout\checkout.isml
    - Insert the following script in top of the checkout.isml.

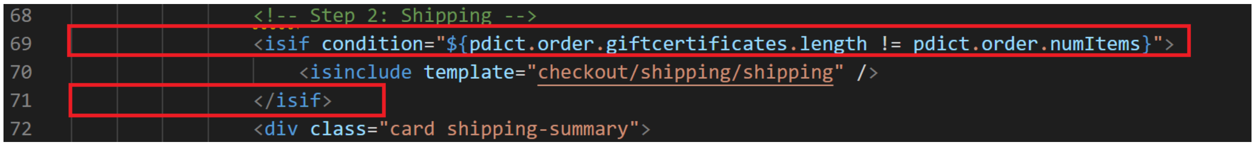
```
assets.addJs('/js/giftcardCheckout.js');
assets.addJs('/js/giftcard.js');
```

as shown in the screenshot.



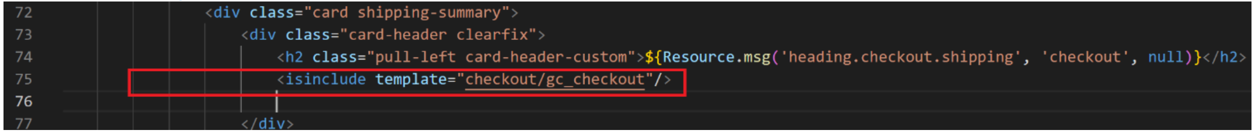
```
    <isscript>
        var assets = require('*/cartridge/scripts/assets.js');
        assets.addJs('/js/giftcardCheckout.js');
        assets.addJs('/js/checkout.js');
        assets.addCss('/css/checkout/checkout.css');
        assets.addJs('/js/giftcard.js');
    </isscript>
```

- Insert the given condition `<isif condition ="${pdict.order.giftcertificates.length != pdict.order.numItems}">` above the line of `<isinclude template="checkout/shipping/shipping" />` in checkout.isml. Refer to the screenshot.



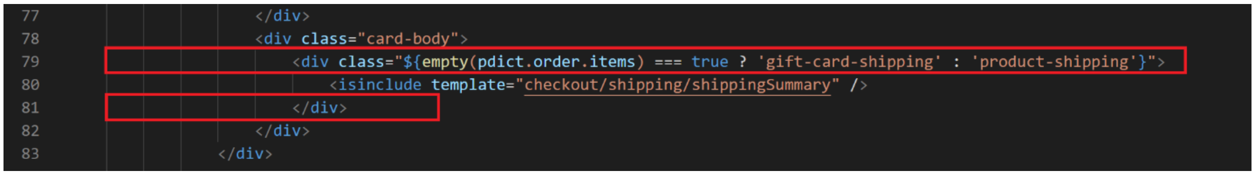
```
<!-- Step 2: Shipping -->
<isif condition="${pdict.order.giftcertificates.length != pdict.order.numItems}">
    <isinclude template="checkout/shipping/shipping" />
</isif>
<div class="card shipping-summary">
```

- Insert the given snippet `<isinclude template="checkout/gc_checkout"/>` by replacing the content below this line `<h2 class="pull-left card-header-custom">${Resource.msg('heading.checkout.shipping', 'checkout', null)}</h2>` in checkout.isml. Refer to the screenshot.



```
<div class="card shipping-summary">
    <div class="card-header clearfix">
        <h2 class="pull-left card-header-custom">${Resource.msg('heading.checkout.shipping', 'checkout', null)}</h2>
        <isinclude template="checkout/gc_checkout"/>

    </div>
```

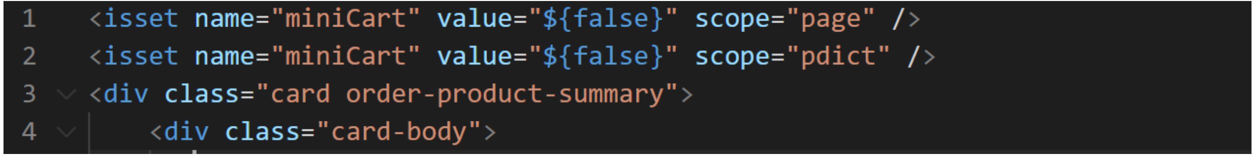
- Insert the given snippet `<div class="${empty(pdict.order.items) === true ? 'gift-card-shipping' : 'product-shipping'}">` above the line of `<isinclude template="checkout/shipping/shippingSummary" />` in checkout.isml. Refer to the screenshot.



```
        </div>
        <div class="card-body">
            <div class="${empty(pdict.order.items) === true ? 'gift-card-shipping' : 'product-shipping'}">
                <isinclude template="checkout/shipping/shippingSummary" />
            </div>
        </div>
    </div>
```

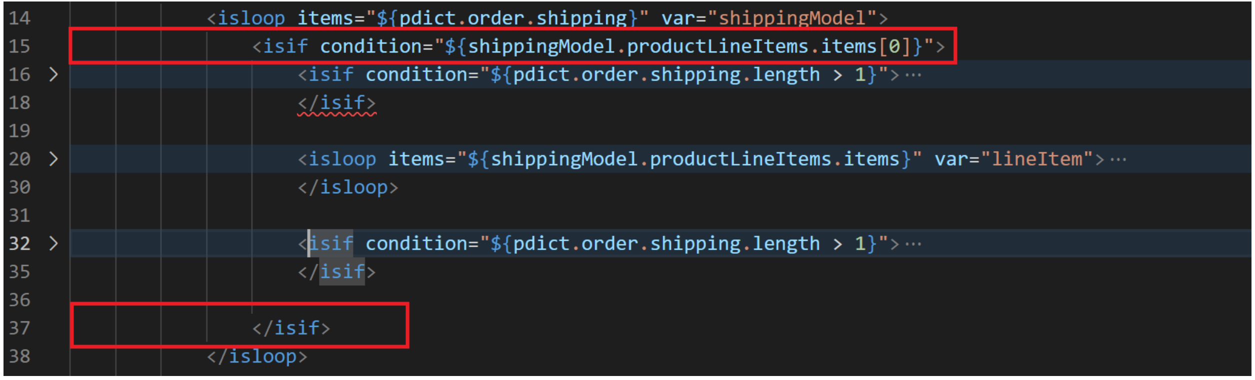
15. Changes for checkout\orderProductSummary.isml

- Insert the given snippet `<isset name="miniCart" value="${false}" scope="pdict" />` in orderProductSummary.isml. Refer to the screenshot.



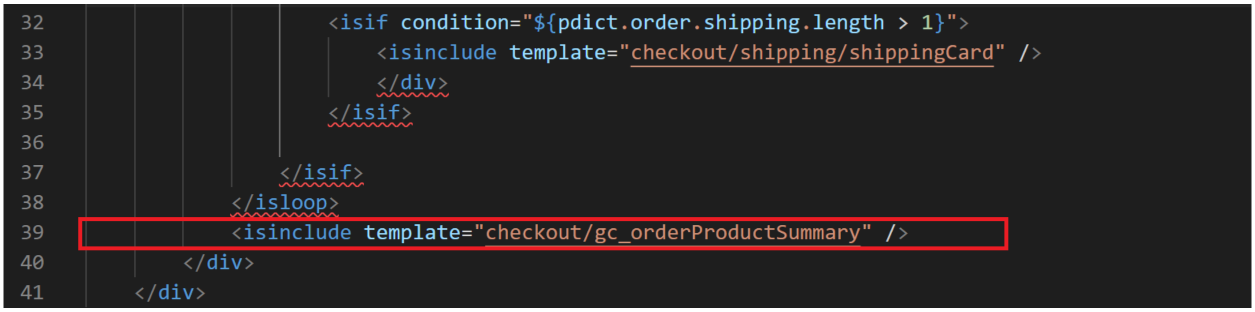
```
<isset name="miniCart" value="${false}" scope="page" />
<isset name="miniCart" value="${false}" scope="pdict" />
<div class="card order-product-summary">
    <div class="card-body">
```

- Insert the given snippet `<isif condition="${shippingModel.productLineItems.items[0]}">` below the line of `<isloop items="${pdict.order.shipping}" var="shippingModel">` in orderProductSummary.isml. Refer to the screenshot.



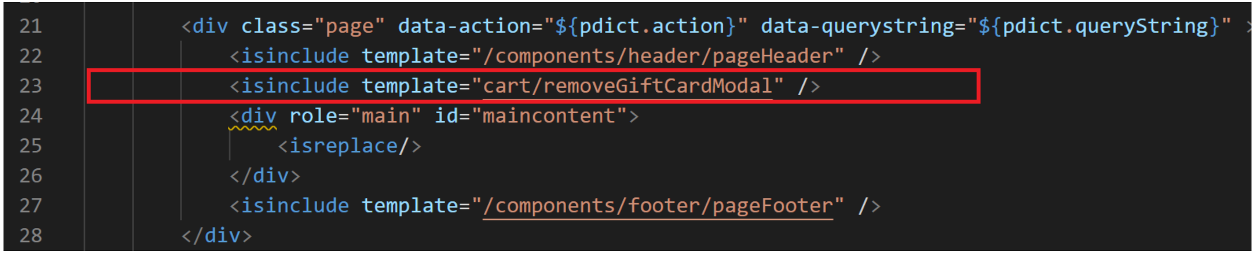
```
<isloop items="${pdict.order.shipping}" var="shippingModel">
    <isif condition="${shippingModel.productLineItems.items[0]}">
        <isif condition="${pdict.order.shipping.length > 1}">…
        </isif>

        <isloop items="${shippingModel.productLineItems.items}" var="lineItem">…
        </isloop>

        <isif condition="${pdict.order.shipping.length > 1}">…
        </isif>

    </isif>
</isloop>
```

- Insert the given snippet `<isinclude template="checkout/gc_orderProductSummary" />` below the isloop. Refer to the screenshot.



```
                <isif condition="${pdict.order.shipping.length > 1}">
                    <isinclude template="checkout/shipping/shippingCard" />
                    </div>
                </isif>

            </isif>
        </isloop>
        <isinclude template="checkout/gc_orderProductSummary" />
    </div>
</div>
```

16. Changes for common\layout\page.isml
    - Insert the given snippet `<isinclude template="cart/removeGiftCardModal" />` below the line of `<isinclude template="/components/header/pageHeader" />` in page.isml. Refer to the screenshot.



```
<div class="page" data-action="${pdict.action}" data-querystring="${pdict.queryString}"
    <isinclude template="/components/header/pageHeader" />
    <isinclude template="cart/removeGiftCardModal" />
    <div role="main" id="maincontent">
        <isreplace/>
    </div>
    <isinclude template="/components/footer/pageFooter" />
</div>
```

External Interfaces

This is not applicable as no external interfaces are being used in e-gift certificate cartridge.

Firewall Requirements

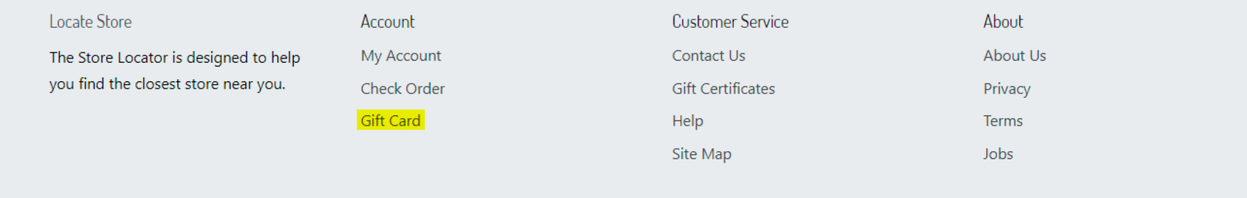
This is not applicable as no firewall requirements are being used in e-gift certificate cartridge.

4. Testing

Listed below are the currently developed use cases and ways to test them:

Gift Card content asset on the storefront

1. Once the Business Manager is all set up, click on the storefront tab on the Business Manager.
2. The storefront opens, navigate to the footer of the Homepage.
3. Check if there is a “Gift Card” option under the “Account” content slot.

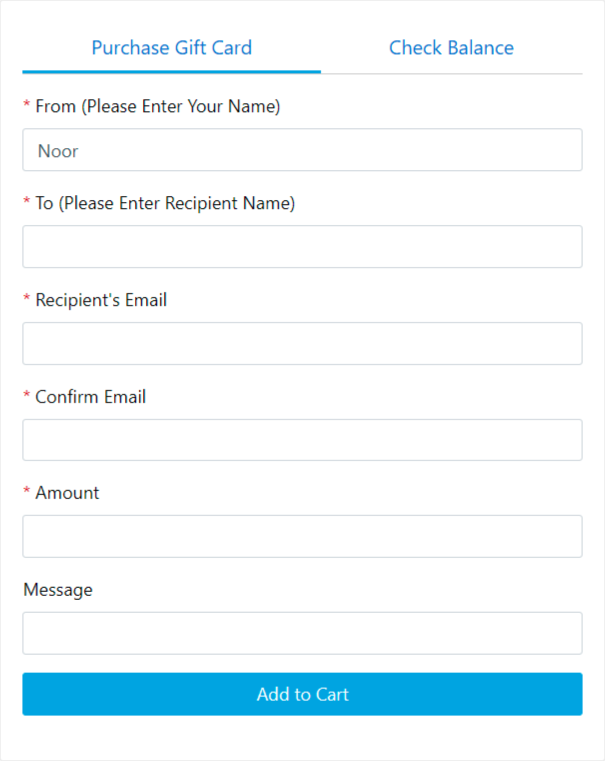


The Store Locator is designed to help you find the closest store near you.

Gift Card purchase form

1. Click the Gift Card option on the storefront footer.
2. The Gift Card opens.
3. You should be able to see the Gift Card purchase form.

4. If you are registered customer, you should be able to see your name already added in the “From” field of the form.



* From (Please Enter Your Name)

Noor

* To (Please Enter Recipient Name)

* Recipient's Email

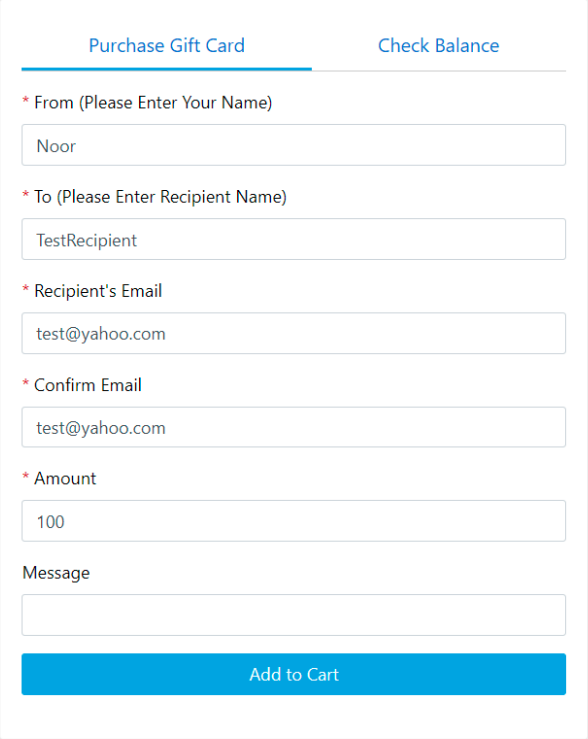
* Confirm Email

* Amount

Message

Add Gift Card to cart

1. Add required information on the form, with appropriate values.



* From (Please Enter Your Name)

Noor

* To (Please Enter Recipient Name)

TestRecipient

* Recipient's Email

test@yahoo.com

* Confirm Email

test@yahoo.com

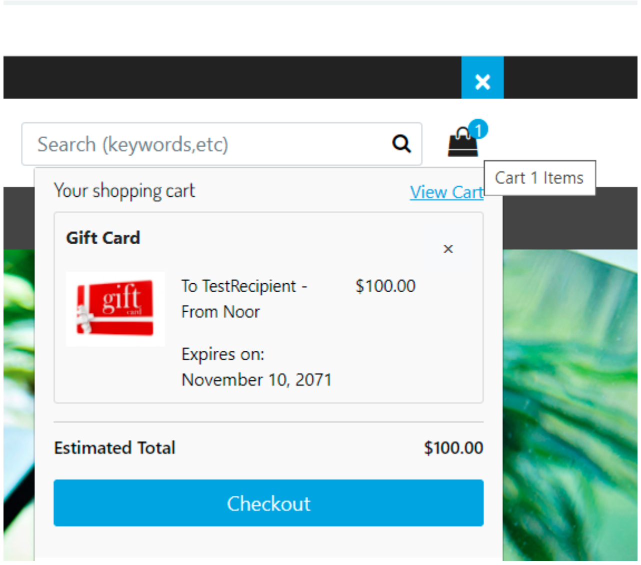
* Amount

100

Message

2. Click “Add to Cart” button present at the end of the form.

3. You should be able to see the Gift Card added to your mini cart, having the same price as the amount you added in the Gift Card form.



Search (keywords,etc)

Cart 1 Items

Your shopping cart View Cart

Gift Card

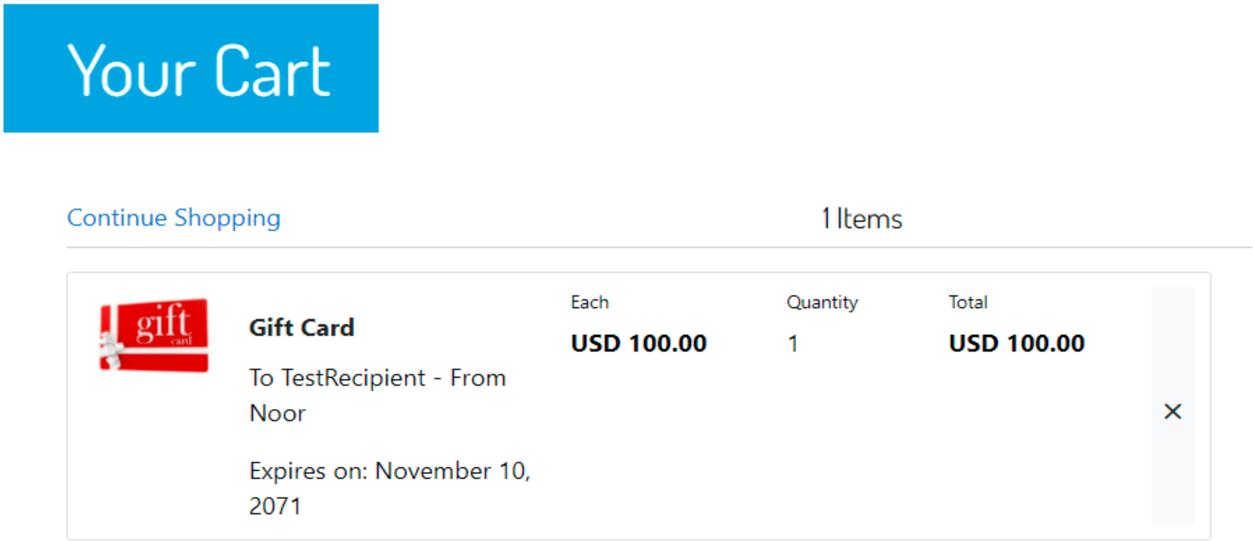
To TestRecipient - From Noor $100.00

Expires on: November 10, 2071

Estimated Total $100.00

Checkout

4. Open your cart page, you should be able to see the gift card there too.



Your Cart

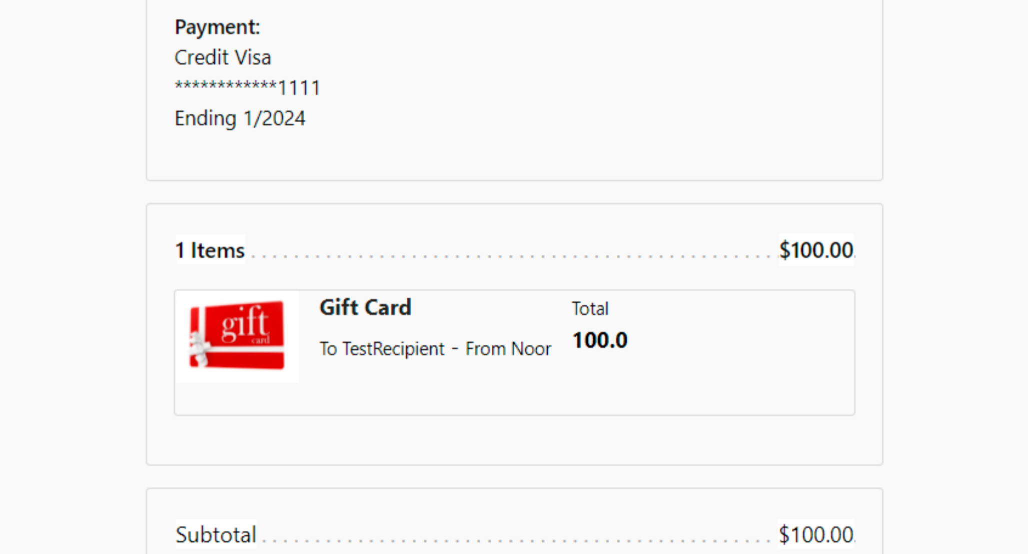
Continue Shopping 1 Items

		Each	Quantity	Total
	Gift Card To TestRecipient - From Noor Expires on: November 10, 2071	**USD 100.00**	1	**USD 100.00**

5. The form should be cleared.

Purchase Gift Card

1. On the cart page, click “Checkout” button.
2. Fill the necessary information and click “Place Order”
3. On the “Thank you” page, see the gift card tile.



Payment:
Credit Visa
************1111
Ending 1/2024

1 Items	**$100.00**

Gift Card
To TestRecipient - From Noor

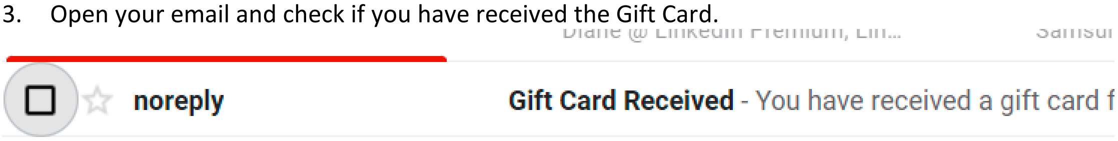
Total
100.0

Subtotal	$100.00

4. Open your email, see the order confirmation email.
5. This email should contain the Order details, with the Gift Card.

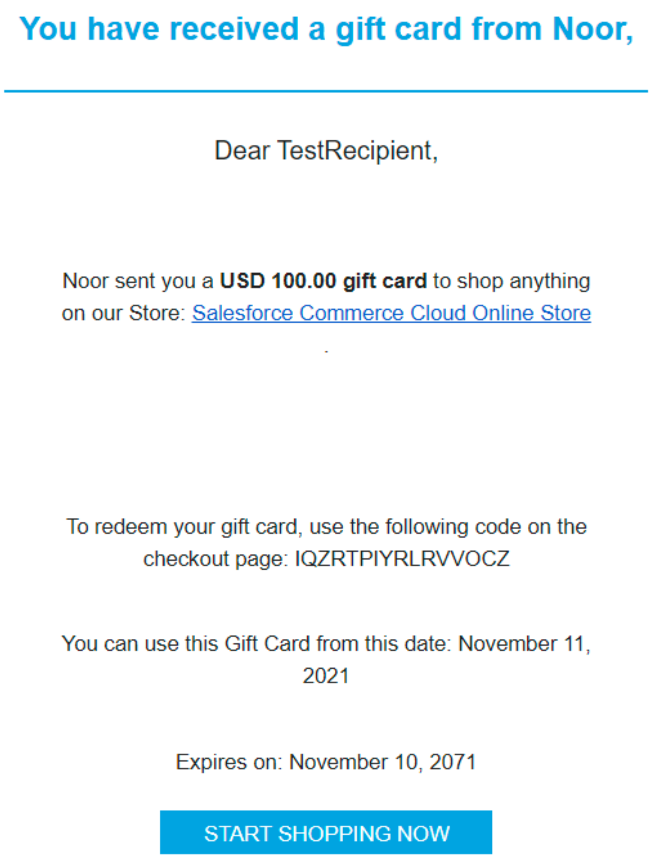
Gift Card Received

1. Follow steps above and put your own email in the “Recipients Email” field in the form.
2. Complete order as shown above.
3. Open your email and check if you have received the Gift Card.



noreply **Gift Card Received** - You have received a gift card f

4. You should get an email like this, having Gift Card code and other information:



You have received a gift card from Noor,

Dear TestRecipient,

Noor sent you a **USD 100.00 gift card** to shop anything on our Store: Salesforce Commerce Cloud Online Store

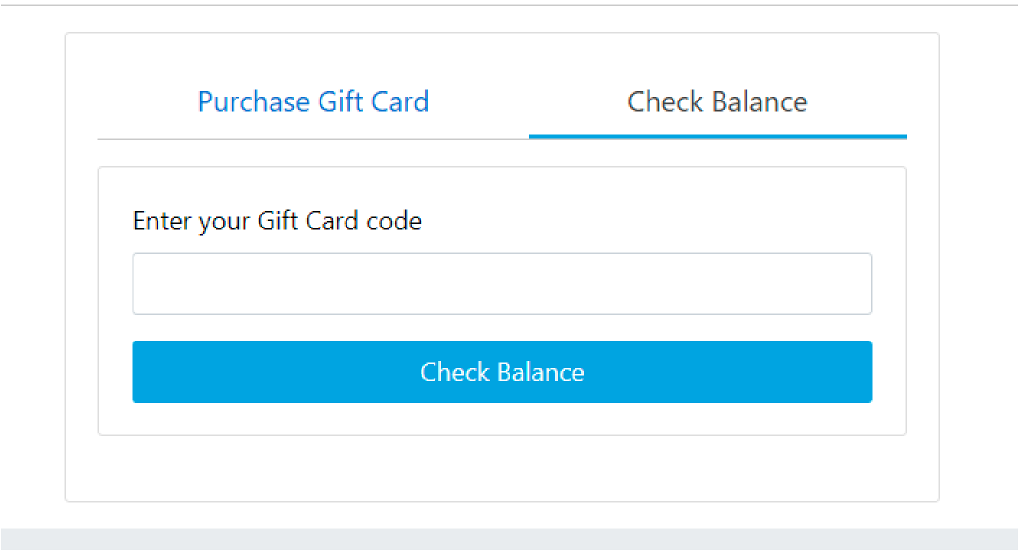
To redeem your gift card, use the following code on the checkout page: IQZRTPIYRLRVVOCZ

You can use this Gift Card from this date: November 11, 2021

Expires on: November 10, 2071

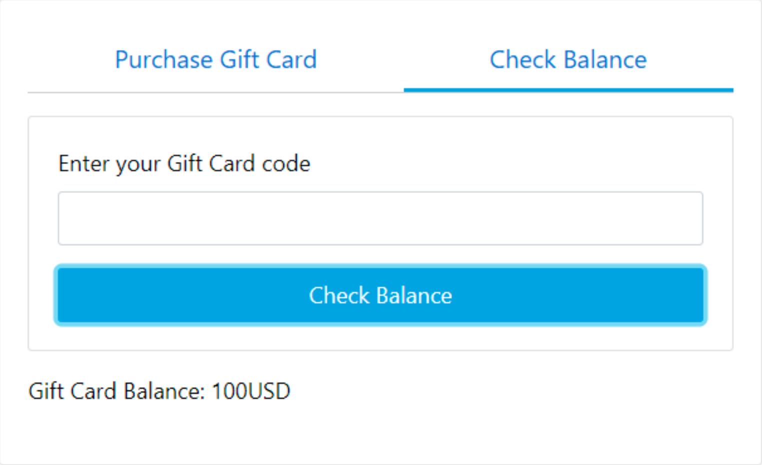
Check Balance

1. Navigate back to the “Gift Card” page on the storefront.
2. Click on the “Check Balance” tab



3. Enter the code from the Gift Card that you sent to yourself in the above test.
4. Click the “Check Balance” button.

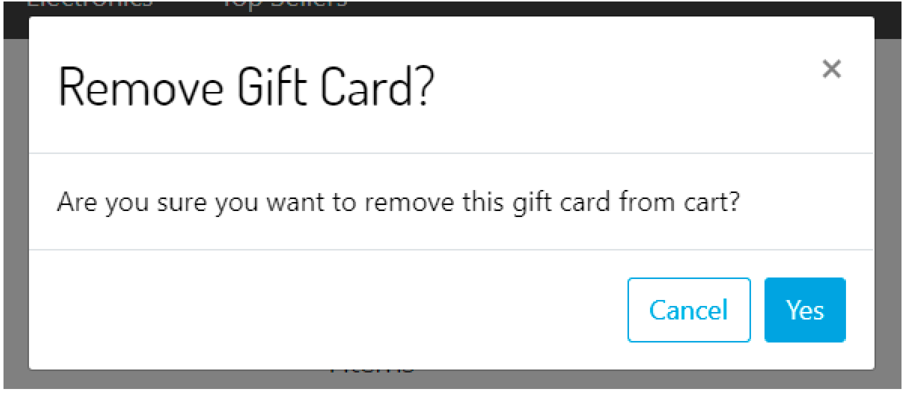
5. You should be able too the remaining balance displayed below the button and the form is cleared.



Gift Card Balance: 100USD

Remove Gift Card

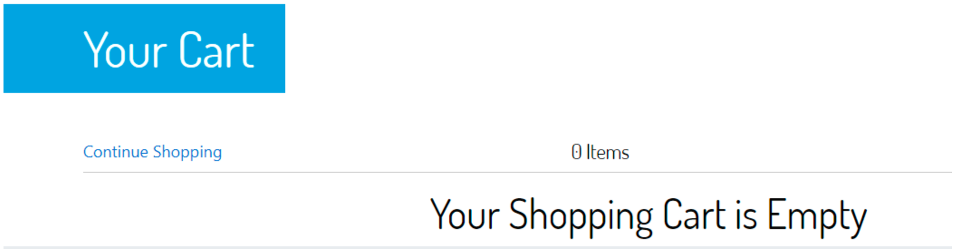
1. Navigate to the “Gift Card” page.
2. Fill the form and add the gift card to your cart, as shown in the tests above.
3. Once the gift card is added to card, open the “Cart” page.
4. Click on the “X” button on the Gift Card tile.
5. A pop-up should be displayed like this:



Remove Gift Card?

Are you sure you want to remove this gift card from cart?

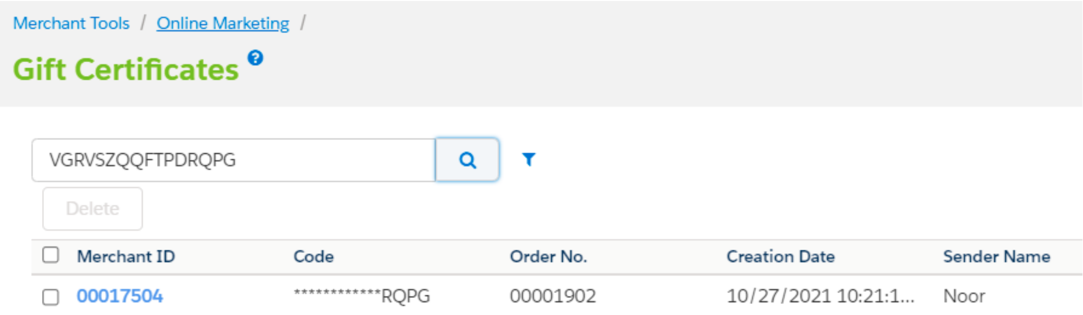
6. Click “Yes”.
7. Gift Cart should be removed from your cart.



Your Shopping Cart is Empty

Gift Certificate Created in the Business Manager

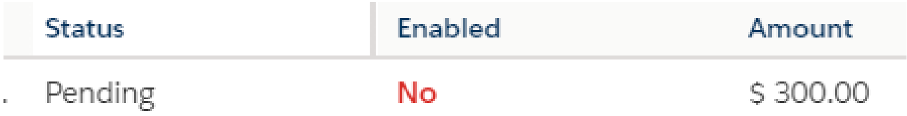
1. Open the Business Manager and click on your site.
2. Go to Merchant Tools > Online Marketing > Gift Certificates.
3. Here you will see the list of all the Gift Certificates previously created.
4. Enter the code of the Gift Certificate that you previously purchased in the search field.
5. Click search
6. The Gift Certificate should be displayed.



VGRVSZQQFTPDRQPG

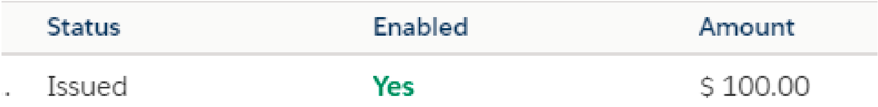
☐ Merchant ID	Code	Order No.	Creation Date	Sender Name
☐ 00017504	************RQPG	00001902	10/27/2021 10:21:1...	Noor

7. This Gift Certificate should have Status "Issued" and Enabled "No" set. This is because the gift certificate is enabled later on after the Job "enableGiftCertificate" is run.



Status	Enabled	Amount
Pending	No	$ 300.00

Only once the Job is run and the "enableDate" for the Gift Certificate is here, notice that the Gift Certificate has Status "Issued" and Enabled "Yes".

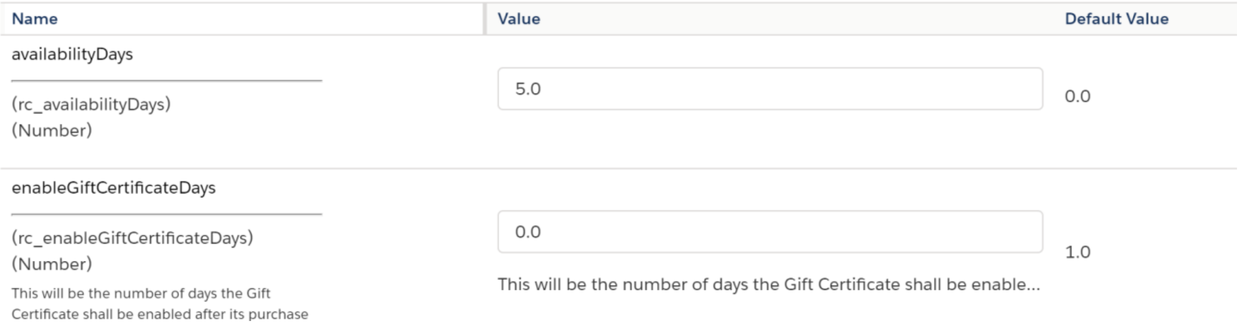


Status	Enabled	Amount
Issued	Yes	$ 100.00

Gift Certificate Enable

In the previous test, we discussed that the gift certificates are enabled once the job is run. In this test, we will run the job and test the functionality.

1. In the Business Manager, go to Merchant Tools> Site Preferences> Custom Preferences.
2. Navigate to the Preference group with the ID "rc_giftCertificateAvailabilityDays" and select it.
3. Set the value for "rc_enableGiftCertificateDays" as 0 and click "Save".



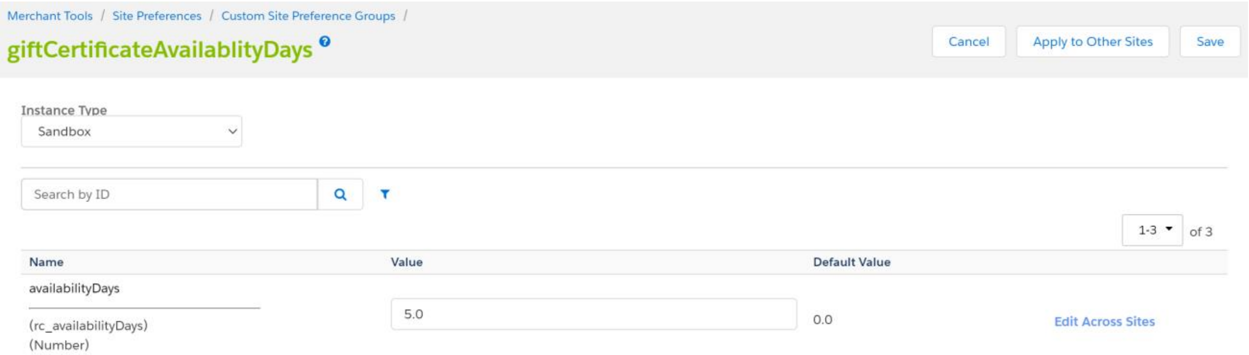
Name	Value	Default Value
availabilityDays (rc_availabilityDays) (Number)	5.0	0.0
enableGiftCertificateDays (rc_enableGiftCertificateDays) (Number) This will be the number of days the Gift Certificate shall be enabled after its purchase	0.0 This will be the number of days the Gift Certificate shall be enable...	1.0

4. Now purchase a gift card, you will see in the email, it will have the same day in the "You can use this gift certificate from" as the day you are purchasing it.
5. In the Business Manager, open Administration> Operations> Jobs.
6. Open the "rc_EnableGiftCertificates" job and click "Run Now".
7. Now, go to Merchant Tools > Online Marketing > Gift Certificates.
8. Search your gift certificate by searching its Code.
9. You will notice it is now Enabled.

Gift Certificate Expiry Reminder Email

1. In the Business Manager, go to Merchant Tools> Site Preferences> Custom Preferences.

2. Navigate to the Preference group with the ID “giftCertificateAvailabilityDays” and select it.
3. Set the value for “availabilityDays” as any value from 1-5 and click “Save”.

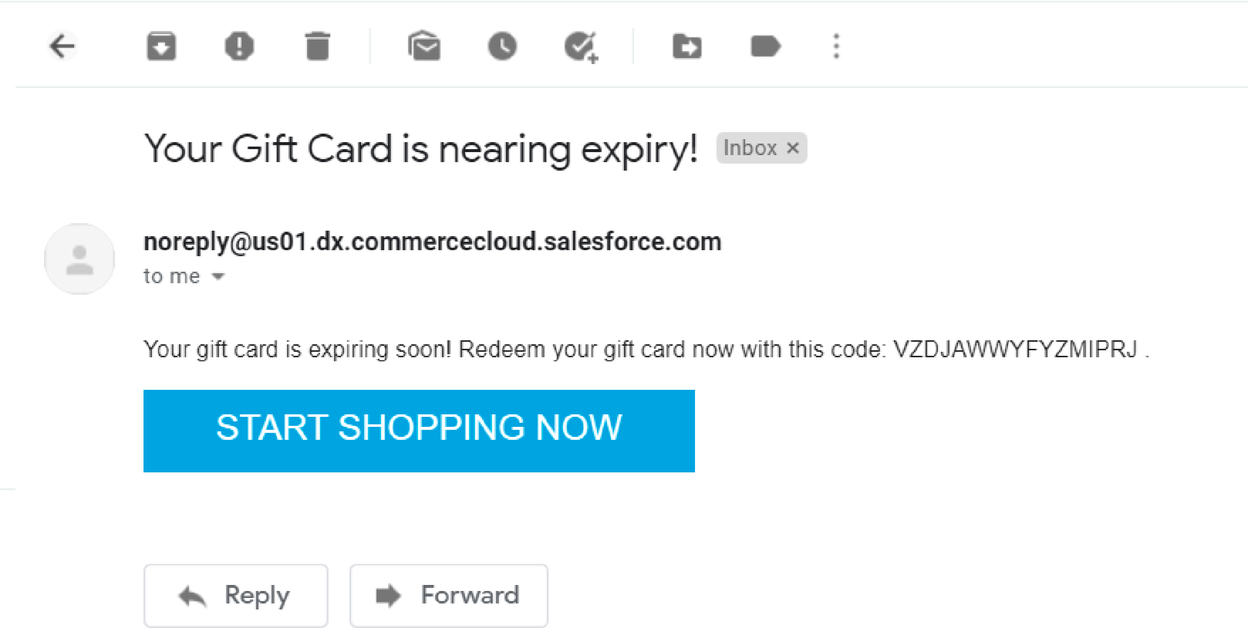

Merchant Tools / Site Preferences / Custom Site Preference Groups /
Cancel
Apply to Other Sites
Save
Search by ID
1-3 of 3
Edit Across Sites

giftCertificateAvailablityDays

Instance Type: Sandbox

Name	Value	Default Value
availabilityDays (rc_availabilityDays) (Number)	5.0	0.0

4. Now open the storefront and purchase a gift card, you will notice that it will have expiry in 3 days’ time.
5. In the Business Manager, open Administration> Operations> Jobs.
6. Open the “SendGCExpiryReminderEmail” job and click “Run Now”
7. No open you Recipient email.
8. You will notice the Expiry Reminder email has been sent.



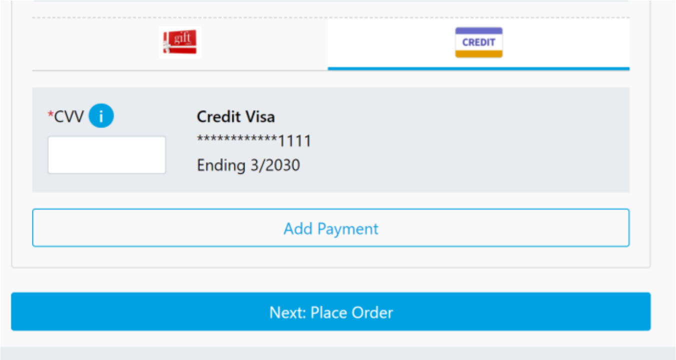
Your Gift Card is nearing expiry!

noreply@us01.dx.commercecloud.salesforce.com

Your gift card is expiring soon! Redeem your gift card now with this code: VZDJAWWYFYZMIPRJ .

Gift Certificate Redeem

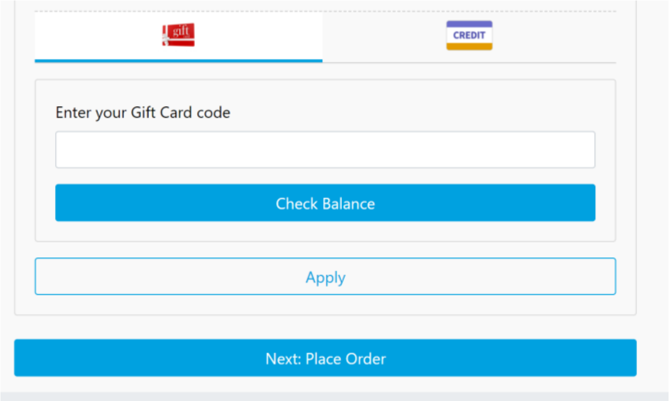
1. Add any product except Gift Certificate (cannot pay for Gift Certificates with Gift Certificates) in the cart.
2. Move to checkout page and fill billing information
3. In Payment Section, you can see Gift Certificate as a payment method with other integrated methods.



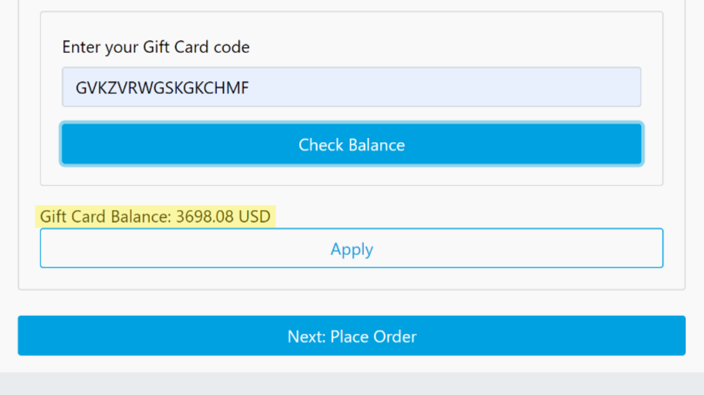
*CVV

Credit Visa
************1111
Ending 3/2030

4. Click on Gift certificate tab.

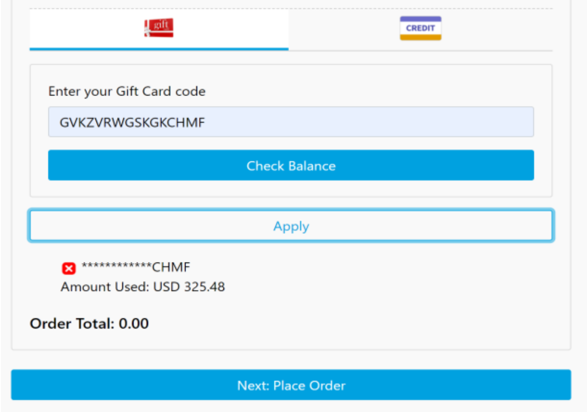


5. Enter your gift certificate code.
6. Click on Check Balance button if you want to check balance of gift certificate.


Enter your Gift Card code
GVKZVRWGSKGKCHMF

Gift Card Balance: 3698.08 USD

7. Click on Apply button to redeem your gift certificate amount.

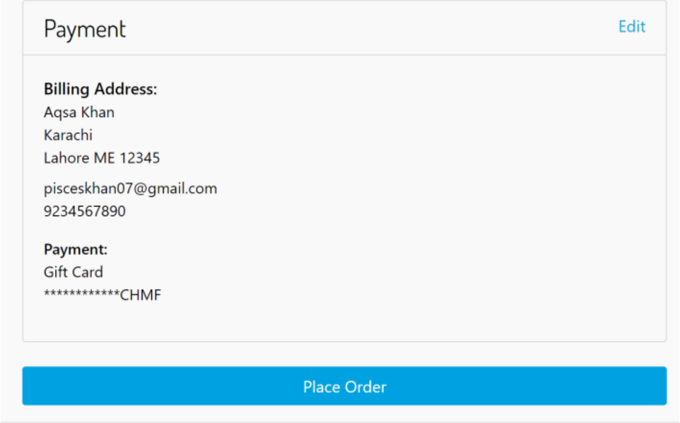


Enter your Gift Card code
GVKZVRWGSKGKCHMF

☒ ************CHMF
Amount Used: USD 325.48

Order Total: 0.00

8. If you don’t want to redeem amount you can click on close button and delete the redemption.
9. Now place your order.

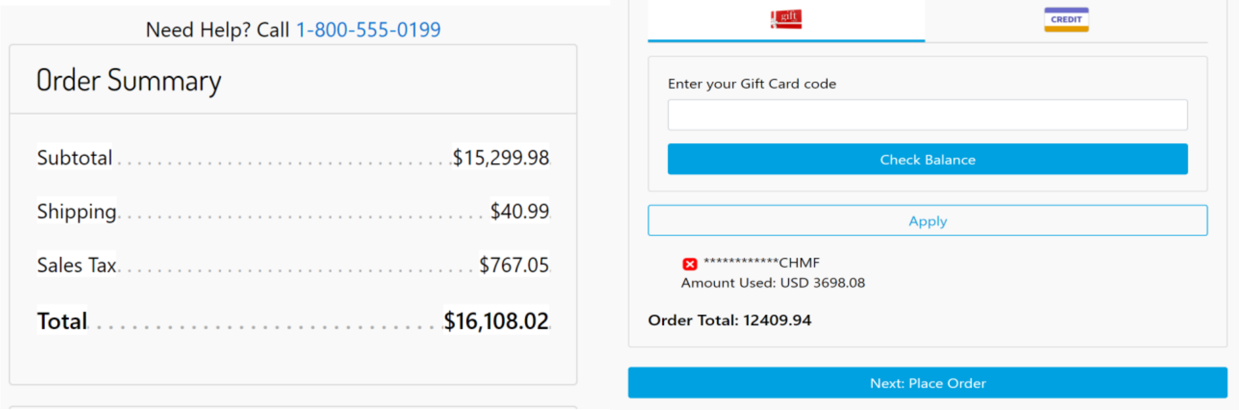


Billing Address:
Aqsa Khan
Karachi
Lahore ME 12345

pisceskhan07@gmail.com
9234567890

Payment:
Gift Card
************CHMF

10. If your order total amount is more than the gift certificate amount you can use gift certificate method and remaining amount can be paid from other payment methods.



Need Help? Call 1-800-555-0199

Order Summary

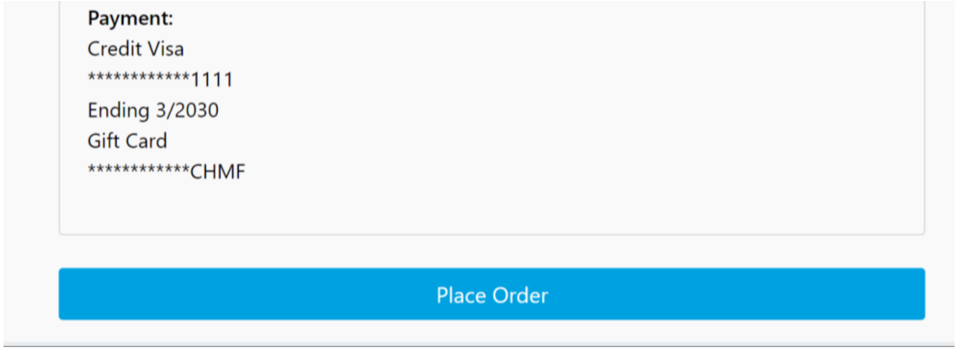
Subtotal	$15,299.98
Shipping	$40.99
Sales Tax	$767.05
Total	$16,108.02

Enter your Gift Card code

☒ ************CHMF
Amount Used: USD 3698.08

Order Total: 12409.94

11. Both payment methods will be shown on order page.

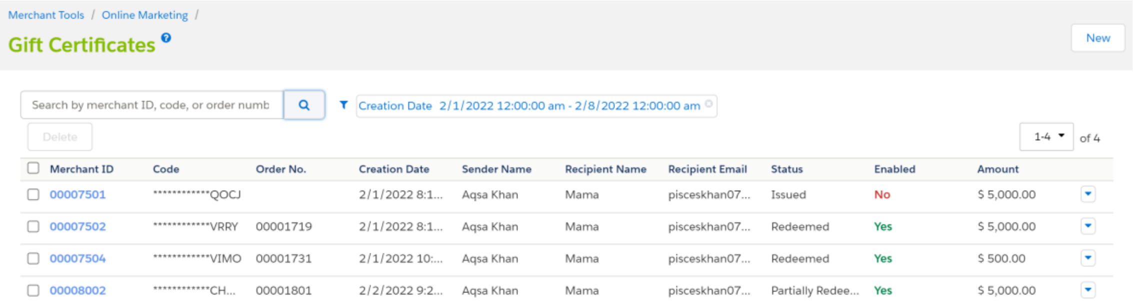


Payment:
Credit Visa
************1111
Ending 3/2030
Gift Card
************CHMF

12. Then click on Place Order.

Gift Certificate Status

1. Merchant Tools < Online Marketing < Gift Certificates

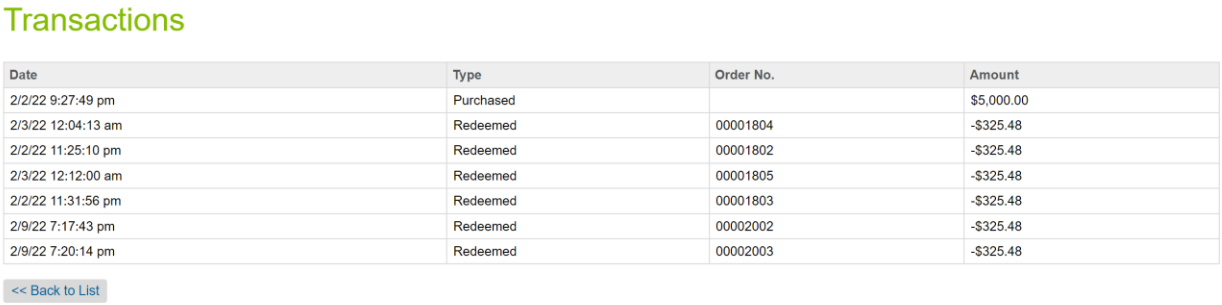

Merchant Tools / Online Marketing /

Gift Certificates

New
Search by merchant ID, code, or order numb
Creation Date 2/1/2022 12:00:00 am - 2/8/2022 12:00:00 am
Delete
1-4
of 4

□	Merchant ID	Code	Order No.	Creation Date	Sender Name	Recipient Name	Recipient Email	Status	Enabled	Amount
□	00007501	************QOCJ		2/1/2022 8:1...	Aqsa Khan	Mama	pisceskhan07...	Issued	No	$ 5,000.00
□	00007502	************VRRY	00001719	2/1/2022 8:1...	Aqsa Khan	Mama	pisceskhan07...	Redeemed	Yes	$ 5,000.00
□	00007504	************VIMO	00001731	2/1/2022 10:...	Aqsa Khan	Mama	pisceskhan07...	Redeemed	Yes	$ 500.00
□	00008002	************CH...	00001801	2/2/2022 9:2...	Aqsa Khan	Mama	pisceskhan07...	Partially Redee...	Yes	$ 5,000.00

2. Enter the gift certificate code.
3. Click on your gift certificate.
4. Merchant can see transaction record along with order number of gift certificate amount.


Transactions

Date	Type	Order No.	Amount
2/2/22 9:27:49 pm	Purchased		$5,000.00
2/3/22 12:04:13 am	Redeemed	00001804	-$325.48
2/2/22 11:25:10 pm	Redeemed	00001802	-$325.48
2/3/22 12:12:00 am	Redeemed	00001805	-$325.48
2/2/22 11:31:56 pm	Redeemed	00001803	-$325.48
2/9/22 7:17:43 pm	Redeemed	00002002	-$325.48
2/9/22 7:20:14 pm	Redeemed	00002003	-$325.48

5. Operations, Maintenance

Data Storage

A custom object named “rc_GiftCertificateObj” is stored in the Business Manager. It is used to send expiry notification to customers when e-gift certificate is not being used after purchasing. It has the following attributes:

Attribute ID	Attribute Name	Description
rc_GiftCertificateCode	String - key attribute	Unique string code for gift certificate
rc_RecipientEmail:	Email	Recipient’s Email ID for sending email
rc_ExpirationDate:	Date	Expiration Date of the e-gift certificate
rc_customerID	String	Customer number of the customer whose gift card is added
rc_ enableDate	Date	The date in which gift card is enabled

The “rc_SendGCExpiryReminderEmail” job also sends the expiry date of the e-gift certificate to the customer.

Deletion of Custom Object

When the “rc_EnableGiftCertificates” job runs, it checks two conditions to delete the custom object from memory. If the gift card is fully redeemed or its expiration date is passed, the custom object will be deleted.

Availability

Since e-gift certificate cartridge is not an integration cartridge. GiftCertificate is stored within SFCC in GiftCertificateLineItem system object. The availability of data is only dependent on SFCC.

Failover/Recovery Process

This is not applicable as e-gift certificate cartridge perform data CRUD operations on SFCC system object GiftCertificateLineItem and data is persisted within SFCC.

Support

Please check the GitHub repo link

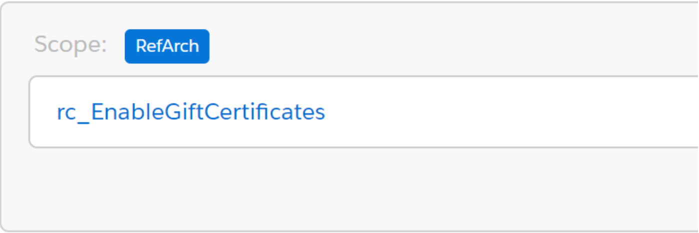
6. User Guide

Roles, Responsibilities

A Business manager administrator role is required to install the cartridge and import the metadata on SFCC instance.

Business Manager

1. The “rc_EnableGiftCertificates” job has site scope and contains the following one custom step:

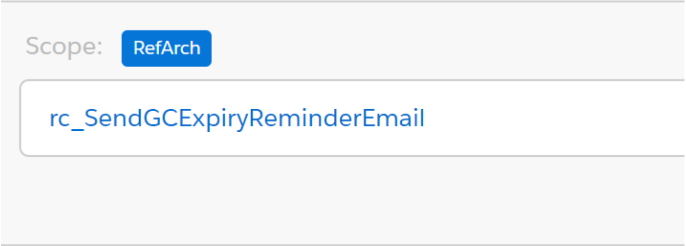


Scope: RefArch

rc_EnableGiftCertificates

This step enables the e-gift certificate for the customer’s use.

2. The “rc_SendGCExpiryReminderEmail” job has site scope and contains the following one custom step:



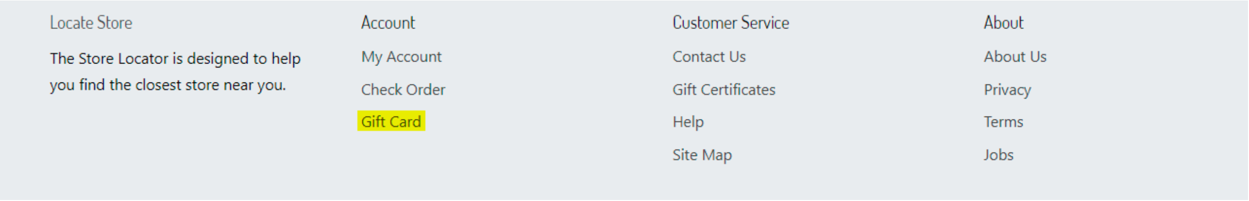
Scope: RefArch

rc_SendGCExpiryReminderEmail

This step sends the e-gift certificate expiry reminder email to the customer to use the gift certificate as soon as possible before expiration.

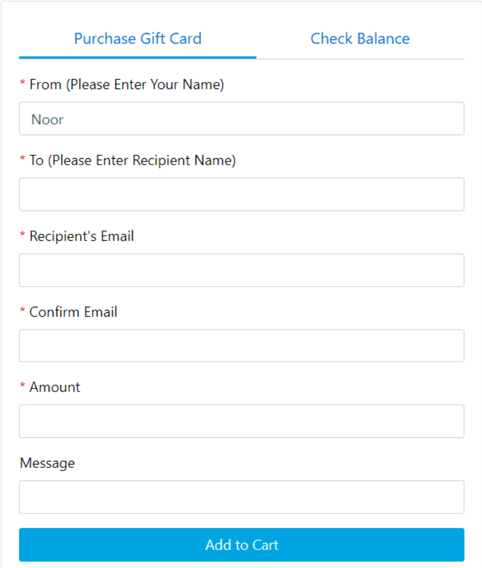
Storefront Functionality

1. Visit the site and check the footer for the “Gift Card” option under the “Account” heading.



Locate Store	Account	Customer Service	About
The Store Locator is designed to help you find the closest store near you.	My Account	Contact Us	About Us
	Check Order	Gift Certificates	Privacy
	Gift Card	Help	Terms
		Site Map	Jobs

2. Open the Gift Card page and see the tabs to either Purchase a Gift Card or Check Balance.



* From (Please Enter Your Name)

Noor

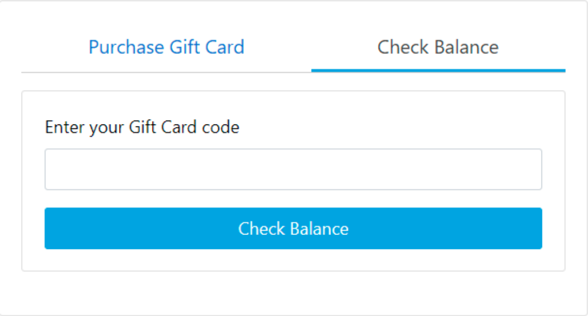
* To (Please Enter Recipient Name)

* Recipient's Email

* Confirm Email

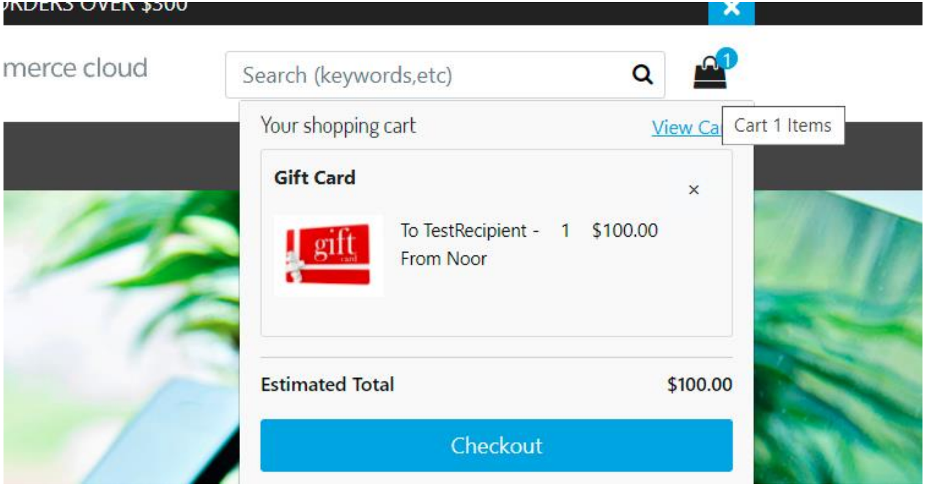
* Amount

Message



Enter your Gift Card code

3. Add the Gift Card to your cart by filling the information in the form under the “Purchase Gift Card” tab.
4. The Gift Card is automatically added to your cart.



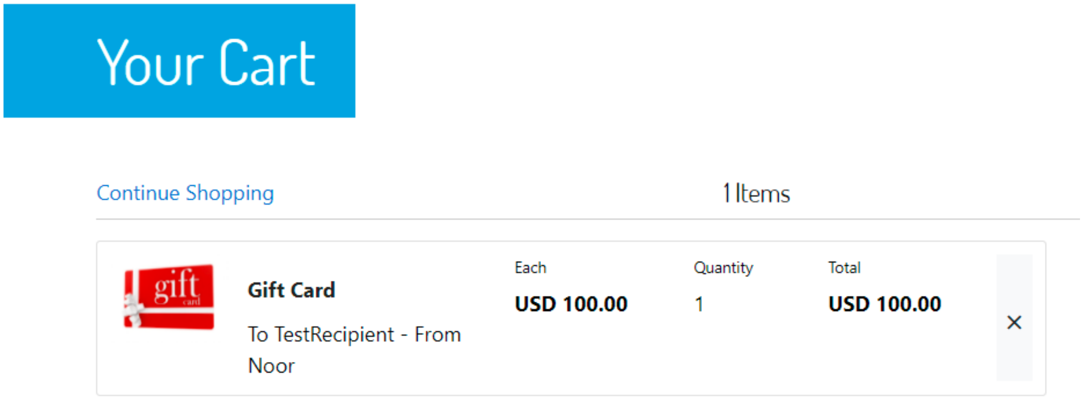
Your shopping cart

Gift Card

To TestRecipient - From Noor 1 $100.00

Estimated Total $100.00

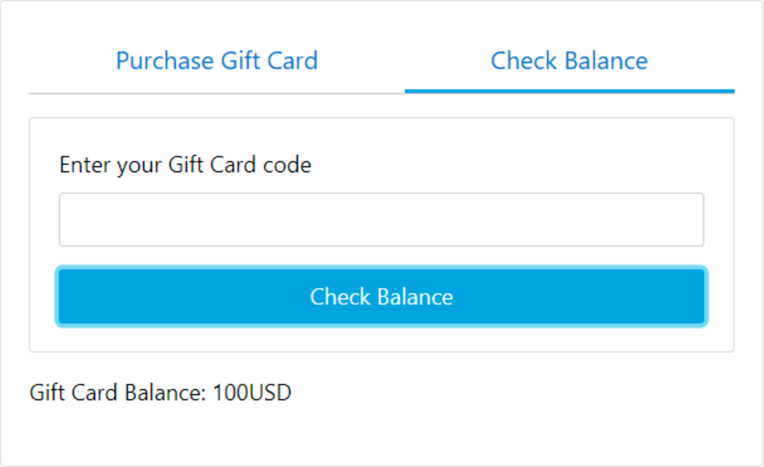
5. Open the Cart page and see the Gift Card tile, it should be similar to that of any other product.



	Each	Quantity	Total
Gift Card To TestRecipient - From Noor	USD 100.00	1	USD 100.00

6. You can remove this Gift Card from your cart as any other product.

7. Complete the order by the usual check out flow.
8. Check the order confirmation email and the Gift Card Recipient email.
9. You can check the balance on your Gift Card by entering the Gift Card code in the field on the form under the “Check Balance” tab on the Gift Card page.



Gift Card Balance: 100USD

7. Known Issues

No known issues.

8. Release History

Version	Date	Changes
22.1.0	2022-06-10	Initial release