

1. Home	2
1.1 Adyen Integration	2
1.2 AfterPay Integration	9
1.3 Apple Pay Created accounts and Apple pay documents	11
1.4 Avalara Sales Tax Integration	11
1.5 Braintree Credit Card Integration	13
1.6 Brands page for Sunburn site	21
1.7 CSC - Customer Service Center	23
1.8 Custom Preferences	28
1.9 Custom rendering template with 6 content slots	30
1.10 Email module Seafolly/Sunburn	31
1.11 Error / Not Found[404]/ Forbidden/Offline Pages	34
1.12 Fonts	36
1.13 Getting started	39
1.13.1 Making a template	40
1.14 Gift vouchers and Gift cards	40
1.14.1 Free Gift Voucher	40
1.14.2 Gift voucher purchase page	42
1.14.3 Prices for e-gift vouchers	43
1.14.4 Single page checkout using a Gift Voucher	43
1.15 Google Tag Manager (GTM)	45
1.16 Header and Footer	46
1.17 Mailing list subscription	47
1.18 Mini-Bag and Basket page	50
1.19 My Account	51
1.19.1 Customer migration - reset password flow	51
1.19.2 My Account Pages	54
1.20 Non-transactional site support - Maaji	55
1.21 Order Status Updates	56
1.22 Password Reset	63
1.23 PayPal Integration	63
1.24 PCA Predict Integration	68
1.25 PDP - product details page and Product Set page	72
1.26 PLP - Product Listing Page	73
1.26.1 Color Refinements Configuration	73
1.26.2 Images swap on hover	73
1.27 Product carousels	74
1.28 Project general info	75
1.29 Release 1	76
1.30 Sass, CSS, JS, Sprites, Grunt	76
1.30.1 Using common variables, styles, and mixins for Seafolly / Sunburn	77
1.31 Seafolly Integrations	86
1.31.1 Customers Export	87
1.31.2 Customers Import	95
1.31.3 Gift Vouchers Import	96
1.31.4 Newsletter Export	101
1.31.5 Stores Import	107
1.31.6 Transfer To FTP	107
1.32 Search Details	108
1.33 Single Page Checkout	109
1.33.1 Address Fields Constraints	109
1.33.2 Shipping Countries	110
1.33.3 Shipping Methods Preferences	110
1.33.4 Tax Preferences	111
1.34 Site Switcher	112
1.35 Storefront Homepage[Seafolly/Sunburn]	112
1.36 Storefront Inline Editor (SIE) integration	114
1.37 Store locator	117
1.38 TradeState Integration	118

Home

Welcome to your new documentation space!

This is the home page for your documentation space within Confluence. Documentation spaces are great for keeping technical documentation organised and up to date.

Next you might want to:

- Customise the home page** - Click "Edit" to start editing your home page
- Check out our sample pages** - Browse the sample pages in the sidebar for layout ideas
- Create additional pages** - Click "Create" and choose "Blank Page" to get started
- Manage permissions** - Click "Space Tools" and select "Permissions" in the sidebar to manage what users see

Search this documentation

- documentation-space-sample
- featured

Featured Pages

- Getting started
- Making a template

Popular Topics

Recently Updated Pages

-  **Free Gift Voucher**
22 minutes ago • updated by Iliyan a Mihaylova • view change
-  **Free Gift Voucher**
May 09, 2018 • updated by Liliya Yocheva • view change
-  **Free Gift Voucher**
May 07, 2018 • updated by aneesh • view change
-  **Adyen Integration**
May 03, 2018 • updated by Iliyana Mihaylova • view change
-  **Storefront Inline Editor (SIE) integration**
May 03, 2018 • updated by Max Yozoff • view change

Adyen Integration

The integrated version of the Adyen cartridge is 17.1.0. according the documentation:

[Adyen Integration Guide-v17.1.0.pdf](#)

Sandbox Account

Adyen test customer area can be accessed here:

Login URL	Account	Username	Password
https://ca-test.adyen.com/	MaajiSwimwear	admin	Maaji2018*

User Accounts

In order for the BM services to work a web service user account must be created. To do that go to **Settings > Users > Add new user** with the following details: ([users-list.png](#))

- User type – Web service
- Under Roles & Accounts select all available options

Username

Password	7c&ge?pR(C5!6Jy<-5UXw+&@U
Status	Active
Encryption Key	10001 BCF3A67A589342DC4E2BEFCA3A86B73826489359E50F28F9A91884CD3CCB6AA98600E225A6A237EE359549A622EA
CSE Library Location	https://test.adyen.com/hpp/cse/js/8215205960383370.shtml

Payment Methods

There are different payment methods that are available for the merchant account. They can be found under **Settings > Payment Methods**. ***These methods are enabled by the Adyen team.***

PayPal

The PayPal payment method is not enabled by default and some configurations are required on our side and by the Adyen team. The process is described below.

1. Configure PayPal sandbox
 - a. Log into <https://developer.paypal.com/> and create new business account (**Sandbox > Accounts > Create Account**)
 - b. Login into <https://sandbox.paypal.com/> with the new account
 - c. Add new **Third Party Permissions (Profile > Selling Tools > API Access > Grant API Permissions)** with username **sell1_1287491142_biz_api1.adyen.co** and click "Lookup"
 - d. Grant these permissions:
 - i. Use Express Checkout to process payments.
 - ii. Issue a refund for a specific transaction.
 - iii. Process your shopper's credit or debit card payments.
 - iv. Authorize and capture your PayPal transactions.
 - v. Obtain information about a single transaction.
 - vi. Obtain authorization for pre-approved payments and initiate pre-approved transactions.
 - vii. Generate consolidated reports for all accounts. (In some countries, you might get an error message informing you that the feature is not available in the country where you have your PayPal account. In this case, uncheck/deselect this option.)
 - viii. Use Express Checkout to process mobile payments. (If you do not support or plan to support mobile payments, leave this option unchecked.)
2. ***Contact Adyen support to link the test business account to the testing Adyen account***

PayPal Developer Account

In order to integrate the PayPal functionality, a developer account must be created. You can create a new business account here <https://www.paypal.com/us/webapps/mpp/account-selection> or use the following credentials:

Login URL	Email	Password
https://developer.paypal.com/	antonia.dimitrova@tryzens.com	P99v2EYCne?Prcl

The PayPal Sandbox credentials are:

Login URL	Email	Password	Type
https://www.sandbox.paypal.com	proj.maaji.co@gmail.com	cra2yPuma65	Business
	proj.seafolly.buyer@tryzens.com proj.seafolly.buyer3@tryzens.com	cra2yPuma65	Personal

Gmail Test Account

We also have created a Gmail email that is used as a sandbox login above with the following data:

Login URL	Email	Password

For information about PayPal by Adyen refer to:

- <https://docs.adyen.com/developers/payment-methods/paypal>
- <https://docs.adyen.com/developers/payment-methods/paypal/sandbox-account-setup#setup>

Server Communication & Notifications

The server communication settings are under **Settings > Server Communication**. There we must add new “**Standard Notification**” that will send data to the Business Manager to handle the PayPal payments.

URL	https://dev03-ap01-seafolly.demandware.net/on/demandware.store/Sites-MaajiCO-Site/es_CO/Adyen-Notify or similar sandbox/dev/staging/prod instance url with Adyen-Notify action, i.e: https://development-ap01-seafolly.demandware.net/on/demandware.store/Sites-MaajiCO-Site/es_CO/Adyen-Notify https://staging-ap01-seafolly.demandware.net/on/demandware.store/Sites-MaajiCO-Site/es_CO/Adyen-Notify https://production-ap01-seafolly.demandware.net/on/demandware.store/Sites-MaajiCO-Site/es_CO/Adyen-Notify
SSL Version	SSL (insecure cyphers)
Accept expired	
Accept self-signed	true
Accept untrusted Root Certificates	true
Method	HTTP POST
Active	true
User	seafolly
Password	London@2020!

Additional Response Data

The Adyen cartridge requires some additional data to be sent with the notifications. This can be configured in **Settings > API & Response**. ([api-urls.png](#) & [api-data.png](#))

Here we can find the API URLs that we'll use for the services in the Business Manager. Scroll down to “**Additional Data in API Response**” and enable the following:

- 3D Secure
 - Raw 3D Secure details result
 - Raw 3D Secure result
 - 3D Secure result
- Risk
 - Fraud Result
- Payment
 - Authorisation amount
 - Authorisation amount (dynamic zero authorisation)
 - Recurring details

Skins

The Adyen payment skins can be created and configured under **Skins** in the left menu ([skins-list.png](#)). There is currently one configured skin [OcCPQyiv \(skin-options.png\)](#)

Skin Code: OcCPQyiv	
HMAC Key	B822E2E20DE6633EC1AB0A01A0F4F46485768FF22478AE1C724068F29B3D5DF5
Skin Options	
	Use inline frame for VbV and MSC 3-D Secure interaction: true
	Capture delay: immediate
	Support partial payments: true
	Support device fingerprint: true
	Break out of frame: same window

Skins have other options like payment methods & custom fields than can also be configured. For this skin we have enabled the following payment methods:

- Visa
- MasterCard
- American Express
- PayPal – for configuration refer to the Payment Methods section above

Business Manager Settings

Cartridge Path

Add the following to the cartridge path of the site :int_adyen_controllers:int_adyen after app_seafolly_core

System and Custom Objects

The following System and Custom object extension definitions must be imported the the BM:

- [system-objects.xml](#) - CustomerPaymentInstrument, Order, PaymentTransaction, SitePreferences
- [custom-objects.xml](#) - adyenNotification

System preferences & Services, Payment methods & Processors

Import the following files into the BM in the appropriate sections:

- [payment-processors.xml](#)
- [payment-methods.xml](#)
- [services.xml](#)
- [jobs.xml](#)

Services, Profiles and Credentials

The Adyen services can be configured in Administration/Operations/Services. We have 4 services with the following configurations. The site name ID suffix can be changed for every site that you must have the Adyen services in use.

Name	Type	Enabled	Service Mode	Log Name Prefix *	Communication log enabled	Profile	Credentials
adyen.http.payment.3dsecure. MaajiCO	http	yes	live	Adyen-MaajiCO	yes	Adyen	Adyen.Send.Authorize.3D
adyen.http.payment.methods. MaajiCO	http	yes	live	Adyen-MaajiCO	yes	Adyen	Adyen.Methods
adyen.http.payment.save. MaajiCO	http	yes	live	Adyen-MaajiCO	yes	Adyen	Adyen.Save
adyen.http.payment.send. MaajiCO	http	yes	live	Adyen-MaajiCO	yes	Adyen	Adyen.Send.Authorize

Credentials:

Name	Url	User *	Password
Adyen.Methods	https://test.adyen.com/hpp/directory.shtml		
Adyen.Save	https://pal-test.adyen.com/pal/servlet/Recurring/listRecurringDetails	ws_007192@Company.MaajiSwimwear	7c&ge?pR{C5!6Jy<~
Adyen.Send.Authorize	https://pal-test.adyen.com/pal/servlet/Payment/authorise	ws_007192@Company.MaajiSwimwear	7c&ge?pR{C5!6Jy<~
Adyen.Send.Authorize.3D	https://pal-test.adyen.com/pal/servlet/Payment/authorise3d	ws_007192@Company.MaajiSwimwear	7c&ge?pR{C5!6Jy<~

* User - Adyen web service name

** Password - Adyen web service password

Profile:

Name	Connection timeout (ms)	Enable Circuit Breaker	Max Circuit Breaker Calls	Circuit Breaker Interval (ms)	Enable Rate Limit	Max Rate Limit Calls	Rate Limit Interval (ms)
Adyen	2000	false	0	0	false	0	0

Payment Methods & Processors

Payment Method ID	Name	Payment Processor
CREDIT_CARD	Credit Card	ADYEN_CREDIT
Adyen	PayPal	Adyen

Custom Preferences

The Adyen cartridge has 1 custom preference groups that need to be configured: **Adyen**. It can be found under **{SiteName}/Merchant Tools/Custom preferences/**

Preference ID	Preference Name	
		Maaji Colombia
AdyenCseEnabled		true
AdyenCseJsPublicKey <i>(Adyen "Web service" encryption key)</i>		10001 BCF3A67A589342DC4E2BEFCA3A86B73826489359E50F28F9A91884CD3CCB6AA98600E225A
AdyenCseProdJsURL		
AdyenCseTestJsURL <i>(Adyen "Web service" CSE Library location)</i>		https://test.adyen.com/hpp/cse/js/8215205960383370.shtml
AdyenNotificationDelayMinutes		1
AdyenPaymentDeleteServicePrefix		
AdyenPaymentMethodsServicePrefix		adyen.http.payment.methods
AdyenPaymentSaveServicePrefix		adyen.http.payment.save
AdyenPaymentSecureServicePrefix		adyen.http.payment.3dsecure
AdyenPaymentSendServicePrefix		adyen.http.payment.send
Adyen_Debug		false
Adyen_HMACkey <i>(Adyen skin HMAC key)</i>		B822E2E20DE6633EC1AB0A01A0F4F46485768FF22478AE1C724068F29B3D5DF5
Adyen_Mode		TEST
Adyen_PaymentSelection		one
Adyen_address1Delimiter		

Adyen_allowedMethods		paypal
Adyen_directoryLookup		true
Adyen_enableAVS		false
Adyen_merchantCode		MaajiCOM
Adyen_recurringType		ONECLICK
Adyen_skinCode		OcCPQyiv
adyen_notification_password <i>(Adyen "Standard notification" user)</i>		London@2020!
adyen_notification_user <i>(Adyen "Standard notification" password)</i>		seafolly

Testing

Cards and numbers **without 3DS**

<https://docs.adyen.com/developers/test-cards/test-card-numbers>

Card Type	Number	Type	Issuing Country
Visa	4111111111111111	Consumer	NL
	4111111145551142 (*)	Classic	NL
	4988438843884305	Classic	ES
	4166676667666746	Classic	NL
	4646464646464644	Classic	PL
	4444333322221111	Corporate	GB
	4400000000000008	Debit	US
	4977949494949497	Gold	FR
MasterCard			
	222300004841010	Credit	NL
	222352044356010	Debit	NL
	2222410740360010	Corporate	NL
	510008111222332	Bijenkorf	NL
	5103221911199245	Prepaid	US
	5100290029002909	Consumer	NL
	5555341244441115 (*)	Consumer	NL
	5577000055770004	Consumer	PL
	5136333333333335	Consumer	FR
	5585558555855583	Consumer	ES
	5555444433331111	Consumer	GB
	555555555554444	Corporate	GB
AmEx	5000000000000004	Debit	US
	5424000000000015 (*)	Pro	EC
AmEx	370000000000002		NL

Diners	36006666333344		US
Discover	6011601160116611		US
	6445644564456445		GB

(*) Submitting the CVV for this card is optional.

Cards and numbers **with 3DS**

<https://docs.adyen.com/developers/test-cards/test-card-numbers#3dsecuretestcardsmastercardmaestrovisaamex>

Card Number	Card Type	Username	Password	Country
5212345678901234	MC	user	password	JP
4212345678901237	Visa	user	password	CA
345177925488348	Amex	user	password	
3569990010095833	JCB	user	password	US
6731012345678906	International	user	password	NL
6771830999991239	Maestro	user	password	GB

CVV/CID

<https://docs.adyen.com/developers/test-cards/test-card-numbers#expirydateandcvv>

Expiration Date	CVV2 / CVC3	CID (American Express)
08/2018	737	7373
10/2020	737	7373

After the linking of Adyen and PayU, the orders must be placed with statuses "APPROVED", "REJECTED" or "PENDING" in the "Name on card" field.

The screenshot shows a payment form with the following fields:

- Logos:** American Express, VISA, and PayPal.
- NAME ON CARD:** APPROVED
- CARD NUMBER:** 4111 1111 1111 1111
- Expiry Date:** 10 2020
- SECURITY CODE:** 737
- Checkbox:** Utiliza esta dirección para facturación (checked)
- Buttons:** ENVIAR ORDEN (pink button), GODADDY VERIFIED & SECURED (green shield icon).

At the bottom, it says: By continuing you agree to our [Terms & Conditions](#).

Useful Links

<https://docs.adyen.com/developers/test-cards/test-card-numbers>

<https://docs.adyen.com/api-explorer/#/PaymentSetupAndVerificationService/v32/payments>

<https://ca-test.adyen.com/ca/ca/config/accountapi.shtml> Additional data in API response
<https://docs.adyen.com/developers/api-reference/hosted-payment-pages-api/directory-request>
<https://docs.adyen.com/developers/ecommerce-integration/cse-integration-ecommerce>
<https://docs.adyen.com/developers/features/hosted-payment-pages/hmac-signature-calculation>

Credit Card fields validation & formatting

<https://github.com/braintree/card-validator>

<https://github.com/braintree/credit-card-type>

<https://github.com/braintree/formatter.js>

AfterPay Integration

The integrated version of the AfterPay cartridge is 1.0.0 according the documentation:

[Afterpay Payment LINK Integration Documentation.docx](#)

Sandbox Account

Environment	Login Url	User	Password	Type
Sandbox	https://portal-sandbox.afterpay.com/merchant	Iliyana.Mihaylova@tryzens.com	London@2020!	Merchant
		venislava.duncheva@tryzens.com *	345ertDFG	
Sandbox	https://portal-sandbox.afterpay.com/	antonia.dimitrova@tryzens.com	London@2020!	Customer

* Linked to the Sandbox Merchant we use.

Customer Data

There is one sandbox customer, currently created with the following data:

- Email: antonia.dimitrova@tryzens.com
- Password: London@2020!
- Phone: 0488 800 000
- Verification Code: 111111
- Credit Card:
 - Number: 4111111111111111
 - CVV: 123
 - Ex. Date: 10/2020

Company Credentials

Environment	Merchant ID	Merchant Key
Sandbox	33716	f361d954cb91ce7670b4d186a403dbd9eb72d200b0108fa4b71ed2fe0f1bac1397a9033441edb965af29ba02e1e5ff1

Business Manager

System preferences & Services, Payment methods &Processors

To set up the AfterPay integration in the BM do the following:

- 1) import [system-objecttype-extensions-afterpay.xml](#)

2) import services-afterpay.xml

3) for Seafolly AU & Sunburn:

- create Payment Processor called "AFTERPAY_CREDIT"
- import payment-methods-afterpay.xml

4) add the "int_afterpay" cartridge **before** any other cartridges to all site paths (**it must be the first one on the right of the list, e.g. int_avatax:int_braintree:app_seafolly_controllers:app_seafolly_core:int_afterpay**)

If Payment Method "AFTERPAY_PBI" is disabled, the AfterPay functionality on PDP, Basket Page & checkout is also disabled!

Custom Preferences

Preference ID	Preference Name	Preference Values	
		Sunburn	Seafoll
apUserID	User ID	33716	
apUserKey	User Key	f361d954cb91ce7670b4d186a403dbd9eb72d200b0108fa4b71ed2fe0f1bac1397a9033441edb965af29ba02e1e5ff171	
apRedirectConfirmUrl	Redirect Confirm Url		
apRedirectCancelUrl	Redirect Cancel Url		
apDisplayPdpInfo	Afterpay Display Product Details Page Information	Yes	Yes
apDisplayCartInfo	Afterpay Display Cart Page Information	Yes	Yes
apPaymentMode	Afterpay Payment Mode	DIRECT CAPTURE (DIRECT_CAPTURE)	DIRECT CAPTURE (DIRECT_CAPTU
apLearnMoreUrl	Afterpay Terms URL		
apJavascriptURL	Afterpay Javascript URL		
apServiceName	Afterpay Service Name	afterpay.http.defaultendpoint	afterpay.http.defaultendpoint
apMaxThreshold	Afterpay maximum threshold	2000	2000
apMinThreshold	Afterpay minimum threshold	0	0
apLearnMoreImgSrc	Afterpay Learn More Image Source		
apLearnMoreImgSrcMobile	Afterpay Learn More Image Source - Mobile		
apCheckoutUrl	Afterpay Checkout URL		

Testing

- Sandbox is not attached to a payment host – responses are simulated via the CVV below:
 - 000 - Transaction Approved
 - 051 - Transaction Declined

Apple Pay Created accounts and Apple pay documents

Apple pay Development Account	
Url	https://developer.apple.com
User	Dimitar.Dimitrov@tryzens.com
Password	London@2020@!!

This account has admin rights in Apple pay and apple iTunes. It can be used for **creation of sandbox test accounts**.

Apple Pay SANDBOX TEST accounts		
	Apple username / password	Email Password
Test user 1	ddtestuser7777.sandbox@gmail.com / London@2020@!!	London2020!!
Test user 2	ddtestuser88.sandbox@gmail.com / London2020!!	London2020!!
Test user 3	ddtestuser55.sandbox@gmail.com / London2020!!	London2020!!

This will be the new user created for the Apple Pay account.

The standard creds could be used to enter the account.

ID:QA Tryzens <qa.team.jw.applepay@gmail.com>

Password: London@2020

Helpful documents for Apple Pay Integration

[Apple Pay on the Web with Adyen in SCC.docx](#)

[Jack Wills - Apple Pay test user manual.pdf](#)

Avalara Sales Tax Integration

The integrated version of the Avalara cartridge is 16.1.0 according the documentation:

[AvaTax_Integration_16.1.0.docx](#)

Sandbox Account

The Avalara Console can be accessed with the following credentials:

Environment	Login Url	Username	Password
Sandbox	https://admin-development.avalara.net/login.aspx https://sandbox.admin.avalara.com/ (newer version)	Zhivko.Rusev@tryzens.com antonia.dimitrova@tryzens.com	London@2020! London@2020!

Company Data, Account ID & License

For our integration we'll need the following data:

- **Account ID** - visible in the top right corner of the screen under "Account#". See [company-data.png](#)
- **License Key** - the license key is sent to the account admin when setting up the company and is not visible/accessible in the sandbox console. However, it can be reset under **Settings > Reset License Key**. All account admin user will receive the new key via email. See attached [reset-key.png](#)
- **Company Code** - this can be found under **Organizations**. See [company-data.png](#)

Currently, we have the following data:

Environment	Company Code	Account Name	Account ID/Number	License Key	WebService URL
Sandbox	SeaSun	Seafolly Pty Ltd	2000229531	7AA6AFF9E6E6C878	https://development.avalara.net/

Transactions

List of transactions can be found under "Transactions" in the main menu. By default the list is filtered to the daily ones and shows 10 entries. See [transactions-list.png](#)

When we select the number under Doc Code we can view all the information related to that one transaction ([transactions-view.png](#))

Business Manager

System Object Types & Services

Avalara has several services and custom preferences meta that need to be included in the Business Manager and must be configured.

- 1) import [avatax-metadata.xml](#) into the BM sandbox
- 2) import [avalara-services.xml](#) into the BM environment
- 3) add the cartridge to site Site path (**only for the Seafolly US site ???**)

Custom Preferences

Avalara has only one custom preferences group that needs to be configured for the Seafolly US. *The other 3 sites are not configured in the BM and in the code, so even configured they won't work!*

It is called **AvaTax** and can be found under [{SiteName}/Merchant Tools/Custom preferences/](#)

The preferences and the respective values are described below:

Preference ID	Preference Name	Preference Values			
		Seafolly AU	Seafolly SG	Seafolly US	Sunburn
ATEnable	Enable Avatax	No	No	Yes	No
ATEnableAddressValidation	Enable Address Validation	No	No	No	No
ATEnableTesting	Enable Test Pipelines	No	No	No	No
AtExcludedCountries	AVS Excluded Countries				
AtShipFromCity	Avatax ShipFrom City				
AtShipFromStateCode	Avatax ShipFrom StateCode				
AtShipFromZipCode	Avatax ShipFrom ZipCode				
AtShipFromCountryCode	Avatax ShipFrom Country Code				
AtCompanyCode	Avatax Company Code			SeaSun	
AtDefaultShippingMethodTaxCode	Avatax Default Shipping Method Tax Code (Tax)				
AtDefaultProductTaxCode	Avatax Default Product Tax Code (Tax)				

Services, Profiles and Credentials

The Avalara services can be configured in **Administration/Operations/Services**. For sandbox configured services refer to "[avalara-services.xml](#)". The same information is represented below.

There are 3 services using the same configuration, profile & credentials:

Name	Type	Enabled	Service Mode	Log Name Prefix	Communication Log Enabled	Profile	Credentials
avatax.soap.CancelTax	SOAP	Yes	Live	Avalara	Yes	Avalara_Tax_Service	Avalara_Sandbox_Avatax
avatax.soap.GetTax	SOAP	Yes	Live	Avalara	Yes	Avalara_Tax_Service	Avalara_Sandbox_Avatax
avatax.soap.PostTax	SOAP	Yes	Live	Avalara	Yes	Avalara_Tax_Service	Avalara_Sandbox_Avatax

Credentials:

Name	Url	User	Password
Avalara_Sandbox_Avatax	https://development.avalara.net/Tax/TaxSvc.asmx	2000229531 *	7AA6AFF9E6E6C878 **

* User - Account ID/Number

** Password - License Key

Profile:

Name	Connection timeout (ms)	Enable Circuit Breaker	Max Circuit Breaker Calls	Circuit Breaker Interval (ms)	Enable Rate Limit	Max Rate Limit Calls	Rate Limit Interval (ms)
Avalara_Tax_Service	30000	false	0	0	false	0	0

Useful Links

Description	Url
Avalara Help Center	https://help.avalara.com
Avalara Developer	https://developer.avalara.com
Testing	https://developer.avalara.com/tests/testing-your-integration/
SOAP API	https://developer.avalara.com/api-reference/avatax/soap/
License Key	https://help.avalara.com/Frequently_Asked_Questions/001/How_do_I_find_my_license_key%3F

Braintree Credit Card Integration

The integrated version of the Braintree cartridge is 17.1.1. according the documentation:

[Braintree_Integration_Guide_for_Salesforce_Commerce_Cloud.docx](#)

Braintree sandbox account

To gain access to the Braintree console & sandbox account use the following login data:

Login url	Username	Password
https://braintreepayments.sandbox.gateway.com	braintreecommunity	braintreecommunity

Authorizations & API Keys

To view the SDK keys, edit & create new tokenization & API keys navigate to **Account > User** and click on the "View Authorizations" link under section **API Keys, Tokenization Keys, Encryption Keys**. For reference see attached image "[sandbox-api-keys.png](#)"

The currently defined sandbox keys are the following:

API Keys	SDK Keys
Public Key: szmkqmn4s258v886	Merchant ID: 22cqryhtv6nsptgp
Private Key: 89138e48fd494b876dde5023eadba238	Environment: sandbox
Environment: sandbox	CSE Key: MIIBCgKCAQEAYaWZm3ytl8M4Vfc02Uhm55YvlyzsHC6m16QJmspw4A9Uxt80RsYZ1JZzQdt0Z8FytW

Payment methods, Merchants & Settings

You can see a list of all available merchants and their accepted payment methods under section "**Merchant Accounts**" in **Settings > Processing**. Refer to attached image "[sandbox-merchants.png](#)". When creating new merchant account you can choose the merchant currency & PayPal acceptance, but you can NOT choose the available credit cards!

NOTE: Merchant ID and Merchant Account ID are not to be confused! **Merchant ID** is uniquely generated by Braintree and can not be changed. **Merchant Account ID** can be created to accept different payments and currencies.

This section also gives you access to editing different settings like:

- Payment Methods - PayPal, ApplePay, AndroidPay
- Fraud Tools - AVS & CVV rules, Advanced Fraud Tools
- Vaulting - Credit card verification
- Transactions

Here's a list of the currently defined sandbox merchant accounts:

Merchant Account ID	Default	Currency	Enabled Payment Methods
tryzens	yes	EUR	PayPal Visa MasterCard Maestro MaestroUK Discover
seafolly_us	no	USD	PayPal Visa MasterCard Maestro MaestroUK Discover
seafolly_au	no	AUD	PayPal Visa MasterCard Maestro MaestroUK
seafolly_sg	no	SGD	PayPal Visa MasterCard Maestro MaestroUK
sunburn	no	AUD	PayPal Visa MasterCard Maestro MaestroUK

For available credit cards, supported regions & currencies refer to:

- <https://www.braintreepayments.com/en-bg/features/credit-cards>
- <https://articles.braintreepayments.com/get-started/currencies>
- <https://developers.braintreepayments.com/guides/credit-cards/overview>
- <https://articles.braintreepayments.com/guides/fraud-tools/3d-secure#card-types>

3D Secure

This feature is not enabled by default. To enable it contact the Braintree support team.

For additional information about 3DS refer to:

- <https://articles.braintreepayments.com/guides/fraud-tools/3d-secure>
- <https://developers.braintreepayments.com/guides/3d-secure/overview>
- <https://braintree.github.io/braintree-web/3.22.0/ThreeDSecure.html>
- <https://articles.braintreepayments.com/guides/fraud-tools/3d-secure#card-types>

Advanced Fraud Tools

In order for AFT to be usable they must be enabled in **Settings > Processing** under section **Fraud Tools > Advance Fraud Tools** by clicking the toggle button.

Additional information can be found here : <https://articles.braintreepayments.com/guides/fraud-tools/advanced/overview>

AFT risk decisions & statuses : <https://articles.braintreepayments.com/guides/fraud-tools/advanced/kount-standard#risk-decisions>

CVV Rules

CVV Rules must be enabled & configured for the card rejection to work. To do that go to **Settings > Processing**, under section **Fraud Tools > CVV** click on the **Options** link.

Additional info here: <https://articles.braintreepayments.com/guides/fraud-tools/basic/avs-cvv-rules#recommended-setup-options>

Risk Threshold Rules

Risk threshold rules can be added to gain additional verification and help detect fraud. The rules can be added in **Settings > Processing**, under section **Fraud Tools > Risk Thresholds**.

Additional info and best practices can be found here: <https://articles.braintreepayments.com/guides/fraud-tools/basic/risk-threshold-rules>

Transactions & History

- **Transactions list** - To access information about the transactions, their payment information, status & type, chose "**Transactions**" under "**Advanced Search**" in the left menu. Refer to attached screenshot "**sandbox-transactions-list.png**".
- **Transaction Info** - From the transactions list you click on the transaction ID link to see the full transaction information. Depending on the transaction status you can choose to "Submit for settlement", "Refund" or "Void". Refer to attached "**sandbox-transaction-data.png**".
- **Transaction Summary** - You can find transactions summary grouped by status, successful & unsuccessful data when choosing "**Transaction Summary**" under "**Transactions**" in the left menu. Refer to "**sandbox-transactions-summary.png**".
- **Customers list** - To view the customers list choose "**Vault**" under "**Advanced Search**" in the left menu. For an example list of customers refer to "**sandbox-vault-customers.png**".
- **Customer payment method info** - When the customers list loads, clicking on the token link against each credit card gives you full information such as customer information, card information, billing address & last 10 transactions. Refer to "**sandbox-card-payments.png**".
- **Expiring Credit Cards** - You can see a list of expiring cards under "**Expiring Cards**" in the "**Vault**" section in the left menu. Refer to "**sandbox-cards-expire.png**".

Business Manager

System preferences & Services, Payment methods & Processors

To have the sites up and running import the metadata needed & configure each one of the defined sites. To do so:

1) import the "[services.xml](#)" metadata into the sandbox;

2) import the "[payment-meta.xml](#)" metadata into the sandbox;

3) configure each site:

- import "[payment-processors.xml](#)" and "[payment-methods.xml](#)" meta data into the sandbox (currently using the same data & settings);
- make sure that the "CREDIT_CARD" payment method is enabled & configured to use the "BRAINTREE_CREDIT" processor (Check under **{SiteName}/Merchant Tools/Ordering/Payment Methods** & **{SiteName}/Merchant Tools/Ordering/Payment Processors**);
- make sure that the defined credit cards are enabled (Check under **{SiteName}/Merchant Tools/Ordering/Payment Methods**);
- configure the Braintree Credit Card custom preferences for each site - refer to the next section.

4) add the cartridge to site Site path.

Custom Preferences

The Braintree cartridge has 2 custom preference groups that need to be configured: **BRAINTREE_COMMON** and **BRAINTREE_CREDIT**, for common & card specific setting respectively. Both can be found under **{SiteName}/Merchant Tools/Custom preferences**/

Currently the development sandboxes have the following configurations:

- Braintree Common Preferences - {SiteName}/Merchant Tools/Custom preferences/BRAINTREE_COMMON

Preference ID	Preference Name	Preference Values			
		Seafolly US	Seafolly AU	Seafolly SG	Sunburn
BRAINTREE_Merchant_ID	Merchant ID **	22cqryhtv6nsptgp	22cqryhtv6nsptgp	22cqryhtv6nsptgp	22cqryhtv6nsptgp
BRAINTREE_ECVZ_Access_Token	ECVZ Access Token				
BRAINTREE_ECVZ_Api_Url	ECVZ API URL				
BRAINTREE_Merchant_Account_ID	Merchant Account ID * *	seafolly_us	seafolly_au	seafolly_sg	sunburn
BRAINTREE_Tokenization_Key	Tokenization Key				
BRAINTREELogging_Mode	Logging *	all	all	all	all
BRAINTREE_SG_Controllers_Cartridge_Name	SiteGenesis Controllers Cartridge Name	app_seafolly_controllers	app_seafolly_controllers	app_seafolly_controllers	app_seafolly_cc
BRAINTREE_Log_Name_Prefix	Log Name Prefix ***	Braintree_service_US	Braintree_service_AU	Braintree_service_SG	Braintree_service

* Logging - In order for communication logging to work, log level must be set to DEBUG for "root" category and INFO/DEBUG levels must be enabled for "Log Files" under **Administration/Operations/Custom Log Settings**.

** Merchant ID and Merchant Account ID are not to be confused. **Merchant ID** is uniquely generated by Braintree and can not be changed. **Merchant Account ID** can be created in the Braintree console to accept different payments and currencies.

*** Log Name Prefix - As defined in the respective site service under "Log Name Prefix".

- Braintree Credit Cards Preferences - {SiteName}/Merchant Tools/Custom preferences/BRAINTREE_CREDIT

Preference ID	Preference Name	Preference Values			
		Seafolly US	Seafolly AU	Seafolly SG	Sunburn
BRAINTREE_Payment_Model	Payment Model	Auth and Immediate Settlement (sale)			
BRAINTREE_Vault_Mode	Store Payment Method in Vault	Do not vault (not)			
BRAINTREE_Fraud_Tools_Enabled	Advanced Fraud Tools	Enabled	Enabled	Enabled	Enabled
BRAINTREE_Credit_Card_Statement_Name	Credit Card Statement Name				
BRAINTREE_3DSecure_Enabled	3DSecure	Enabled	Enabled	Enabled	Enabled
BRAINTREE_3DSecure_Skip_Client_Validation_Result	Skip 3DSecure result	Disabled	Disabled	Disabled	Disabled
BRAINTREE_Hosted_Fields_Styling	Hosted Fields Styling				
BRAINTREE_Hosted_Fields_Advanced_Options	Hosted Fields Advanced Options				
BRAINTREE_CREDIT_Custom_Fields	Default Custom Fields				
BRAINTREE_CREDIT_Descriptor_Phone	Descriptor Phone				
BRAINTREE_CREDIT_Descriptor_Name	Descriptor Name				
BRAINTREE_CREDIT_Descriptor_URL	Descriptor URL				

Services, Profiles and Credentials

The Braintree services can be configured in **Administration/Operations/Services**. Each of the sites must have their own service with the following naming convention **int_braintree.http.xml.payment.Braintree.{SiteID}**. For sandbox configured services refer to "[services.xml](#)". The same information is represented below.

Currently there are 4 services using the same configuration, profile & credentials:

Name	Type	Enabled	Service Mode	Log Name Prefix
int_braintree.http.xml.payment.Braintree.SeafollyAU	http	yes	live	Braintree_service_AU
int_braintree.http.xml.payment.Braintree.SeafollySG	http	yes	live	Braintree_service_SG
int_braintree.http.xml.payment.Braintree.SeafollyUS	http	yes	live	Braintree_service_US
int_braintree.http.xml.payment.Braintree.Sunburn	http	yes	live	Braintree_service_SB

* Log Name Prefix - as defined in the custom BRAINTREE_COMMON preferences under Preference ID "Log Name Prefix".

Credentials:

Name	Url	User *	Password **
Braintree_Sandbox_Credentials	https://api.sandbox.braintreegateway.com/	szmkqmn4s258v886	89138e48fd494b876dde5023eadba238

* User - API public key

** Password - API private key

Profile:

Name	Connection timeout (ms)	Enable Circuit Breaker	Max Circuit Breaker Calls	Circuit Breaker Interval (ms)	Enable Rate Limit	Max Rate Limit Calls	Rate Limit Interval (ms)
Braintree_Default_Profile	60000		0	0		0	0

Testing

Credit Cards

Info can be found here: <https://developers.braintreepayments.com/reference/general/testing/java>

Card Type	Supported	Enabled	Number	Expected Result	Actual Result w/ 3DS (validation/order)	Actual Result w/o 3DS (validation/order)
Visa	Yes	Yes	4111111111111111	successful	successful/order created	successful/order created
			4005519200000004	successful	unsuccessful	successful/order created
			4009348888881881	successful	unsuccessful	successful/order created
			4012000033330026	successful	unsuccessful	successful/order created
			4012000077777777	successful	successful(no 3ds verify needed)/order created	successful/order created
			4012888888881881	successful	unsuccessful	successful/order created

			4217651111111119	successful	unsuccessful	successful/order created
			4500600000000061	successful	unsuccessful	successful/order created
			4000111111111115	unsuccessful	unsuccessful	successful/order created
			4000111111111511	fraud detected gateway reject	unsuccessful	successful/order failed
Maestro	Yes	Yes	6304000000000000	successful	unsuccessful	unsuccessful
Amex	No	No	378282246310005	successful	unsupported	successful/order failed
			371449635398431	successful	unsupported	successful/order failed
			378734493671000	unsuccessful	unsupported	successful/order failed
DinersClub	Yes	No	36259600000004	successful	unsuccessful	successful/order created
			38520000009814	unsuccessful	unsuccessful	successful/order created
Discover	Yes	Yes	6011111111111117	successful	unsuccessful	successful/order created
			6011000990139424	unsuccessful	unsuccessful	successful/order created
JCB	No	No	3530111333300000	successful	unsupported	unsupported
			3566002020360505	unsuccessful	unsupported	unsupported
MasterCard	Yes	Yes	555555555554444	successful	unsuccessful	successful/order created
			2223000048400011	successful	unsuccessful	successful/order created
			5105105105105100	unsuccessful	unsuccessful	successful/order created

Amounts

Amount	Expected Result	Actual Result
0.01 - 1999.99	Authorized/Settled	Authorized/Settled
2000.00 - 3000.99	Processor Declined	Processor Declined
3001.00 - 4000.99	Authorized/Settled	
4001.00 - 4001.99	Authorized/Settlement Declined on PP Authorized/Settled on all others	
4002.00 - 4002.99	Authorized/Settlement Pending on PP Authorized/Settled on all others	
4003.00 - 5000.99	Authorized/Settlement Declined on PP Authorized/Settled on all others	

5001.00	Gateway Reject	
5001.01	Processor Declined on PP Authorized/Settled on all others	
5001.02	Authorized/Processor Unavailable Authorized/Settled	
5002.00 and up	Authorized/Settled	

CVV

CVV	CID (Amex)	Expected Result	Result w/o CVV Rule	Result with CVV Rule
200	2000	N (does not match)	order created; risk status: approve; transaction: settled	order rejected/failed; transaction: rejected
201	2001	U (not verified)	order created; risk status: approve; transaction: settled	order rejected/failed; transaction: rejected
301	3001	S (issuer does not participate)		
not provided	not provided	I (not provided)		
all others	all others	M (matches)	order created	order created

3DSecure

3DS Cards by CardinalCommerce.

Testing info & CardinalCommerce test cases here: <https://developers.braintreepayments.com/guides/3d-secure/testing-go-live/java>

Also see: [CardinalCommerce Test Cards for Braintree 3DS](#)

All cards expiry date is: 01/2020, CVV: 123 (1234 for AmEx)

Cards in **green** should be the ones that pass verification!

Card Type	Supported	Enabled	Number	Expected Result	Actual Result
Visa	Yes	Yes	4000000000000002	verified	verified
			4000000000000010	failed verification	failed ; form not loading
			4000000000000028	failed verification	failed ; form not loading
			4000000000000101	no verification	no verification
			4000000000000044	failed (timeout)	failed
			4000000000000051	failed (not enrolled)	failed
			4000000000000069	failed (unavailable)	failed
			4000000000000077	failed (not active)	failed
			4000000000000085	failed (lookup error)	failed
			4000000000000093	failed (auth error)	form not loading
			4000000000000036	failed (auth unavailable)	failed; form not loading
			4000990000000004	failed (bypass auth)	failed
MasterCard	Yes	Yes	5200000000000007	verified	verified
			5200000000000015	failed verification	failed ; form not loading
			5200000000000023	failed verification	failed ; form not loading
			5200000000000908	no verification	no verification

			5200000000000106	activation during shopping; verified	verified
			5200000000000049	failed (timeout)	failed
			5200000000000056	failed (not enrolled)	failed
			5200000000000064	failed (unavailable)	failed
			5200000000000072	failed (not active)	failed
			5200000000000080	failed (lookup error)	failed
			5200000000000098	failed (auth error)	form not loading
			5200000000000031	failed (auth unavailable)	failed; form not loading
			5200990000000009	failed (bypass auth)	failed
AmEx	Yes	Yes	340000000003961	verified	verified
			340000000006022	failed verification	failed verification
			340000000000033	failed verification	
			340000000003391	no verification	no verification
			340000000008309	failed (timeout)	
			340000000008135	failed (not enrolled)	
			340000000007780	failed (unavailable)	
			340000000008416	failed (not active)	
			340000000006337	failed (lookup error)	
			340000000009299	failed (auth error)	
			340000000002302	failed (auth unavailable)	
			340000000004415	activation during shopping; verified	
			3400990000000001	failed (bypass auth)	

Attachments

Images	Meta Data	Documentation	FSD	Design
sandbox-api-keys.png	payment-processors.xml	Braintree Integration Guide		
sandbox-merchants.png	payment-methods.xml	CardinalCommerce Test Cards for Braintree 3DS		
sandbox-vault-customers.png	payment-meta.xml services.xml			
sandbox-transactions-list.png				
sandbox-transaction-data.png				
sandbox-transactions-summary.png				
sandbox-card-payments.png				
sandbox-cards-expire.png				
sandbox-risk-tresholds.png				

Useful Links

Description	Link
Braintree Cartridge	https://xchange.demandware.com/docs/DOC-35457

Hosted Fields Overview	https://developers.braintreepayments.com/guides/hosted-fields/overview/javascript/v3
Hosted Fields Examples	https://developers.braintreepayments.com/guides/hosted-fields/examples/javascript/v3
Hosted Fields Styling Options	https://braintree.github.io/braintree-web/3.23.0/module-braintree-web_hosted-fields.html#~styleOptions
Hosted Fields Documentation	https://braintree.github.io/braintree-web/3.23.0/HostedFields.html
Transaction Statuses	https://developers.braintreepayments.com/reference/general/statuses#transaction
Testing Information	https://developers.braintreepayments.com/reference/general/testing/java
AVS & CVV Rules	https://articles.braintreepayments.com/guides/fraud-tools/basic/avs-cvv-rules
Processor Responses	https://developers.braintreepayments.com/reference/general/processor-responses/authorization-responses
Testing 3DS Information	https://developers.braintreepayments.com/guides/3d-secure/testing-go-live/java
Testing Credit Cards Info	https://developers.braintreepayments.com/guides/credit-cards/testing-go-live/java

Linked Integrations

Description	Link
PayPal by Braintree	PayPal Integration

Brands page for Sunburn site

For Sunburn website we create Brands page. This is a guide how to create and use this page.

Brands Page

To create Brands page first you must create category Brands.

To make this category go to Merchant Tools. Open Products and Catalogs and then select Catalogs link. From this page select the catalog for which you want to add this category.

Select All	ID	Site Assignments	Categories	Products Owned	Products Assigned	Recommendations	Status	
<input type="checkbox"/>	apparel-catalog		28	4,474	4,467	0	Online	Edit
<input type="checkbox"/>	electronics-catalog		15	159	159	0	Online	Edit
<input type="checkbox"/>	storefront-catalog-en	SiteGenesis Seafolly Australia Seafolly United States Seafolly Singapore Sunburn	71	0	4,632	1,865	Online	Edit
<input type="checkbox"/>	storefront-catalog-non-en	SiteGenesisGlobal	42	0	4,473	1,865	Online	Edit
Refresh							New	Delete

This will open a new page and from it you can make Brands category.

Select All	ID	Name	Description	Status	View	Sorting
<input type="checkbox"/>	newarrivals	New Arrivals			Edit	
<input type="checkbox"/>	womens	Womens			Edit	
<input type="checkbox"/>	mens	Mens			Edit	
<input type="checkbox"/>	electronics	Electronics			Edit	
<input type="checkbox"/>	gift-certificates	Gift Certificates			Edit	
<input type="checkbox"/>	top-seller	Top Sellers			Edit	
<input type="checkbox"/>	brands	Brands			Edit	
<input type="checkbox"/>	hidden	Hidden Category	Hidden Category to store categories such as sale etc.			

[Edit All](#) [Edit Selected](#) [Sort](#) [Unsort](#) [New](#) [Copy](#) [Delete](#)

Rendering template

We create rendering template for this page and is called 'brandListingPageRendering'. You can find it in rendering/category path

This template will be set for category rendering template.

Presentation Attributes	
Thumbnail Image:	<input type="text"/> Select
Standard Image:	<input type="text"/> Select
Rendering Template:	<input type="text" value="rendering/category/brandListingPageRendering"/> Select
Category Landing Slot Image:	<input type="text"/> Select
Custom CSS File:	<input type="text"/> Select

Brands logos to main navigation

We add 'Brand Page' checkbox to category attributes. When you want to add brand images to menu navigation you must check this checkbox. This will enable dropdown with selected brands.

Subcategories

When this category is created you can add subcategories to it. To do this click on brands id link to open new page.

Categories

Brands [Edit](#)

These are the sub-categories of the current category. To change the order of the sorted categories, use the up and down buttons to move selected categories. To add a category to the list of sorted categories, select the category and use the Sort button.

Select All	ID	Name	Description	Status	View
<input type="checkbox"/>	brand 1	brand 1	brand 1		Edit
<input type="checkbox"/>	brand 2	brand 2	brand 2		Edit

[Edit All](#) [Edit Selected](#) [Sort](#) [Unsort](#) [New](#) [Copy](#) [Delete](#)

From this page you can create a new subcategory by clicking 'New' action button. This will open a new page. On General tab enter the required information and click 'Apply'. Then you can select Category Attributes tab and add brand logo image. Find 'Images for brand logo' field and select image for brand logo and click 'Apply'.

Alternative URL:	<input type="text"/>
HTML Editor	<input type="text"/>
Images for brand logo:	<input type="text"/> Select
Apply Reset	

Preference Id

The id of this preference is **brandLogo**. This preference is from type Image. You can find it on the follow path Administration > Site Development > System Object Types > Category

CSC - Customer Service Center

- General
 - Enabling CSC Module
- Meta Data
 - System Objects
 - Customer Service Center Settings

General

An order is marked as either "Shipped", "Cancelled" or "Returned" in AP21 when an Order Status Update (OSU) feed (from Fusion Factory Flow middleware) is sent to and received by SCC. Then SCC surfaces the updated order status in the Order Summary screen within the BM Customer Services Center.

Customer Service Center is turned off by default. You must have permission to be able to use the Customer Service Center module.

Enabling CSC Module

To be able to use Customer Service Center, you must have been granted permission in Business Manager; the following permissions must be granted to a Customer Service Representative:

1. In Business Manager: Administration > Organization > Roles & Permissions.
2. Click the Business Manager modules tab
3. Select your site.
4. Enable the following permissions:
 - Orders
 - Customer Service Center
5. Click the Functional Permissions tab.
6. Select your site.
7. Enable the following permissions:
 - Adjust_Item_Price
 - Adjust_Shipping_Price
 - Adjust_Order_Price
 - Delete_Order_Note
 - Create_Order_On_Behalf_Of
 - Search_Orders
8. Click the Locale Permissions tab.
9. Check locales to enable them, uncheck to disable them for this role.
10. Click Apply.

Changes been made

The screenshot shows the Order Summary page in the Salesforce Commerce Cloud interface. At the top, there's a navigation bar with icons for commerce cloud, Welcome, and Order Summary. On the right, there are user details: devadmin and Seafolly Australia. The main content area displays an order summary for Order No: 00003101. The order details include:

- Creation Date:** 11/16/2017 12:14 pm
- Source Code:** [redacted]
- Order Status:** OPEN
- Confirmation Status:** CONFIRMED
- Export Status:** EXPORTREADY
- Shipping Status:** NOTSHIPPED
- Payment Status:** PAID
- Is A Gift:**
- Gift Message:** Iliyana Mihaylova
- Warehouse Status:** Order Shipped
- Warehouse History:** Being Processed on 01-01-2017, Order Shipped on 01-01-2017

A red box highlights the gift message and warehouse status information. Below this, the Shipment 1 section shows the shipping address and item details:

Name	Availability	Quantity	Price	Tax	Total
Button Front Shirt Color: Black Size: 6 701643464166	Available (99)	1	A\$ 30.71	A\$ 1.46	A\$ 30.71

At the bottom, there are buttons for Add Note and a summary of charges:

Shipping Method: Express Shipping	Tax: A\$ 0.47
Items Total: A\$ 30.71	
Shipping Total: A\$ 0.95	
Tax Total: A\$ 1.93	
Order Total: A\$ 40.66	

Meta Data

System Objects

Orders

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://www.demandware.com/xml/impex/metadata/2006-10-31">
    ...
    <type-extension type-id="Order">
        ...
        <custom-attribute-definitions>
            ...
            <attribute-definition attribute-id="orderHeaderStatus">
                <display-name xml:lang="x-default">Warehouse
                Status</display-name>
                <type>enum-of-string</type>
                <mandatory-flag>false</mandatory-flag>
                <externally-managed-flag>false</externally-managed-flag>
                <value-definitions>
                    <value-definition default="true">
                        <display xml:lang="x-default">Being
                        Processed</display>
                        <value>Being Processed</value>
                    </value-definition>
                    <value-definition>
                        <display xml:lang="x-default">Order
                        Shipped</display>
                        <value>Order Shipped</value>
                    </value-definition>
                    <value-definition>
                        <display xml:lang="x-default">Order
                        Cancelled</display>
                        <value>Order Cancelled</value>
                    </value-definition>
                    <value-definition>
                        <display xml:lang="x-default">Order
                        Returned</display>
                        <value>Order Returned</value>
                    </value-definition>
                </value-definitions>
            </attribute-definition>
            <attribute-definition attribute-id="orderHeaderTime">
                <display-name xml:lang="x-default">Warehouse
                History</display-name>
                <type>set-of-string</type>
                <mandatory-flag>false</mandatory-flag>
                <externally-managed-flag>false</externally-managed-flag>
            </attribute-definition>
            ...
        </custom-attribute-definitions>
    <group-definitions>
```

```
...
<attribute-group group-id="OrderHeader">
    <display-name xml:lang="x-default">Warehouse
OSU</display-name>
    <attribute attribute-id="orderHeaderStatus" />
    <attribute attribute-id="orderHeaderTime" />
</attribute-group>
...
</group-definitions>
</type-extension>
```

```
...
...
</metadata>
```

Customer Service Center Settings

Order Summary

```
{
  "_v" : "4",
  "data": {
    "attributes": {
      "gift_message": {
        "read": {
          "resource": "DefaultShipment",
          "path": "gift_message"
        }
      }, "gift": {
        "read": {
          "resource": "DefaultShipment",
          "path": "gift"
        }
      }
    }
  },
  "layout" : [ {
    "layout" : [ {
      "id" : "column1",
      "width" : 3
    }, {
      "id" : "column2",
      "width" : 3
    }, {
      "id" : "column3",
      "layout" : [ {
        "id" : "status_title",
        "width" : 12
      }, {
        "id" : "status_listing",
        "width" : 12
      } ],
      "width" : 6
    } ]
  }, {
    "widgets" : {
      "status_title" : {
        "type" : "text",
        "data" : {
          "template" : {
            "default" : "Order Status:"
          }
        }
      }
    }
  }
}
```

```
        }
    },
    "status_listing" : {
        "type" : "attribute_listing",
        "config" : {
            "attributes" : [ "status", "confirmation_status", "export_status",
"shipping_status", "payment_status", "gift", "gift_message",
"c_orderHeaderStatus", "c_orderHeaderTime" ],
            "columns" : 1,
            "orientation" : "horizontal",
            "show_empty" : true
        }
    },
    "column1" : {
        "type" : "attribute_listing",
        "config" : {
            "attributes" : [ "order_no" ],
            "columns" : 1,
            "show_empty" : true
        }
    },
    "column2" : {
        "type" : "attribute_listing",
        "config" : {
            "attributes" : [ "creation_date", "source_code" ],
            "columns" : 1,
            "meta" : {
                "source_code" : {
                    "label" : {
                        "default" : "Source Code"
                    }
                }
            },
            "show_empty" : true
        }
    },
    "column3" : {
        "type" : "attribute_layout"
```

```
    }  
}  
}
```

Custom Preferences

Storefront Configs	Storefront Configurations
--------------------	---------------------------

Name	Value	Default Value
Default List Price Book ID		usd-list-prices
Disable Responsive Design	Disable Responsive Design for Storefront	No
GoogleVerificationTag - Content Attribute		
Build Notification Email List	Semicolon separated list	
Slide Slow effect	Please visit http://jquery.malsup.com/cycle/browser.html for complete visual effects	curtainX (expand from center horizontally)
Enable Infinite Scroll	This Boolean toggles the option for infinite scrolling or pagination from within the search grid.	No
Enable MultiShipping	The attribute is used to enable/disable Multi-Shipment Checkout for each site via BM.	Yes
Enable In Store Pickup	This Boolean enables the storefront ability to perform tasks related to in store orders	Yes
Default Country Code	Default country code for store searches	USA
Store Lookup Unit	Used in finding the store near the ZIP code specified in the storefront by the end user.	miles
Store Lookup Max Distance	Store Lookup Max Distance	75
Customer Service Email		noreply@salesforce.com
Contact Us Phone Number		
Rate Limiter Threshold	This number indicates the number of times failures for Logins, GiftCert Balances, and Order Tracking requests will be allowed before displaying a CAPTCHA-style pop-up rate limiter	5
Check TLS	When true, this causes a TLS check to take place based on whether or not a cookie ('dw_TLSWarning') has been set on the browser	No

Seafolly Configs	Seafolly Configurations
------------------	-------------------------

Name	Value	Default Value
Image Swap Enable		Yes
Prices for e-gift vouchers	Add 50 100 200 600 Add price values to gift vouchers	
Recently viewed items	Enter a number to choose how much recently viewed element to show in storefront	
Enable recently viewed	Enable recently viewed items	
Enable recommendations	Enable recommendations	
Shipping method preferences		
Supported Shipping Countries		
SPC address fields constraints		
Test Email Path		
Site URL in Site Selector switcher	Custom preference for the URL of the site option in the Site selector switcher.	

Custom rendering template with 6 content slots

For our needs we created a template with 6 content slots.

Configured slots

1. First slot
 - a. Slot ID: first-page-content-slot
 - b. Rendering template: Not set
2. Second slot
 - a. Slot ID: second-page-content-slot
 - b. Rendering template: Not set

3. Third slot
 - a. Slot ID: third-page-content-slot
 - b. Rendering template: Not set
4. Fourth slot
 - a. Slot ID: fourth-page-content-slot
 - b. Rendering template: Not set
5. Fifth slot
 - a. Slot ID: fifth-page-content-slot
 - b. Rendering template: Not set
6. Sixth slot
 - a. Slot ID: sixth-page-content-slot
 - b. Rendering template: Not set

For each slot you can set different rendering template.

Rendering template

This rendering template is called sixslots and you can find it in rendering/category

Category

To use this template you must set it like rendering template to some category. Follow this path Merchant Tools > Products and Catalogs > Catalogs to open catalogs. Select catalog and then category for which you want to make this change. You must select Edit button action to change rendering template. On Category Attributes tab find Rendering Template field.

Presentation Attributes	
Thumbnail Image:	<input type="text"/> Select
Standard Image:	<input type="text"/> Select
Rendering Template:	<input type="text" value="rendering/category/sixslots"/> Select
Category Landing Slot Image:	<input type="text"/> Select
Custom CSS File:	<input type="text"/> Select

Email module Seafolly/Sunburn

All emails could be tested on following link on development instance:

https://development-ap01-seafolly.demandware.net/on/demandware.store/Sites-SeafollyAU-Site/en_AU/TestEmail

https://development-ap01-seafolly.demandware.net/on/demandware.store/Sites-Sunburn-Site/en_AU/TestEmail

Testing:

- **Welcome email on successful registration**

Enter a valid Email: Email address of your profile;

- **Forgot password email**

Enter a valid Email: Email address of your profile;

- **New password confirmation email (after password reset)**

Enter a valid Email: Email address of your profile;

- **Email product to a friend**

Friend's Name: Test friend name

Message to Friend: Test message to your friend

Product ID: The ID of the product you want to send

Email to Send to: The Email address of your friend

- **Email wish list to a friend**

Friend's Name: Test friend name
Message to Friend: Test message to your friend
Email to Send to: The Email address of your friend

- **Email with purchased e-gift card details**

Email Subject: Test message display in subject line
Friend's Name: Test friend name
Message to Friend: Test message to your friend
Gift Code: Test gift card code
Gift Amount: Test gift amount (number value)
Email to Send to: The Email address of your friend
Gift pin: Test pin (number value)
Gift Expiry: Test Expiry Date

- **Email with returned e-gift card details**

Friend's Name: Test friend name
Gift Code: Test gift card code
Gift Amount: Test gift amount (number value)
Email to Send to: The Email address of your friend
Gift pin: Test pin (number value)
Gift Expiry: Test Expiry Date

- **Order confirmation email**

Enter a valid Email : Email address you want the mail to sent
Order Number: the order number of the test order which has been placed already

- **Order dispatched/unable to fulfil email**

Enter a valid Email : Email address you want the mail to be sent
Order Number: The order number of the test order which has been placed already

- **Order cancellation email**

Enter a valid Email : Email address you want the mail to be sent
Order Number: The order number of the test order which has been placed already

- **Order returned email**

Order Number: the order number of the test order which has been placed already and has a request send for return of the items. The request for return is made on the order details page.

Email module has two cartridges: **app_emailmodule** and **test_emailmodule**. It works with the custom objects **emailTemplate** and **emailComponent**. The text in the emails is allocated in **content assets**. Read more information in the attachment "[email_module_user_guide.docx](#)".

_emails Structure

1. Welcome email on successful registration --> emailTemplateID == newAccount

Structure:

```
$newsletter-header$ - asset: newsletter-header
$newaccount-newsletter-body$ - asset: newaccount-newsletter-body
$newsletter-footer$ - asset: newsletter-footer
```

2. Email product to a friend -> emailTemplateID == sendToFriend

Structure:

```
$newsletter-header$ - asset: newsletter-header
$send-to-friend-body-customer-msg$ - asset: send-to-friend-body-customer-msg
$send-to-friend-body-product-info$ - email template: email_module/sendtofriend
$newsletter-footer$ - asset: newsletter-footer
```

3. Email wish list to a friend -> emailTemplateID == wishList

Structure:
\$newsletter-header\$ - asset: newsletter-header
\$wish-list-body-customer-msg\$ - asset: wish-list-body-customer-msg
\$newsletter-footer\$ - asset: newsletter-footer

4. Order confirmation email -> emailTemplateID == orderConfirmation

Structure:
\$newsletter-header\$ - asset: newsletter-header
\$order-confirmation-body\$ - asset: send-to-friend-body-customer-msg
\$order-details\$ - email template: email_module/orderconfirmation
\$newsletter-footer\$ - asset: newsletter-footer

5. Order despatched/unable to fulfil email-> emailTemplateID == dispatchedOrderEmail

Structure:
\$newsletter-header\$ - asset: newsletter-header
\$dispatched-email-content-msg\$ - asset: dispatched-email-content-msg
\$order-details\$ - email template: email_module/orderconfirmation
\$newsletter-footer\$ - asset: newsletter-footer

6. Order cancellation email-> emailTemplateID == cancelledOrderEmail

Structure:
\$newsletter-header\$ - asset: newsletter-header
\$cancelled-email-content-msg\$ - asset: cancelled-email-content-msg
\$order-details\$ - email template: email_module/orderconfirmation
\$newsletter-footer\$ - asset: newsletter-footer

7. Order returned email -->emailTemplateID == returnedEmail

Structure:
\$newsletter-header\$ - asset: newsletter-header
\$returned-order-message-content\$ - asset: returned-order-message-content
\$orderreturnrequest\$ - email template: email_module/returnordermail
\$newsletter-footer\$ - asset: newsletter-footer

8. Forgot password email --> emailTemplateID == resetPassword

Structure:
\$newsletter-header\$ - asset: newsletter-header
\$resetpassword-newsletter-body\$ - asset: resetpassword-newsletter-body
\$newsletter-footer\$ - asset: newsletter-footer

9. New password confirmation email (after password reset) --> emailTemplateID == passwordResetSuccess

Structure:
\$newsletter-header\$ - asset: newsletter-header
\$pass-reset-succ-body\$ - asset: resetpassword-newsletter-body
\$newsletter-footer\$ - asset: newsletter-footer

10. Email with purchased e-gift card details --> emailTemplateID == giftCardEmail

Structure:
\$newsletter-header\$ - asset: newsletter-header
\$gift-purchase-email-content\$ - asset: gift-purchase-email-content
\$gift-purchase-voucher-content\$ - asset: gift-purchase-voucher-content
\$newsletter-contact-us\$ - asset: newsletter-contact-us
\$newsletter-footer\$ - asset: newsletter-footer

11. Email with returned e-gift card details --> emailTemplateID == giftCardRefundedEmail

Structure:
\$newsletter-header\$ - asset: newsletter-header
\$gift-refunded-email-content\$ - asset: gift-refunded-email-content
\$newsletter-contact-us\$ - asset: newsletter-contact-us
\$newsletter-footer\$ - asset: newsletter-footer

Seafolly Email module content configurations:

Demandware_Automated_Emails_SF.docx

Sunburn Email module content configurations:

Demandware_Automated_Emails_SB.docx

Email module user guide:

email_module_user_guide.docx

Designs:

email-design-Sunburn.zip, email-design-seafolly.zip

Email templates screenshots:

email_templates.zip

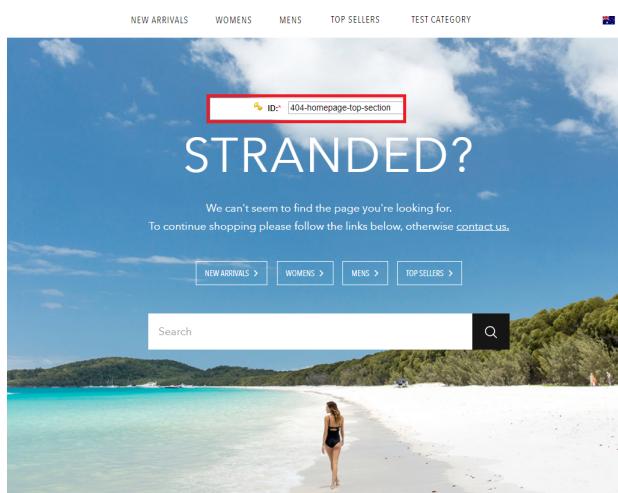
Email components screenshots:

email_components.zip

Error / Not Found[404]/ Forbidden/Offline Pages

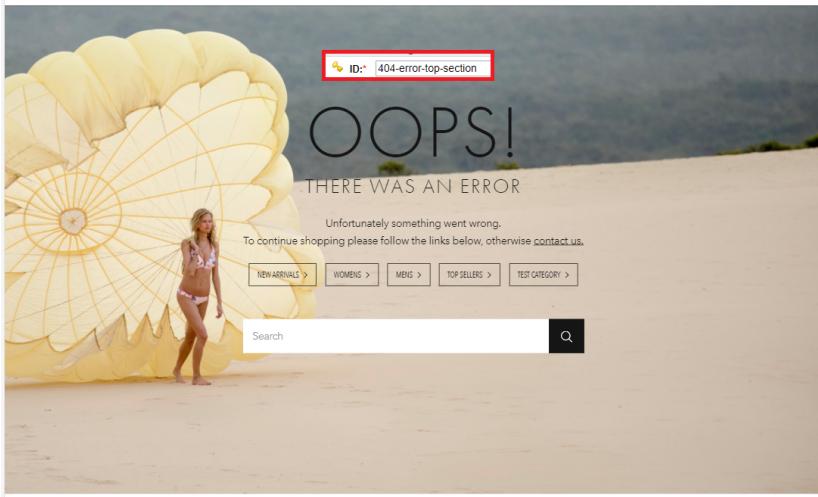
404 Page :

Content Asset ID : aid="404-homepage-top-section"



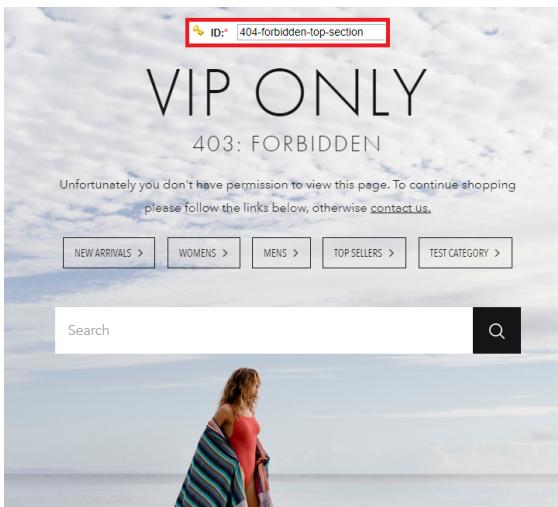
Error Page:

Content Asset ID : aid="404-error-top-section"



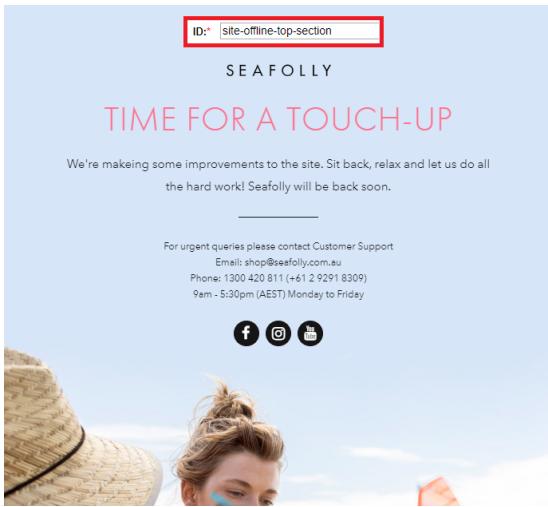
Forbidden Page:

Content Asset ID : aid="404-forbidden-top-section"



Site Offline Page:

Content Asset ID : aid="site-offline-top-section"



Fonts

Scripts from Adobe TypeKit

Code to go near the top of the <head> tag. (additional info attached)

Sunburn

Proxima Nova Default =====

```
<script src="https://use.typekit.net/ceu3cvj.js"></script>
<script>try{Typekit.load({ async: true });}catch(e){}</script>
```

Proxima Nova Advanced =====

```
<script>
(function(d) {
  var config = {
    kitId: 'ceu3cvj',
    scriptTimeout: 3000,
    async: true
  },
  h=d.documentElement,t=setTimeout(function(){h.className=h.className.replace(/\bwf-loading\b/g,"")+
wf-inactive"},config.scriptTimeout),tk=d.createElement("script"),f=false,s=d.getElementsByTagName("script")[0],a=h.className+=
"wf-loading";tk.src='https://use.typekit.net/'+config.kitId+'.js';tk.async=true;tk.onload=tk.onreadystatechange=function(){a=this.readyState;if(f||a&&a
!="complete"&&a!="loaded")return;f=true;clearTimeout(t);try{Typekit.load(config)}catch(e){};s.parentNode.insertBefore(tk,s)
})(document);
</script>
```

The advanced embed code loads asynchronously. It provides better control over rendering, but it requires custom CSS to hide the flash of unstyled text.

Seafolly

Futura PT Default =====

```
<script src="https://use.typekit.net/tss5fnp.js"></script>
<script>try{Typekit.load({ async: true });}catch(e){}</script>
```

Futura PT Advanced =====

```
<script>
(function(d) {
  var config = {
    kitId: 'tss5fnp',
    scriptTimeout: 3000,
    async: true
  },
  h=d.documentElement,t=setTimeout(function(){h.className=h.className.replace(/\bwf-loading\b/g,"")+
wf-inactive"},config.scriptTimeout),tk=d.createElement("script"),f=false,s=d.getElementsByTagName("script")[0],a=h.className+=
"wf-loading";tk.src='https://use.typekit.net/'+config.kitId+'.js';tk.async=true;tk.onload=tk.onreadystatechange=function(){a=this.readyState;if(f||a&&a
!="complete"&&a!="loaded")return;f=true;clearTimeout(t);try{Typekit.load(config)}catch(e){};s.parentNode.insertBefore(tk,s)
})(document);
</script>
```

Images for these scripts:

Using font family names in your CSS

In addition to using the Selectors setting to tell Typekit where to apply this font, you can use the font directly in your CSS rules with the following `font-family` values:

```
"proxima-nova"
```

Here's an example:

```
h1 {  
  font-family: "proxima-nova";  
}
```

You can also include a fallback font for older browsers by listing it after the Typekit `font-family` values. Here's an example of that:

```
h1 {  
  font-family: "proxima-nova", sans-serif;  
}
```

Using font family names in your CSS

In addition to using the Selectors setting to tell Typekit where to apply this font, you can use the font directly in your CSS rules with the following `font-family` values:

```
"futura-pt"
```

Here's an example:

```
h1 {  
  font-family: "futura-pt";  
}
```

You can also include a fallback font for older browsers by listing it after the Typekit `font-family` values. Here's an example of that:

```
h1 {  
  font-family: "futura-pt", sans-serif;  
}
```

Using weights & styles in your CSS

To access specific weights & styles, or *variations*, of this font in your CSS, use these values for `font-family`, `font-weight`, and `font-style`:

Light	"futura-pt"	300	normal	
Light Italic	"futura-pt"	300	italic	
Book	"futura-pt"	400	normal	
Book Italic	"futura-pt"	400	italic	
Medium	"futura-pt"	500	normal	
Medium Italic	"futura-pt"	500	italic	
Demi	"futura-pt"	600	normal	
Demi Italic	"futura-pt"	600	italic	
Heavy	"futura-pt"	700	normal	
Heavy Italic	"futura-pt"	700	italic	

Show variation-specific font-family names. [When are these necessary?](#)

Using weights & styles in your CSS

To access specific weights & styles, or *variations*, of this font in your CSS, use these values for `font-family`, `font-weight`, and `font-style`:

Thin	"proxima-nova"	100	normal	
Thin Italic	"proxima-nova"	100	italic	
Light	"proxima-nova"	300	normal	
Light Italic	"proxima-nova"	300	italic	
Regular	"proxima-nova"	400	normal	
Regular Italic	"proxima-nova"	400	italic	
Medium	"proxima-nova"	500	normal	
Medium Italic	"proxima-nova"	500	italic	
Semibold	"proxima-nova"	600	normal	
Semibold Italic	"proxima-nova"	600	italic	

Show variation-specific font-family names. [When are these necessary?](#)

Using weights & styles in your CSS

To access specific weights & styles, or *variations*, of this font in your CSS, use these values for `font-family`, `font-weight`, and `font-style`:

Medium	"proxima-nova"	500	normal	
Medium Italic	"proxima-nova"	500	italic	
Semibold	"proxima-nova"	600	normal	
Semibold Italic	"proxima-nova"	600	italic	
Bold	"proxima-nova"	700	normal	
Bold Italic	"proxima-nova"	700	italic	
Extrabold	"proxima-nova"	800	normal	
Extrabold Italic	"proxima-nova"	800	italic	
Black	"proxima-nova"	900	normal	
Black Italic	"proxima-nova"	900	italic	

Show variation-specific font-family names. [When are these necessary?](#)

Font files:

[Fonts.zip](#)

JIRA related: <https://jira.tryzens.com/browse/SFBP-68>

Fonts names:

1. Installed in DW:

```

//Avenir
font-family: 'Avenir';
font-family: 'AvenirNext';
font-family: 'AvenirNext-Condensed';
font-family: 'Linotype - AvenirLTStd-Heavy';
font-family: 'Linotype - AvenirLTStd-HeavyOblique';
font-family: 'Linotype - AvenirLTStd-Light';
font-family: 'Linotype - AvenirLTStd-LightOblique';
font-family: 'Linotype - AvenirLTStd-Oblique';
font-family: 'Linotype - AvenirLTStd-Roman';
font-family: 'AvenirNext-Regular';
font-family: 'AvenirNext-Bold';
font-family: 'AvenirNextCondensed-Regular';
font-family: 'AvenirNextCondensed-Medium';
font-family: 'Avenir-Book';
font-family: 'AvenirNext-DemiBold';

```

```

//Futura
font-family: 'FuturaStd-Book';
font-family: 'FuturaStd-Light';
font-family: 'FuturaStd-Medium';

```

```

//FontAwesome
font-family: 'FontAwesome';

```

2. Coming from script

```

//Futura
font-family: 'Futura-PT';

//Proxima
font-family: 'Proxima-Nova';

```

Font "LucidaGrande" must be replaced with "Proxima Nova" for Sunburn and "Futura PT" for Seafolly.

Getting started

Here are some tips to get you started. You can edit this page to see how it works!

1. Create a page

- Click "Create" and select "Blank Page" to create your first page.
- New pages are created as children of the page you are currently viewing.

2. Add to your page

- Click "Edit" to enter the Confluence editor and use the page layouts feature to structure your content using sections and columns.
- Use headings to format your text and drag and drop images into your page to provide visual interest.
- Click "Insert" and select "Other Macros" to add macros for navigation, special formatting and other media.

3. Organise your pages

Here are some tips for organising your content.

- Change the page order

The sidebar on the left displays your pages in a hierarchy. If you have Space Administrator permissions you can click "Space Tools" > "Reorder Pages" to move pages around.

On this page:

- 1. Create a page
- 2. Add to your page
- 3. Organise your pages
 - Change the page order
 - Add labels
 - Make templates

- **Add labels**

Labels help keep pages organised and make it easier for you to find the information you need. Click "Labels" at the bottom of a page to add or edit. The "Related pages" section on this page uses labels too!

- **Make templates**

Standardise and speed up the page creation process with templates. You can create and format a template with page layouts, standard headings and instructional text for hints and guidelines. Check out our sample page on "[Making a template](#)"

Related pages

- Getting started
- Making a template

Making a template

Give your authors a helping hand by using templates in your documentation space. You will need Space Administrator permissions to create templates.

To create a template:

1. Go to "Space Tools" in the sidebar, select "Content Tools" and create a new template.
2. Click "Page Layout" and add sections and columns to your page.
3. Add headings and sub-headings as needed.
4. Choose "Instructional Text" from the "Template" menu and add text that is only visible in the editor.
5. Save your template.

Your template will become available in the Create dialog for this space.

Related pages

- Getting started
- Making a template

Useful hint

Confluence Administrators can also make templates that are available across your whole Confluence instance.

Gift vouchers and Gift cards

Free Gift Voucher

1. Select the custom preference for enabling the promotion in Merchant Tools->Site Preferences-> Seafolly Integrations - > **Enable free gift voucher feature**
2. Set amount for the gift voucher which will be issued to the client: Merchant Tools->Site Preferences-> Seafolly Integrations - > **Free gift voucher Amount**

The default amount is 50.

Salesforce Development - develop... Sea folly Australia ▾ Merchant Tools Administration Storefront Toolkit Beta Cache Status OFF

Last Customer Export	5/12/2018 2:24 am	
Last Customer Import	1/1/2017 11:00 am	
Gift Card X-API-Version	1.0	
Gift Card X-Auth Header	bb8c1d27-8b00-461b-b6ba-e2346bc890cc	
Order Settlement Time	86,400,000	Defines how much time in milliseconds must be kept an OSU file in the process...
Customer Services email address	Iliyana.Mihaylova@tryzens.com	Email address which will be used for notifications from the OSU flow. If the ref...
Check free gift voucher feature is exist or not		Yes <input type="button" value="▼"/> No
Free Gift Voucher Amount		50.0

3. Create coupon code and promotion in BM.
4. Create campaign with ID "**freeGiftVoucher**".
5. Apply the coupon code assigned to the campaign to the Basket.
6. If the coupon is eligible and is successfully added to the Basket the free gift voucher will be created automatically.
7. The promotion could be from "Order" level, with "Amount of Qualifying Products", THE WANTED AMOUNT TO BE SPENT, discount 0.01.
8. When user checkouts, he will pay 0 for the voucher, so its value will be displayed as "FREE" in his order confirmation email.
9. The promotions adjustments will be removed from the Basket, only the free voucher will be a bonus.
10. Subject could be populated into the Promotion Callout Message field and Message could be populated into the Promotion Details field.

(For gift vouchers or coupons), you must assign the promotion to a campaign or A/B test. You must do this from within the specific campaign or A/B test. You can see where the promotion is scheduled in the **Schedule/Qualifiers/Compatibility** "Promotion Rule" to save the data for that section.

Select Language: Default

ID: [*]	FreeGVPromo
Name:	FreeGVPromo
Enabled:	Yes
Archived:	No
Searchable:	Yes
Callout Message:	Your \$50 Gift Voucher
Promotion Details:	With love, Seafolly xx
Tags:	<input type="text"/> Add New Tag: <input type="text"/> <input type="button" value="Add"/>
Image:	<input type="file"/>
Compatibility	
Exclusivity:	<input type="button" value="GLOBAL (Cannot be combined with any promotion)"/>



11. After the order is placed with that voucher, it will be updated in the order and its value from 0 will be changed to the value from the custom preference **Free gift voucher Amount**.

12. Same value will be removed from the order with Price Adjustment with promo ID "**dummyFreeGiftVoucherPromotion**".

13. GiftCertificateLineItems sold for free have boolean custom attribute which marks them as "**Free gift certificate from promotion**". Custom attribute ID "**isFreeGiftCertificate**".

Gift voucher purchase page

Pipeline GiftCert-Purchase:

https://demo-ap01-seafolly.demandware.net/on/demandware.store/Sites-Sunburn-Site/en_AU/GiftCert-Purchase

Here we are using 3 content assets.

Gift Image section is managed by content asset ID : **gift-details**

Giftcard description asset ID : **gift-voucher-description**

Giftcard delivery & description asset ID : **returns**

We are using foundation **interchange** plugin for manage the images for mobile, tab and desktop [source : <https://foundation.zurb.com/sites/docs/interchange.html>]

Demo Code : ``

GIFT CARD

\$50 \$100 \$200 \$600

Friend's Name *

Friend's Email Address *

Subject *

Message

You have 250 characters left out of 250

My Name *

ADD TO CART

DESCRIPTION **ID: returns**

DELIVERY & RETURNS **ID: gift-voucher-description**

Interchange **JavaScript**
Interchange uses media queries to dynamically load responsive content that is appropriate for the user's device.

Prices for e-gift vouchers

We add radio buttons with prices so Customer can choose amount for e-gift voucher. We hide amount input field to prevent Customers to enter the values in it.

Configuration

1. Open BM
2. Select Merchant Tools
3. Select Site Preferences
4. Select Custom Preferences
5. Then select Seafolly Configs, this will open all preferences for Seafolly project
6. Find Prices for e-gift vouchers
7. To add prices for vouchers to all sites, enter the desired value in value field and click "Add" to enter it. To add a new value follow the same procedure until you enter all prices for vouchers.
Example: For a \$ 50 voucher, enter 50 in value field.
8. If you want to set a different prices for different sites, click "Edit Across Sites" link. This will open a new window and from it you can set vouchers prices for different sites.

Preference Id

The id of this preference is radioButtonGiftCartPrice. You can find it on the follow path Administration > Site Development > System Object Types > SitePreferences

Single page checkout using a Gift Voucher

Meta Data:

The following custom attributes are added to:

1. GiftCertificate system object
ID: giftCertificatePin
Type: String

2. PaymentTransaction system object

ID: giftCardValidationID

Type: String

Site Preferences:

ID	Name	Type	Value
giftCardXApiVersion	Gift Card X-API-Version	String	DEV: 1.0 PROD: 1.0
giftCardXAuth	Gift Card X-Auth Header	String	DEV: bb8c1d27-8b00-461b-b6ba-e2346bc890cc PROD: fd45616f-172f-4ffc-9c7b-3baf70649314

**It is recommended to get and install Insomnia app from <https://insomnia.rest/>
Use this application to check AP21 services behavior and responses.**

Workflow

1. Customer selects Gift Card as a payment method.
2. Customer adds Card number and pin.
3. Web service call to AP21 is executed in order to check if gift card is valid.

Sample request:

request method: GET
request URL: <https://devflow.seafolly.com.au/giftcard/9999999968/1816>
query: ?site-id=SeafollyAU

header:

Name	Value
1. X-Auth	bb8c1d27-8b00-461b-b6ba-e2346bc890cc
2. X-API-Version	1.0
3. Accept	application/json
4. Content-Type	application/json

4. If the gift card is valid we add it to the basket. Otherwise, the relevant error message is displayed on the storefront.
5. AP21 confirms the voucher is valid:
 - 5.1 If there are sufficient funds on the gift voucher to cover the total order value, the customer is allowed to select [Place Order] CTA button. Web service call to AP21 is executed for the reservation of the gift voucher.

Example request:

request method: POST
request URL: <https://devflow.seafolly.com.au/giftcard/9999999968/1816>
query: ?site-id=SeafollyAU
request Body: { "amount": 1 }

header:

Name	Value
1. X-Auth	bb8c1d27-8b00-461b-b6ba-e2346bc890cc
2. X-API-Version	1.0
3. Accept	application/json
4. Content-Type	application/json

- a) If the gift voucher payment capture process is successful, then SCC surfaces an Order Confirmation page.
- b) If the gift voucher payment capture process is unsuccessful, then SCC surfaces a failed payment message on the Checkout page; the customer will then reattempt the payment or choose a different payment method.

- 5.2 If there are insufficient funds on the gift voucher to cover the total order value, the order total balance is recalculated. Then the customer

should select a secondary payment method.

Then a web service call to AP21 is executed for the reservation of the gift voucher.

a) If the gift voucher reservation process is successful, then payment from the secondary payment method is captured.

1a) If the secondary payment method is successful, then SCC surfaces an Order Confirmation page.

2a) If the secondary payment method is unsuccessful, then error message is displayed on the Checkout page. The gift voucher is only reserved for certain time and its balance remains the same.

(QA Team, keep in mind that when you place order using gift voucher, it will be lock for one hour)

b) If the gift voucher reservation process is unsuccessful, then SCC surfaces a failed payment message on the Checkout page; the customer will then reattempt the payment or choose a different payment method.

The capture payment process for the secondary payment method is not executed.

[Sampe gift vouchers - GV.xlsx](#)

Google Tag Manager (GTM)

Google Analytics & Tag Manager Accounts

For this integration we need to have access to the Analytics and Tag Manager consoles.

Login URL	Email	Pass
https://analytics.google.com	antonia.dimitrova@tryzens.com	London@2020!
https://tagmanager.google.com	antonia.dimitrova@tryzens.com	London@2020!

For testing purposes there is one published GTM container (dev03-ap01-seafolly.demandware.net) that is linked to the Analytics Dev03-SF testing property.

The linking is done when creating **new tag of type Universal Analytics** and adding **variable of type Google Analytics Settings with the tracking ID (tag-variable.png)**.

Account	Property	Analytics Tracking ID	GTM ID
antonia.dimitrova@tryzens.com	Dev03-SF	UA-111480338-1	GTM-PDR74CQ
	DO NOT USE - Old Shopify E-Comms	UA-11797790-1	no access to GTM

The **Analytics Tracking ID and global tag site snippet** can be found when Selecting the desired property > Admin > Tracking Info > Tracking code.

The **GTM ID and snippet** can be found when Selecting the desired container > Admin > Install Google Tag Manger.

Business Manager

Custom preferences and settings

The integration uses one custom group **GTM_SETTINGS** with several settings, that are described bellow.

Currently all sites use the same settings.

Preference ID	Preference Name	Preference Values			
		Seafolly US	Seafolly AU	Seafolly SG	Sunburn
googleTagManagerEnabled	GTM Enabled	true	true	true	true

googleTagManagerSnippet	GTM Snippet	<!-- Google Tag Manager --> <script>(function(w,d,s,l,i){w[l]=w[l] [];w[l].push({'gtm.start': new Date().getTime(),event:'gtm.js'});var f=d.createElement(s),dl=l!='dataLayer'?&l=+l:j;f.async=true;j.src='https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);})(window,document,'script','dataLayer','GTM-PDR74CQ'); <!-- End Google Tag Manager -->
googleTagManagerSnippetNoScript	GTM NoScript Snippet	<!-- Google Tag Manager (noscript) --> <noscript><iframe src="https://www.googletagmanager.com/ns.html?id=GTM-PDR74CQ" height="0" width="0" style="display:none;visibility:hidden"></iframe></noscript> <!-- End Google Tag Manager (noscript) -->

Promotion listing and clicks

Promotions are the one component of the GTM that must be added by the user in the BM. For that you must find the content asset you want to edit and add the following event (**in bold**) for the items you want to track:

```
<a class="btb btn-primary" onclick="onPromoClick('modern1', 'Modern Geometry', 'modern_geometry')" href="www.google.bg">Shop Now</a>
```

The event has the following structure **onclick="onPromoClick('\${ promotion id }', '\${ promotion name }', '\${ creative }')"**

Adding this code allows the GTM to track the promotion listing and clicks on every page you want.

When listing the promotions on a page, if you have one promotion added to multiple items on the page - the name will be checked and it will be listed only once and given one position. But clicking on those promotion items will be triggered and sent to the GTM every time.

Testing

For testing purposes add the Chrome extension for DataLayerDoctor : <https://chrome.google.com/webstore/detail/data-layer-doctor/pogogcdfpladlIndlcdmedkapcekmjm?hl=en-GB>

The credentials for the extension are:

- user: ian@ianlyons.com
- pass: 3c&*1WSt

When testing on some pages (such as order confirmation, checkout start) some variables defined in several sections get mixed up and not validated correctly. See attached [DataLayerDoctorTesting.docx](#) for example on this.

Other useful testing tools are:

- Tag Assistant by Google - https://chrome.google.com/webstore/detail/tag-assistant-by-google/kejbdbjndbnbjgmefkgdddjlbkphdefk?utm_source=chrome-app-launcher-info-dialog
- DataSlayer - https://chrome.google.com/webstore/detail/dataslayer/ikbablmmjldhamhcldjigniffkkigpo?utm_source=chrome-app-launcher-info-dialog

Useful Links

Description	Link
Enhanced Ecommerce (UA) Developer Guide for GTM & Data Layer	https://developers.google.com/tag-manager/enhanced-ecommerce
Enhanced Ecommerce (Standard)	https://developers.google.com/analytics/devguides/collection/analyticsjs/enhanced-ecommerce
Enhanced Ecommerce Demo	https://ga-dev-tools.appspot.com/enhanced-ecommerce/ https://enhancedecommerce.appspot.com/

Header and Footer

Mailing list subscription

- Overview
- Requirements
- Rules
- Validations
- Scenarios
 - Main
 - Alternative
 - Guest Checkout
 - Guest Checkout (Alternative)
- Library
 - Content Asset
- Meta Data
 - System Object
 - Custom Object

Overview

This functionality enables newsletter subscription for customers (guests and registered ones) from site's footer as well as from My Account > Personal Data, during Guest Checkouts and Cart Order Summary Registration.

Functional specifications agreed with Fusion Factory:

[Mailing list subscription and Customer data flows for Seafolly.docx](#)

Requirements

Please add **int_seafolly** to cartridge path for all sites including Business Manager.

Integration Framework Cartridge

This cartridge requires **bc_integrationframework** cartridge to be present in cartridge path for all sites including Business Manager.

Customer Export

[Seafolly Integrations > Customers Export](#) depends on this module.

Rules

1. Customer email address field is a mandatory field
2. The 3 DOB fields are non-mandatory
 - a. A customer can fill in any combination of these (i.e., DD + MM, or DD + MM +YYYY)

Validations

Fields are validated on both sides - front-end and back-end. Back-end validations must go through NewsletterModel. Both validations uses RegExp and patterns are added as form resource (resources/forms.properties) and is slash-escaped due format requirements and are exported via Resource.ds

Scenarios

Main

This scenario creates Custom Object.

1. A Customer browses the site, and selects the footer-based subscription field
2. The Customer enter their email address & DOB
3. The Customer submits the data via a "Sign Up" CTA button
 - a. SCC surfaces a success message if the email address has not be used to subscribe previously - "Thank you for subscribing"
 - b. SCC surfaces an error message if the email address has been used to subscribe previously - "This email address has already subscribed"
4. SCC exports the Customer's email address & mailing list subscription preferences to Flow middleware
 - a. Flow middleware will manage sending this data to the ESP (Campaign Monitor)

Alternative

This scenario **doesn't** create Custom Object. It just stores subscription state in Profile custom attribute `isSubscribed`.

1. A Customer browses the site, and navigates to My Account
2. If the Customer already has an account, they will log in, and navigate to the 'Personal Details' section
 - a. The Customer can update their mailing list subscription preference here by ticking or unticking the mailing list subscription checkbox
 - b. This is as per the OOTB SiteGenesis functionality
3. If the Customer does not have an account, they can create a new account
 - a. The Customer can specify their mailing list preference during sign up ticking or unticking the mailing list subscription checkbox
 - b. The default state of this checkbox will be unticked
 - c. This is as per the OOTB SiteGenesis functionality
4. SCC will export the Customer's email address & mailing list subscription preferences to Flow middleware
 - a. Flow middleware will manage sending this data to the ESP (Campaign Monitor)

Guest Checkout

This scenario creates Custom Object if customer is checking out as a guest, ticks mailing list subscription checkbox and submits the form on SingleShipping page. Also registration page subscription tick is ticked.

1. A Customer browses the site and add items to their bag
2. The Customer proceeds to checkout and opts to use the Guest Checkout functionality
3. SCC surfaces a mailing list subscription checkbox
 - a. The Customer can specify their mailing list preference by ticking or unticking the mailing list subscription checkbox
4. The Customer completes checkout
5. If the Customer chose to opt in for the mailing list, SCC will export the Customer's email address & mailing list subscription preferences to Flow middleware
 - a. Flow middleware will manage sending this data to the ESP (Campaign Monitor)

Guest Checkout (Alternative)

This scenario creates Custom Object.

1. A Customer browses the site and add items to their bag
2. The Customer proceeds to checkout
3. The Customer successfully completes checkout
4. SCC surfaces an Order Confirmation page with the common footer
5. If the Customer chose to opt in for the mailing list, please refer to Scenarios > Main
6. If the Customer chose not to opt in for the mailing list, they can browse or exit the site

Library

Content Asset

Please include/replace the form inside asset **footer-social-email** (Footer Social and Email).

This should be done as Shared Library.

Newsletter subscription form

```
<h3>Be the First to Know</h3>
<p>Receive the latest Seafolly news and get $20 off in your birthday
month*</p>
<form action="#" method="post" class="newsletter-form">
    <input type="text" name="email" class="input-text" placeholder="Enter
Your Email" value="" />
    <input type="text" name="date_of_birth" class="input-text"
placeholder="DD / MM / YYYY" value="" />
    <button type="submit" value="Submit" class="fa fa-envelope"><span
class="visually-hidden">Submit</span></button>
    <div class="message"></div>
</form>
```

Meta Data

System Object

Profile System Object Meta Data

```
<type-extension type-id="Profile">
    ...
    <custom-attribute-definitions>
        ...
        <attribute-definition attribute-id="isSubscribed">
            <display-name xml:lang="x-default">Newsletter
subscription</display-name>
            <type>boolean</type>
            <mandatory-flag>false</mandatory-flag>
            <externally-managed-flag>false</externally-managed-flag>
        </attribute-definition>
        ...
    </custom-attribute-definitions>
    <group-definitions>
        <attribute-group group-id="Standard">
            ....
            <attribute attribute-id="isSubscribed" system="false" />
        </attribute-group>
        ...
    </group-definitions>
    ...
</type-extension>
```

Custom Object

EmailSubscriptionObject Meta Data

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://www.demandware.com/xml/impe.../2006-10-31">
    ...
    <custom-type type-id="EmailSubscriptionObject">
        <display-name xml:lang="x-default">Newsletter Subscription Object</display-name>
        <description xml:lang="x-default">Newsletter Subscription Object</description>
        <staging-mode>no-staging</staging-mode>
        <storage-scope>site</storage-scope>
        <key-definition attribute-id="emailID">
            <type>string</type>
            <min-length>0</min-length>
        </key-definition>
        <attribute-definitions>
            <attribute-definition attribute-id="DateOfBirth">
                <display-name xml:lang="x-default">Date of Birth</display-name>
                <description xml:lang="x-default">Date of Birth</description>
                <type>string</type>
                <localizable-flag>false</localizable-flag>
                <mandatory-flag>false</mandatory-flag>
                <externally-managed-flag>false</externally-managed-flag>
                <min-length>0</min-length>
            </attribute-definition>
            <attribute-definition attribute-id="SiteID">
                <display-name xml:lang="x-default">Site ID</display-name>
                <type>string</type>
                <localizable-flag>false</localizable-flag>
                <mandatory-flag>false</mandatory-flag>
                <externally-managed-flag>false</externally-managed-flag>
                <min-length>0</min-length>
            </attribute-definition>
        </attribute-definitions>
        <group-definitions>
            <attribute-group group-id="NewsletterSubscriptions">
                <display-name xml:lang="x-default">Newsletter Subscriptions</display-name>
                <attribute attribute-id="emailID"/>
                <attribute attribute-id="DateOfBirth"/>
            </attribute-group>
        </group-definitions>
    </custom-type>
    ...
</metadata>
```

Content assets and content slots

Mini-bag

To display custom messages we use content slot. This slot have id "minicart-banner".

Classes mini-bag-promo-content and mini-bag-promo-subcontent helps for content styling.

```
<div class="mini-bag-promo-content">Save 20% on all orders over $ 200</div> <div class="mini-bag-promo-subcontent">Offer valid until 12am16/12/2017</div>
```

Basket page

To display payment methods in basket page we create one content asset for this task. This is asset with id "cart-payment-methods". It will be displayed after payment methods.

My Account

- [Overview](#)
- [Pages / Functionalities](#)

Overview

Pages / Functionalities

- [Customer migration - reset password flow](#)
- [My Account Pages](#)

Customer migration - reset password flow

- [Overview](#)
- [Rules](#)
- [Validation](#)
- [Scenario](#)
- [Meta Data](#)
 - [System Objects](#)
 - [Site Preferences](#)
 - [Customer Groups](#)

Overview

When a migrated (from Magento) Customer browses the site, navigates to the 'My Account' landing page and attempts to log in to their account but fails with invalid password, he gets a message informing him that he needs to reset his password.

Rules

The re-set password message is only surfaced on the My Account log-in page if a migrated, registered customer tries to log in to their account and has not re-set their password. Once this type of Customer has re-set their password, this message should not be surfaced again.

Validation

When invalid password is detected along with first time customer (after import), script checks if customers email is equal to his login and if not, sets email to be equal to the login. This is required because password reset procedure handles customer login but sends data to customer email.

Scenario

1. A migrated, registered Customer browses the new site
2. The Customer navigates to the My Account landing page and attempts to log-in to their account for the first time on the new site
 - a. The Customer enters their email address and password (from the old site)
3. The password will be auto-generated so the customer won't know it.
4. If **lastLoginTime** of the customer is null and he's a member of **Migrated** customer group, SCC surfaces a message to advise that this a new a site, and password re-set is required
 - a. In case customer's **Creation Date** is after site's preference **lastMigrationDate** then he gets a different message without changing the procedure.
5. After customer hit re-set password button he gets a lightbox message to advise the customer an email has been sent to them and the Customer follows the re-set password flow i.e. the same process as the OOTB "Forgot Password" flow for account log-in.
 - a. Customer are unassigned from **Migrated** group
 - b. The Customer checks their email account and follows the instruction in the email to re-set their password
 - c. Customer creates new password
 - d. Customer re-attempts log-in to the new site using the new password

In cases where customer successfully log-in or hit re-set password button he won't get Migrated customer page.

Meta Data

System Objects

System Object Meta

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://www.demandware.com/xml/impe.../2006-10-31">
    ...
    <type-extension type-id="SitePreferences">
        <custom-attribute-definitions>
            ...
            <attribute-definition attribute-id="lastMigrationDate">
                <display-name xml:lang="x-default">Last Customer Import</display-name>
                <description xml:lang="x-default">When a migrated customer tries to log-in, he's prompted to reset his password. If he was migrated before this date, he'll get a message that he's migrated to our new site otherwise it will be another one.</description>
                <type>datetime</type>
                <mandatory-flag>false</mandatory-flag>
                <externally-managed-flag>false</externally-managed-flag>
            </attribute-definition>
            ...
        </custom-attribute-definitions>
        <group-definitions>
            ...
            <attribute-group group-id="Integration_Seafolly">
                <display-name xml:lang="x-default">Integration Seafolly</display-name>
                ...
                <attribute attribute-id="lastMigrationDate"/>
                ...
            </attribute-group>
            ...
        </group-definitions>
    </type-extension>
    ...
</metadata>
```

Site Preferences

Apply this for SeafollyAU, SeafollySG, SeafollyUS and Sunburn sites!

Preference FOR EACH SITE

```
<?xml version="1.0" encoding="UTF-8"?>
<preferences
xmlns="http://www.demandware.com/xml/impex/preferences/2007-03-31">
...
<custom-preferences>
...
<development>
....
<preference
preference-id="lastMigrationDate">2017-12-31T00:00:00.000+0000</preference>
....
</development>
...
</custom-preferences>
</preferences>
```

Customer Groups

Customer Group

```
<?xml version="1.0" encoding="UTF-8"?>
<customer-groups
xmlns="http://www.demandware.com/xml/impex/customergroup/2007-06-30">
...
<customer-group group-id="Migrated">
    <description>Represents all customers which were migrated before
site launch.</description>
    <custom-attributes/>
</customer-group>
...
</customer-groups>
```

My Account Pages

Content assets and content slots

MY DASHBOARD SIGN IN

content asset ID : create-account-benefits

MY DASHBOARD SIGN IN

RETURNING CUSTOMERS

If you are a registered user, please enter your email and password.

Email Address *

Password *

Remember Me

[Forgot Password](#)

SIGN IN >

NEW CUSTOMERS

Creating an account is easy. Just fill out the form below and enjoy the benefits of being a registered customer.

Subscribe and get a \$20 voucher on your birthday month

Receive news about exclusive offers and products

Save your order history

Speed up the checkout by storing billing & shipping info.

CREATE AN ACCOUNT >

MY DASHBOARD

content asset ID : account-landing

content asset ID : account-nav-registered

MY ACCOUNT

Personal Details

Addresses

Orders

Wish List

asset ID : account-nav-registered

MY DASHBOARD

Dinu SK | [Sign out](#)

content asset ID : account-landing

ACCOUNT DETAILS
Show or update personal information

ORDERS
Check order status or see order history

WISH LIST
View items on your list or invite friends

ADDRESSES
Manage billing and shipping addresses

Style sheets used for Account pages

_login.scss, _account.scss, _orders.scss, _base_tooltip.scss

Non-transactional site support - Maaji

Overview

nonTransactionalSite custom preference in BM which enables, the following sections will hide from storefront

- Prices
- Add to bag, My Account, Mini cart from header
- Wish list
- Promotional message
- Promotion

Business Manager

Custom preferences and settings

Merchant Tools->Site Preferences->Storefront Configurations->

Currently Maaji site use the settings.

Preference ID	Preference Name	Preference Values	
		Maaji	MaajiCO
nonTransactionalSite	Non Transnational Site Support	true	false

General

Meta Data

System Objecttype Extensions

```

<attribute-definition attribute-id="nonTransactionalSite">
<display-name xml:lang="x-default">Non Transnational Site Support</display-name>
<description xml:lang="x-default">Non Transnational Site Support</description>
<type>boolean</type>
<mandatory-flag>false</mandatory-flag>
<externally-managed-flag>false</externally-managed-flag>
<default-value>false</default-value>
</attribute-definition>

<attribute-group group-id="Seafolly Configs">
<attribute attribute-id="nonTransactionalSite" />
</attribute-group>

```

Order Status Updates

Overview

This module processes xml files with valid order status updates structure and sends transactional emails to customer.

What is expected to happen in real is:

1. Customer places order on SFCC storefront;
2. The order is exported to AP21 through Flow;
3. The order is processed by the warehouse;
 - a. If all the items could be shipped, the order is fulfilled and status "Order Shipped" is send for the order in the OSU file;
 - b. In case if all the items could NOT be shipped or the customer cancel it through the Customer Services, status "Order Cancelled" is send for the order in the OSU file and whole amount of the order is refunded;
 - c. In case if one or more items are cancelled, OSU file for combined product statuses "Shipped" and "Cancelled" must be sent. Status "Order Cancelled" is send for the order and the amount for the products and their shipping is refunded.
4. If items were shipped to the customer, they could be returned. In this case, the item is returned to the warehouse, the status must be updated in AP21 and OSU file only with returned items to be sent to SFCC. Status "Order Returned" is send for the order and the amount for the products and their shipping is refunded.

Requirements

No additional requirements.

Scenario

Job process all files in **FolderPath** matching **FilePattern**. Each of the files is processed, extracting voucher data into useful objects. Then an email is sent to the customer for each of the vouchers.

If all vouchers are valid and there are no validation errors, current file is being deleted.

If there are some vouchers which are not valid, job process only valid ones and current file is being moved into **ErrorPath**.

If there's a fatal error during the job execution, no vouchers will be processed and current file is being moved into **ErrorPath**.

Job Step Parameters

1. **ImportPipeline**: OrderStatusUpdates-ImportXMLFromFile (this should not be changed)
2. **FolderPath**: Folder where your files was uploaded by FF e.g. /in/shipment/SiteID
3. **ErrorPath**: Folder where your files with error will be moved e.g. /in/shipment/SiteID/Errors

Example XML Data

Status Shipped:

[OrderStatusUpdates_Cancelled.xml](#)

Status Returned:

[OrderStatusUpdates_Returned.xml](#)

Status Cancelled:

[OrderStatusUpdates_Shipped.xml](#)

Status Cancelled and Shipped:

[OrderStatusUpdates_Shipped_and_Cancelled.xml](#)

Jobs

Jobs Meta Data

```
<job job-id="OrderStatusUpdateSeafollyAU" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="SeafollyAU"/>
        <step step-id="OSUTransferSeafollyAU" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">FileManager-TransferFromFtp</parameter>
                <parameter name="HostURL">52.62.59.176</parameter>
                <parameter name="UserID">seafolly</parameter>
                <parameter name="Password">W2dReth$</parameter>
                <parameter name="FilePattern">^.+\xml$</parameter>
                <parameter
name="SourceFolder">/ftp/seafolly/root/dw_staging/download/shipment/Seafol
lyAU</parameter>
                <parameter
name="TargetFolder">/src/in/shipments/SeafollyAU</parameter>
                <parameter name="SecureFtp">SFTP</parameter>
                <parameter name="Timeout">10000</parameter>
```

```

        <parameter name="DeleteFile">Delete</parameter>
    </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
<step step-id="OSUSeafollyAU" type="ExecutePipeline"
enforce-restart="false">
    <description/>
    <parameters>
        <parameter
name="ExecutePipeline.Pipeline">FileManager-ImportFiles</parameter>
        <parameter
name="ImportPipeline">OrderStatusUpdates-ImportXMLFromFile</parameter>
            <parameter name="FilePattern">^.+\.xml$</parameter>
            <parameter
name="FolderPath">/in/shipments/SeafollyAU</parameter>
            <parameter
name="ErrorPath">/in/shipments/SeafollyAU/errors</parameter>
            </parameters>
        <rules>
            <on-exit status="ERROR">
                <stop-job/>
            </on-exit>
        </rules>
    </step>
</flow>
<rules/>
<triggers>
    <run-recurring enabled="true">
        <recurrence>
            <date-from>2017-10-04Z</date-from>
            <start-time>01:42:41.000Z</start-time>
            <interval>15m</interval>
            <day-of-week>
                <weekday>Thursday</weekday>
                <weekday>Wednesday</weekday>
                <weekday>Friday</weekday>
                <weekday>Saturday</weekday>
                <weekday>Tuesday</weekday>
                <weekday>Sunday</weekday>
                <weekday>Monday</weekday>
            </day-of-week>
        </recurrence>
    </run-recurring>
</triggers>
</job>

<job job-id="OrderStatusUpdateSeafollySG" priority="0">
    <description/>
    <parameters/>

```

```

<flow>
    <context site-id="SeafollySG"/>
    <step step-id="OSUTransferSeafollySG" type="ExecutePipeline"
enforce-restart="false">
        <description/>
        <parameters>
            <parameter
name="ExecutePipeline.Pipeline">FileManager-TransferFromFtp</parameter>
                <parameter name="HostURL">52.62.59.176</parameter>
                <parameter name="UserID">seafolly</parameter>
                <parameter name="Password">W2dReth$</parameter>
                <parameter name="FilePattern">^.+\xml$</parameter>
                <parameter
name="SourceFolder">/ftp/seafolly/root/dw_staging/download/shipment/Seafol
lySG</parameter>
                <parameter
name="TargetFolder">/src/in/shipments/SeafollySG</parameter>
                    <parameter name="SecureFtp">SFTP</parameter>
                    <parameter name="Timeout">10000</parameter>
                    <parameter name="DeleteFile">Delete</parameter>
            </parameters>
            <rules>
                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
        <step step-id="OSUSeafollySG" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">FileManager-ImportFiles</parameter>
                    <parameter
name="ImportPipeline">OrderStatusUpdates-ImportXMLFromFile</parameter>
                        <parameter name="FilePattern">^.+\xml$</parameter>
                        <parameter
name="FolderPath">/in/shipments/SeafollySG</parameter>
                        <parameter
name="ErrorPath">/in/shipments/SeafollySG/errors</parameter>
                    </parameters>
                    <rules>
                        <on-exit status="ERROR">
                            <stop-job/>
                        </on-exit>
                    </rules>
            </step>
        </flow>
        <rules/>
        <triggers>
            <run-recurring enabled="true">
                <recurrence>
                    <date-from>2017-10-04Z</date-from>

```

```

        <start-time>01:44:41.000Z</start-time>
        <interval>15m</interval>
        <day-of-week>
            <weekday>Thursday</weekday>
            <weekday>Wednesday</weekday>
            <weekday>Friday</weekday>
            <weekday>Saturday</weekday>
            <weekday>Tuesday</weekday>
            <weekday>Sunday</weekday>
            <weekday>Monday</weekday>
        </day-of-week>
    </recurrence>
</run-recurring>
</triggers>
</job>

<job job-id="OrderStatusUpdateSeafollyUS" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="SeafollyUS"/>
        <step step-id="OSUTransferSeafollyUS" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">FileManager-TransferFromFtp</parameter>
                <parameter name="HostURL">52.62.59.176</parameter>
                <parameter name="UserID">seafolly</parameter>
                <parameter name="Password">W2dReth$</parameter>
                <parameter name="FilePattern">^.+\xml$</parameter>
                <parameter
name="SourceFolder">/ftp/seafolly/root/dw_staging/download/shipment/Seafol
lyUS</parameter>
                <parameter
name="TargetFolder">/src/in/shipments/SeafollyUS</parameter>
                <parameter name="SecureFtp">SFTP</parameter>
                <parameter name="Timeout">10000</parameter>
                <parameter name="DeleteFile">Delete</parameter>
            </parameters>
            <rules>
                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
        <step step-id="OSUSeafollyUS" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">FileManager-ImportFiles</parameter>
                <parameter

```

```

name="ImportPipeline">OrderStatusUpdates-ImportXMLFromFile</parameter>
    <parameter name="FilePattern">^.+\xml$</parameter>
    <parameter
name="FolderPath">/in/shipments/SeafollyUS</parameter>
    <parameter
name="ErrorPath">/in/shipments/SeafollyUS/errors</parameter>
    </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
</flow>
<rules/>
<triggers>
    <run-recurring enabled="true">
        <recurrence>
            <date-from>2017-10-04Z</date-from>
            <start-time>01:46:41.000Z</start-time>
            <interval>15m</interval>
            <day-of-week>
                <weekday>Thursday</weekday>
                <weekday>Wednesday</weekday>
                <weekday>Friday</weekday>
                <weekday>Saturday</weekday>
                <weekday>Tuesday</weekday>
                <weekday>Sunday</weekday>
                <weekday>Monday</weekday>
            </day-of-week>
        </recurrence>
    </run-recurring>
</triggers>
</job>

<job job-id="OrderStatusUpdateSunburn" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="Sunburn"/>
        <step step-id="OSUTransferSunburn" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">FileManager-TransferFromFtp</parameter>
                    <parameter name="HostURL">52.62.59.176</parameter>
                    <parameter name="UserID">seafolly</parameter>
                    <parameter name="Password">W2dReth$</parameter>
                    <parameter name="FilePattern">^.+\xml$</parameter>
                <parameter
name="SourceFolder">/ftp/seafolly/root/dw_staging/download/shipment/Sunbur
n</parameter>

```

```

        <parameter
name="TargetFolder">/src/in/shipments/Sunburn</parameter>
        <parameter name="SecureFtp">SFTP</parameter>
        <parameter name="Timeout">10000</parameter>
        <parameter name="DeleteFile">Delete</parameter>
    </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
<step step-id="OSUSunburn" type="ExecutePipeline"
enforce-restart="false">
    <description/>
    <parameters>
        <parameter
name="ExecutePipeline.Pipeline">FileManager-ImportFiles</parameter>
        <parameter
name="ImportPipeline">OrderStatusUpdates-ImportXMLFromFile</parameter>
            <parameter name="FilePattern">^.+\.xml$</parameter>
        <parameter
name="FolderPath">/in/shipments/Sunburn</parameter>
        <parameter
name="ErrorPath">/in/shipments/Sunburn/errors</parameter>
    </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
</flow>
<rules/>
<triggers>
    <run-recurring enabled="true">
        <recurrence>
            <date-from>2017-10-04Z</date-from>
            <start-time>01:48:41.000Z</start-time>
            <interval>15m</interval>
            <day-of-week>
                <weekday>Thursday</weekday>
                <weekday>Wednesday</weekday>
                <weekday>Friday</weekday>
                <weekday>Saturday</weekday>
                <weekday>Tuesday</weekday>
                <weekday>Sunday</weekday>
                <weekday>Monday</weekday>
            </day-of-week>
        </recurrence>
    </run-recurring>
</triggers>

```

```

        </run-recurring>
    </triggers>
</job>

```

Password Reset

- password-returning-message - for users created after migration
- password-migrated-message - for users created before migration
- password-hint - for the password hint
-  reset-password-message - text presented on the password reset page, above the form

PayPal Integration

The integrated version of PayPal is the Braintree cartridge is 17.1.1. according the documentation:

[Braintree_Integration_Guide_for_Salesforce_Commerce_Cloud.docx](#)

PayPal Developer & Sandbox Accounts

PayPal Developer Account

In order to integrate the PayPal functionality, a developer account must be created. You can create a new business account here <https://www.paypal.com/us/webapps/mpp/account-selection> or use the following credentials:

Login Url	Email	Password
https://developer.paypal.com/	antonia.dimitrova@tryzens.com	P99v2EYCne?Prcl

When you register for new account, PayPal automatically creates sandbox business (merchant) & personal (buyer) accounts for testing purposes.

For the Seafolly integration the following business & personal sandbox accounts are created, using **dummy Tryzens emails**:

Seafolly Site	Type	Merchant Email	Pass	API user	API pass	API signature
AU	business	proj.seafolly.au@tryzens.com	cra2yPuma65	proj.seafolly.au_api1.tryzens.com	BBGD3YFTJJ8KXQXT	AFcWxV21C7fd0v3bYYYRCpSSRI31A6Lww
SG	business	proj.seafolly.sg@tryzens.com	cra2yPuma65	proj.seafolly.sg_api1.tryzens.com	3GSJUKKGZQT6LLSU	AFcWxV21C7fd0v3bYYYRCpSSRI31AZFQy
US	business	proj.seafolly.us@tryzens.com	cra2yPuma65	proj.seafolly.us_api1.tryzens.com	YV8XV4TJEN2A5RQP	AFcWxV21C7fd0v3bYYYRCpSSRI31AqsBkC
Sunburn	business	proj.sunburn@tryzens.com	cra2yPuma65	proj.sunburn_api1.tryzens.com	B8BSJE5VSJ939YQS	AFcWxV21C7fd0v3bYYYRCpSSRI31AjuE2i1
All sites	personal	proj.seafolly.buyer@tryzens.com proj.seafolly.buyer3@tryzens.com	cra2yPuma65	-	-	-

New accounts can be created under the "**Accounts**" link in the left menu. See attached [pp-accounts-list.png](#) for current list.

PayPal Sandbox Accounts

After you have obtained the sandbox accounts, all transactions can be accessed on <https://www.sandbox.paypal.com> using the above business & personal accounts credentials.

For business accounts the Merchant ID provided in the above credentials can be found in **Profile** (from top navigation) > **Profile and Settings**. See attached [pp-business-profile.png](#) & [pp-business-merchant-id.png](#)

When creating credit cards in PalPay the test numbers can not be used. Use the following link to generate test numbers: <http://credit-card-genera>

tor.2-ee.com/

The buyer account has the following additional data:

- SSN number: 078-05-1120

Multiple Currencies

The account can be configured to accept multiple currencies. To do so:

- select "Currencies" in the account overview in the left panel. See [paypal-currencies.png](#)
- select the new currency from the drop down and click "Add currency". See [paypal-currencies-edit.png](#)

PayPal Applications

When using PayPal through the Braintree services a linked PayPal application must be setup in the developer console. There is option for mocked PayPal, though it does not work with the Braintree Cartridge.

The creation of new application can be found under "**My Apps & Credentials**" from the top left menu. Currently the following apps are created:

Site	App ID	App Type	Linked Sandbox Account	Client ID
AU	seafolly_au	REST	proj.seafolly.au@tryzens.com	Acxbr_bAdOHXKeqZ6M6sPk9koNFYdkTG_CrcxSy6nR0FBiczVdi7nK4XA7VQq887xoaz0_LSb
SG	seafolly_sg	REST	proj.seafolly.sg@tryzens.com	AXB_6IUmMKxkxE-JUsw7VOc2ukPeN3_0V8gxViiqlY1VEDfRSxvSmjZ26SbUPatJKD-tA9pEd3i
US	seafolly_us	REST	proj.seafolly.us@tryzens.com	AS4uB20PFRicqH3ju0HrOCDxgyz8_2US02O9zfZIJnUIRY6SabVDipxMA1UMjacKz7-skMBngR
Sunburn				

When selecting the app name, a new windows is opened where you can find all the information for that application. In general, when creating the app it is linked to a sandbox account. See [pp-apps-list.png](#) & [pp-apps-view.png](#).

Settings by PayPal

The "**Billing Address Retrieve**" setting must be enabled from PayPal side for the merchant sandbox account, otherwise the "**Merchant PayPal account does not support the Billing Address retrieving. Contact PayPal for details on eligibility and enabling this feature**" error is raised. This happens when the customers use express checkout from the mini bag or basket page, where the billing address is not submitted from our side to PayPal, but is required by the Braintree cartridge.

Braintree sandbox account

To gain access to the Braintree console & sandbox account use the following login data:

Login url	Username	Password
https://sandbox.braintreepayments.com/login	antonia.dimitrova@tryzens.com	UHpgR~JVj47s'Wu

Information about the Braintree Credit Cards integration and other settings can be found here [Braintree Credit Card Integration](#)

PayPal by Braintree setup

As mentioned above, for the PayPal integration through the Braintree SDK to work, a linked PayPal account must be created. Mocked flow is enabled by default for the merchant ([not merchant account ID](#)), though the cartridge integration works with linked account (["A linked PayPal Sandbox account is required to use PayPal Checkout in Sandbox. See <https://developers.braintreepayments.com/guides/paypal/testing-go-live/#linked-paypal-testing> for details on linking your PayPal sandbox with Braintree."](#) error is shown if no linked account is present).

To do so:

- Select **Settings > Processing** from the top navigation
- Enable the **PayPal** in the **Payment Methods** sections in the top of the screen. See attached [pp-bt-linked-account.png](#)
- After the flow is enabled click on "**Options**" to insert the PayPal email & app credentials. See attached [pp-bt-linked-account-edit.png](#)

Currently there is only one linked PayPal application (the Seafolly AU one). Though the cartridge settings suggest that there may be more than

one, but must be enabled and configured by Braintree side.

More information can be found here:

- <https://developers.braintreepayments.com/guides/paypal/testing-go-live/java#linked-paypal-testing>

Business Manager

System preferences & Services, Payment methods &Processors

The PayPal specific meta must be imported into the respective sites.

1. Braintree Services - the services used for the integration are the same ones used for the Braintree Credit Cards integration and don't need to be imported again. If you don't have them, use the following file "[services.xml](#)". Additional info can be found here [Braintree Credit Card Service Settings](#).
2. Braintree Meta - the same goes for the payment meta. If you don't have it already imported then use the following file "[payment-meta.xml](#)"
3. Payment methods & processors - each site must be configured with the respective processors & methods. Currently the configuration for each of the sites is the same, so you can use these files "[payment-processors.xml](#)" & "[payment-methods.xml](#)".
 - a. make sure that the "PayPal" payment method is enabled & configured to use the "BRAINTREE_PAYPAL" processor (Check under **{SiteName}/Merchant Tools/Ordering/Payment Methods & {SiteName}/Merchant Tools/Ordering/Payment Processors**);
 - b. configure the Braintree PayPal custom preferences for each site - refer to the next section.
4. Add the cartridge to each Site path.

Custom Preferences

The integration requires that 2 custom preference groups are edited. The common settings in **BRAINTREE_COMMON** are already setup in the [Braintree Credit Card Custom Preference](#) and described below.

- **Braintree Common Preferences - {SiteName}/Merchant Tools/Custom preferences/BRAINTREE_COMMON**

Preference ID	Preference Name	Preference Values			
		Seafolly US	Seafolly AU	Seafolly SG	Sunburn
BRAINTREE_Merchant_ID	Merchant ID **	22cpyhtv6nsptgp	22cpyhtv6nsptgp	22cpyhtv6nsptgp	22cpyhtv6nspt
BRAINTREE_ECVZ_Access_Token	ECVZ Access Token				
BRAINTREE_ECVZ_Api_Url	ECVZ API URL				
BRAINTREE_Merchant_Account_ID	Merchant Account ID * *	seafolly_us	seafolly_au	seafolly_sg	sunburn
BRAINTREE_Tokenization_Key	Tokenization Key				
BRAINTREELogging_Mode	Logging *	all	all	all	all
BRAINTREE_SG_Controllers_Cartridge_Name	SiteGenesis Controllers Cartridge Name	app_seafolly_controllers	app_seafolly_controllers	app_seafolly_controllers	app_seafolly_cc
BRAINTREE_Log_Name_Prefix	Log Name Prefix ***	Braintree_service_US	Braintree_service_AU	Braintree_service_SG	Braintree_servic

* Logging - In order for communication logging to work, log level must be set to DEBUG for "root" category and INFO/DEBUG levels must be enabled for "Log Files" under [Administration/Operations/Custom Log Settings](#).

** Merchant ID and Merchant Account ID are not to be confused. **Merchant ID** is uniquely generated by Braintree and can not be changed. **Merchant Account ID** can be created in the Braintree console to accept different payments and currencies.

The PayPal specific settings are described below:

- **Braintree PayPal Preferences - {SiteName}/Merchant Tools/Custom preferences/BRAINTREE_PAYPAL**

Preference ID	Preference Name	Preference Values		
		Seafolly AU	Seafolly SG	Seafolly US
BRAINTREE_PAYPAL_Visibility_Express_Button_On_Cart	Visibility of PayPal Express Checkout Button on the Cart page	Yes	Yes	Yes
BRAINTREE_PAYPAL_Visibility_Credit_Button_On_Cart	Visibility of PayPal Credit Button on the Cart page	No	No	No
BRAINTREE_PAYPAL_Payee_Email ***	Payee Email	proj.seafolly.au@tryzens.com	proj.seafolly.sg@tryzens.com	proj.seafolly.us@try:
BRAINTREE_PAYPAL_Display_Name	Merchant Name	Seafolly AU	Seafolly SG	Seafolly US
BRAINTREE_PAYPAL_PayNow_Button_Enabled	PayNow Button	Disabled	Disabled	Disabled
BRAINTREE_PAYPAL_Payment_Model	Payment Model (Intent)	Auth and Immediate Settlement (sale)	Auth and Immediate Settlement (sale)	Auth and Immediate Settlement (sale)
BRAINTREE_PAYPAL_Shipping_Address_Override *	Shipping Address Override	Yes	Yes	Yes
BRAINTREE_PAYPAL_Vault_Mode	Store Payment Method in Vault	Do not vault (not)	Do not vault (not)	Do not vault (not)
BRAINTREE_PAYPAL_Fraud_Tools_Enabled	Advanced Fraud Tools			
BRAINTREE_PAYPAL_Billing_Address_Override **	Billing Address Override	Yes	Yes	Yes
BRAINTREE_PAYPAL_Custom_Fields	Default Custom Fields			
BRAINTREE_PAYPAL_Billing_Agreement_Description	Billing Agreement Description			
BRAINTREE_PAYPAL_Descriptor_Name	Descriptor Name			
BRAINTREE_PAYPAL_Locale	Locale			

* ** Billing/Shipping Address Override - contrary to the fields description, if "Yes" is selected the address we pass to PayPal will be used for the checkout and will override that selected from the buyer account in PayPal. If "No" is selected or no address is passed to PayPal (i.e. during express checkout from mini bag & basket page), then the address will be returned from PayPal buyer's account.

* Billing Address Override - If this option is set to "Yes" and the "Billing Address Retrieve" setting is not enabled for the sandbox PayPal merchant we will receive the "**Merchant PayPal account does not support the Billing Address retrieving. Contact PayPal for details on eligibility and enabling this feature**" error during the express checkout from mini bag & basket page.

*** Payee Email:

- If no payee email is filled the transactions will be settled into the linked PayPal account in the Braintree console. In case that multiple currencies are settled into the account, the accepted PayPal merchant currencies must be enabled. For reference on enabling them see PayPal Multiple Currencies
- **If email if entered, this must be approved and setup by Braintree side on PRODUCTION!**

Testing

Currently all PayPal transactions are always verified and processed.

PayPal suggests that "**Negative Testing**" is to be used to make test. That can be achieved as you enable the setting in the specific sandbox merchant account:

- login into the Developer console - <https://developer.paypal.com>. See credentials above.
- Select "**Accounts**" from the left menu & find the needed sandbox account. Click n the arrow below it and select "**Profile**" link . See [pp-account-settings.png](#)
- In the popup select the "**Settings**" tab. Enable/Disable the negative testing. See [pp-account-setting-enable.png](#)

Negative testing works with specific conditions that need to met for it to work - the total amount or the amount must be specific values. Info can be found here: <https://developer.paypal.com/docs/classic/lifecycle/nt-classic/>

As the negative testing is harder to be used, there is another way to make the payments fail - it is to enable the "**Payment Review**" setting. When it is enabled, **all payments fail** with the following message "**PayPal pending payments are not supported**" (Processor code 2081).

For more info see bellow:

- Testing Guide - https://developer.paypal.com/docs/classic/lifecycle/ug_sandbox/
- Negative Testing - <https://developer.paypal.com/docs/classic/lifecycle/nt-classic/>, https://developer.paypal.com/docs/classic/lifecycle/sb_error-conditions/

Attachments

Images	Meta	Documentation	FSD	Designs
pp-bt-linked-account.png	services.xml	Braintree Integration Guide		
pp-bt-linked-account-edit.png	payment-meta.xml			
pp-apps-view.png	payment-processors.xml			
pp-apps-list.png	payment-methods.xml			
pp-business-profile.png				
pp-business-merchant-id.png				
pp-accounts-list.png				
pp-account-credentials.png				
pp-account-settings.png				
pp-account-setting-enable.png				

Useful Links

Description	Link
Braintree PayPal Integration	https://developers.braintreepayments.com/guides/paypal/overview/javascript/v3
Client Side Implementation	https://developers.braintreepayments.com/guides/paypal/client-side/javascript/v3
Linking PayPal Account	https://developers.braintreepayments.com/guides/paypal/testing-go-live/java#linked-paypal-testing
PayPal Interactive Demo	https://developer.paypal.com/demo/checkout/#/pattern/styles
Braintree PayPal Button Overview	https://braintree.github.io/braintree-web/current/module-braintree-web_paypal.html https://braintree.github.io/braintree-web/current/PayPal.html
Braintree PayPal Checkout Overview	https://braintree.github.io/braintree-web/current/module-braintree-web_checkout.html https://braintree.github.io/braintree-web/current/PayPalCheckout.html
PayPal Button Customization	https://developer.paypal.com/docs/integration/direct/express-checkout/integration-javascript/customize-button/

PayPal Checkout Customization	https://developer.paypal.com/docs/integration/direct/express-checkout/integration-javascript/customize-the-checkout-flow/
Sandbox Overview	https://developer.paypal.com/docs/classic/lifecycle/sb_overview/
Testing Guide	https://developer.paypal.com/docs/classic/lifecycle/ug_sandbox/
Negative Testing	https://developer.paypal.com/docs/classic/lifecycle/nt-classic/
API Error Codes	https://developer.paypal.com/docs/classic/api/errors/
API Error messages	https://developer.paypal.com/docs/api/payments/#errors
Credit Cards Generator	http://credit-card-generator.2-ee.com/
SNN generator & validator	http://www.theonegenerator.com/ssngenerator http://www.ssnvalidator.com/index.aspx

Linked Integrations

Description	Link
Credit Cards by Braintree	Braintree Credit Card Integration

PCA Predict Integration

PCA Predict is a service for address validation and verification.

This case explains the custom JS integration (also known as "Traditional" in PCA terms) of PCA Predict Capture+ into the Seafolly/Sunburn sites. Traditional installation is the one that uses all-purpose API service keys and does not include the form mapping option by PCA Predict (using PCA tagging).

The used PCA predict library is Capture+ v2.30

Sandbox account

The following can be used to gain access to the PCA Predict environment. Sandbox accounts expire in 2-weeks time and have 50 API lookups/day.

Environment	Login Url	Email	User/Account code	Password	Comments
sandbox	https://account.pcapredict.com/Security/Login/	antonia.dimitrova@tryzens.com	TRYZE11122	zN85FxAlG!@oa2z	expired on 18.08.2017
		adimitrova.tryzens1@gmail.com	TRYZE11131	zN85FxAlG!@oa2z	expires 20.02.2018

User/Account code & password are used as login credentials.

API service keys

You can choose 2 ways in order to create a new API key:

- "All services" from the left menu > choose "New Service" button - which navigates to the option bellow (refer to attached image "service-list.png")
- "New Service" from the left menu > choose API Key "Set up" button in the bottom of the page (refer to attached image "service-new.png")

New API key is automatically generated for you and you are redirected to the edit page. Here you can make changes regarding the daily limit amount, service name and IP/Url restrictions. See [service-edit.png](#) for more details.

The account mode and the remaining daily lookups can be seen in the bottom on the left menu (see "welcome.png").

The following API keys are currently present.

Account Code	API Key	Account Expire Date	Status
TRYZE11122	CP85-NG67-WY45-XK86	18.08.2017	Expired
TRYZE11128	UY39-RA14-EH96-UK97	13.11.2017	Expired
	TX98-WC43-EU89-CA92	20.12.2017	Expired
	WD54-YC99-CN58-RE63	24.01.2018	Expired
TRYZE11131	PY21-CW75-FM76-TH99	20.02.2018	Expired
	WD22-NA98-BE63-PB96	10/4/2018	Active
	NK12-XK57-MJ59-DB96	 11 May 2018	Active

Business Manager

System Preferences

The PCA Predict meta must be included in the Site Preferences for the sites. File `pca-predict-meta.xml` contains the required preferences definitions.

To do so go to **Administration / Site Development / Import & Export** and import the xml file.

The file contains declaration of 5 attributes and 1 attribute group. **As the file contains attribute group, only one import is sufficient.**

Custom Preferences

The site-specific PCA Predict settings can be found under `{SiteName}/Merchant Tools/Custom preferences/`, look for the "**PCAPREDICT SETTINGS**" settings group.

Currently, the following configuration is used:

Preference ID	Preference Name	Preference Values			
		Seafolly US	Seafolly AU	Seafolly SG	Sunburn
PCAPREDICT_Api_Key	API Key	PY21-CW75-FM76-TH99	PY21-CW75-FM76-TH99	PY21-CW75-FM76-TH99	PY21-CW75-FM76-TH99
PCAPREDICT_Enabled	Enabled	true	true	true	true
PCAPREDICT_Css_Url	URL to PCA Predict Css File	https://services.postcodeanywhere.co.uk/css/captureplus-2.30.min.css			
PCAPREDICT_Js_Url	URL to PCA Predict Js File	https://services.postcodeanywhere.co.uk/js/captureplus-2.30.min.js			

PCAPREDICT_Countries preference is removed and no longer needed and supported !!!

JS configuration

The PCA Integration uses the `pcaUtils.js` located under `app_seafolly_core/cartridge/js`.

The default setting can be found below:

```

params: {
  enabled: false,
  options: {
    key: ''
  },
  instanceContainerID: '',
  errorContainerID: '',
  showFullFormID: '',
  fullFormCssClass: 'show-full-form',
    fieldValueCssClass: 'not-empty',
    fieldValidCssClass: 'valid',
    fieldErrorCssClass: 'error',
  addressCheckID: '',
  fields:{
    searchID: {selector: '', field: 'Line1', mode: 'search'},
    countryID: {selector: '', field: 'CountryIso2', mode: 'populate'},
    stateID: {selector: '', field: 'Province', mode: 'populate'},
    cityID: {selector: '', field: 'City', mode: 'populate'},
    zipID: {selector: '', field: 'PostalCode', mode: 'populate'},
    address1ID: {selector: '', field: 'Line1', mode: 'populate'},
    address2ID: {selector: '', field: 'Line2', mode: 'populate'}
  }
}

```

Option	Description
enabled	enable/disable the PCA Predict for this implementation; if disabled no search field is shown and full form is loaded
options	Object containing the "options" param of Capture+ Address <ul style="list-style-type: none"> ■ key - the API key to be used
instanceContainerID	the DOM ID of the form/container element
errorContainerID	the DOM ID of the error container element
showFullFormID	the DOM ID of the 'show full form' parent container element; it is not the 'show full form' link
fullFormCssClass	the CSS class to be applied to the form when opened
fieldValueCssClass	the CSS class to be applied to each of the fields if they have value
fieldValidCssClass	the CSS class to be applied to each of the fields if they are valid
fieldErrorCssClass	the CSS class to be applied to each of the fields if they are not valid or not returned from Capture+ Address search
addressCheckID	the DOM ID of the hidden input containing Base64 encoded API Key

fields	<p>Object containing the mapped fields data:</p> <ul style="list-style-type: none"> ■ searchID - search input ■ countryID - country code input ■ stateID - state code input ■ cityID - city name input ■ zipID - zip input ■ address1ID - line1 input ■ address2ID - line2 input <p>Each one of them must have the following properties:</p> <ul style="list-style-type: none"> ■ selector - the DOM ID of the input field ■ field - the PCA Predict ID of the field ■ mode - one or more of the pcaUtil modes (described below)
--------	--

Specific implementation can have the following JS initialization code. The initialization params are merged with the default ones using deep extend (\$.extend(true, target, objN) - recursive merge of params) - if some of the options are omitted the default ones are to be used.

```

document.addEventListener("DOMContentLoaded", function(event) {
  pcaUtils.pcaForm.init({
    enabled: "true",
    instanceContainerID: "dwfrm_billing",
    errorContainerID: "pca-billing-error",
    showFullFormID: "pca-show-billing-form",
    fullFormCssClass: "show-full-form",
    addressCheckID: "address-check-id",
    fields: {
      searchID: {selector:
        "#dwfrm_billing_billingAddress_addressFields_address" },
      countryID: {selector:
        "#dwfrm_billing_billingAddress_addressFields_country" },
      stateID: {selector:
        "#dwfrm_billing_billingAddress_addressFields_states_state" },
      cityID: {selector: "#dwfrm_billing_billingAddress_addressFields_city" },
      zipID: {selector: "#dwfrm_billing_billingAddress_addressFields_postal" },
      address1ID: {selector:
        "#dwfrm_billing_billingAddress_addressFields_address1", mode:
        "search,populate" },
      address2ID: {selector:
        "#dwfrm_billing_billingAddress_addressFields_address2" }
    }
  });
}
);

```

File "**pcapredict/predict_hidden**" must be included in the isml template where you want to use the PCA Predict integration. This file contains the Base64 encoded API key. Currently it is included in the "**header.isml**" template and can be used on every page.

Field modes

The following field modes are available for use - pcaUtils modes are mapped to the respective PCA Predict mode when generating the initialization for the Capture+ Address

pcaUtils Field Mode	PCA Predict Mode (mapped)	Description
default	pca.fieldMode.DEFAULT	search populate
search	pca.fieldMode.SEARCH	enables autocomplete for field

populate	pca.fieldMode.POPULATE	populates the field with the respective returned value
preserve	pca.fieldMode.PRESERVE	prevents overwriting of already manually completed fields; if initially blank will be populated later
country	pca.fieldMode.COUNTRY	load countries list when on focus & sets the search by this country

Use comma delimited list for more than one pcaUtils mode per field.

Attachments

Images	Meta Data	Documentation
welcome.png		
service-list.png		
service-list-tags.png		
service-new.png		
service-edit.png		
service-delete.png		

Useful Links

Description	Link
PCA Predict Overview	https://www.pcapredict.com/address-capture-software/
Capture+ Find API	https://www.pcapredict.com/support/webservice/capture/interactive/find/1/
Capture+ Retrieve API	https://www.pcapredict.com/support/webservice/capture/interactive/retrieve/1/
Example - fields modes	https://www.pcapredict.com/capture-plus/examples/hello.html
Example - jQuery UI	https://www.pcapredict.com/capture-plus/examples/jquery.html
Demos	https://www.pcapredict.com/address-validation/try-it-now/
Capture+ API Address Class	https://www.pcapredict.com/capture-plus/reference/2.30/pca.address.html
Capture+ JS Guide	https://www.pcapredict.com/support/articles/article/captureplus-javascript-guide/

UY39-RA14-EH96-UK97

PDP - product details page and Product Set page

Accordions

To display according to selected product you should enter information in

Size And Fit - for size and fit content

Fabric Care - information about fabric

Description - information about product

Delivery And Returns - information about delivery and returns (is there is no content it will get information from content slot delivery and returns)

If there are no content in Size And Fit, Fabric Care, Description it won't create accordion.

Sunburn Brands

To display brand image to PDP you should do next steps.

1. Go and find product that you want to add brand image.
2. Click and go to Categories tab to assign brand category.
3. Then go back in General tab and find field called Brand.
4. In this field enter the same name like category brand that you assign previously.

You have two cases if there is image in brand category and if the is not.

In first case it will show images selected in category. In second case it will show the name of this brand without image.

To set image for brand logo in category go in selected category and find 'Images for brand logo' field. It there you can set image.

Size Guide

You must set content asset before you set link for size guide. Content asset ID will be the used for Size guide link.

After setting content asset, go in the selected main or children category and find 'Size Chart' field and there set content asset ID.

PLP - Product Listing Page

Color Refinements Configuration

All possible colors of the products should be configured as "MasterColour" custom objects in the BM.

Go to **BM > Merchant Tools > (select Site) > Custom Objects > Custom Object Editor**. Select **Master Colour** for **Object Type**. You can create new color variable or to edit existing one.

The **colourID** field is required and that is the name of the color variable and the color refinement. The values are case sensitive, so upper- and lowercase of the color names are important.

Example: if the product has variation BLACK, the ID of the custom object should be BLACK (Black wouldn't work). Same is for the intervals in the words.

You could enter a **HEX** value for color in the field "Colour" or **Select** an image in the "Image" field if the wanted CSS Color HEX Value does not exist.

You could use http://www.w3schools.com/cssref/css_colors.asp for example HEX values.

Example for the Black refinement:

`master_color_custom_object.png`

Images swap on hover

When customer browsing the desktop site, and hover on product tile the image of this tile will swap with second image on this product.

In order to use this function you must have second image for your products and products set. If you don't choose second image this function it will not work. Because it's necessary to have this second image to make this image swap.

To activate this function you must follow this document.

Configuration

1. Open BM
2. Select Merchant Tools
3. Select Site Preferences

4. Select Custom Preferences
5. Then select Seafolly Configs, this will open all preferences for Seafolly project
6. Find Image Swap Enable
7. To enable this function across all site chose from dropdown "Yes" value.
8. If you want this function to be active for one site chose "Edit Across Sites" link. This will open a new window and from it you can choose different preferences for different sites.

Preference Id

The id of this preference is **imageSwapEnable**. You can find it on the follow path Administration > Site Development > System Object Types > SitePreferences

Product carousels

This page will describe how product carousels work on PDP page.

Configuration

Enable recently viewed

1. Open BM
2. Select Merchant Tools
3. Select Site Preferences
4. Select Custom Preferences
5. Then select Seafolly Configs, this will open all preferences for Seafolly project
6. Find 'Enable recently viewed'
7. To enable this function across all site chose from dropdown "Yes" value.
8. If you want this function to be active for one site chose "Edit Across Sites" link. This will open a new window and from it you can choose different preferences for different sites.

Enable recommendations

1. Open BM
2. Select Merchant Tools
3. Select Site Preferences
4. Select Custom Preferences
5. Then select Seafolly Configs, this will open all preferences for Seafolly project
6. Find 'Enable recommendations'
7. To enable this function across all site chose from dropdown "Yes" value.
8. If you want this function to be active for one site chose "Edit Across Sites" link. This will open a new window and from it you can choose different preferences for different sites.

Site preferences

ID	Name	Value
recentlyViewedItems	Recently viewed items	number
enableRecommendations	Enable recommendations	Yes No
enableRecentlyViewed	Enable recently viewed	Yes No

Set recently viewed items

Enter a number to choose how much recently viewed element to show in storefront by using 'Recently viewed items' preferences

Screens and devices

Mobile - 320px

One product for 'You may Also like' and 'Recently Viewed' carousels - no carousel, center product and without arrows.

Two and more products for 'You may Also like' and 'Recently Viewed' carousels - carousel, center product and arrows.

Tablet - 768px, Laptop - 1024px, Desktop - 1440px

'You may Also like' is active

'You may Also like' is full width style.

Less or equal to 4 for 'You may Also like' without 'Recently Viewed' - no carousel, center product and without arrows.

More than 4 products for 'You may Also like' without 'Recently Viewed' - carousel, center product and arrows.

'Recently Viewed' is active

'Recently Viewed' is full width style.

Less or equal to 4 for 'Recently Viewed' without 'You may Also like' - no carousel, center products and without arrows.

More than 4 products for 'Recently Viewed' without 'You may Also like' - carousel, center products and arrows.

Both

'You may Also like' and 'Recently Viewed' are half width.

One product for 'You may Also like' and just two products for 'Recently Viewed' carousels.

'You may Also like' - no carousel, center product and without arrows.

'Recently Viewed' - no carousel, center product and without arrows.

One product for 'You may Also like' and more than two products for 'Recently Viewed' carousels.

'You may Also like' - no carousel, center product and without arrows.

'Recently Viewed' - carousel, center product and arrows.

Just two products for 'You may Also like' and 'Recently Viewed' carousels

'You may Also like' and 'Recently Viewed' carousels - no carousel, center product and without arrows.

More than two products for 'You may Also like' and just two products for 'Recently Viewed' carousels

'You may Also like' - carousel, center product and arrows.

'Recently Viewed' - no carousel, center products and without arrows.

More than two products for 'You may Also like' and 'Recently Viewed' carousels

'You may Also like' - carousel, center product and arrows.

'Recently Viewed' - carousel, center product and arrows.

JS files

We include Owl Carousel v2.2.1 to this project.

Project general info

SVN - <https://ecommerce-svn.tryzens.com/svn/seafolly/>

FSDs

Approved BRD

Designs in inVision

Designs in Zeplin

Proposed architecture:

[Seafolly TO BE Proposed Architecture - v0.5.pdf](#)

Sandbox accounts for 3-rd parties

Supported browsers and devices:

Latest version of FF and Chrome, IE11 and IE Edge, Safari on Mac, iPad Air, iPhone 6S and 7 Plus (latest iOS version), Samsung Galaxy S5.

Build Suite user guide:

[BuildSuite_Seafolly.docx](#)

Development process user guide:

[Development Process.docx](#)

2-factor commands

[2-factor-commands.txt](#)

Storefront Login for all sites and instances : **storefront / london2020**

WebDav acces:



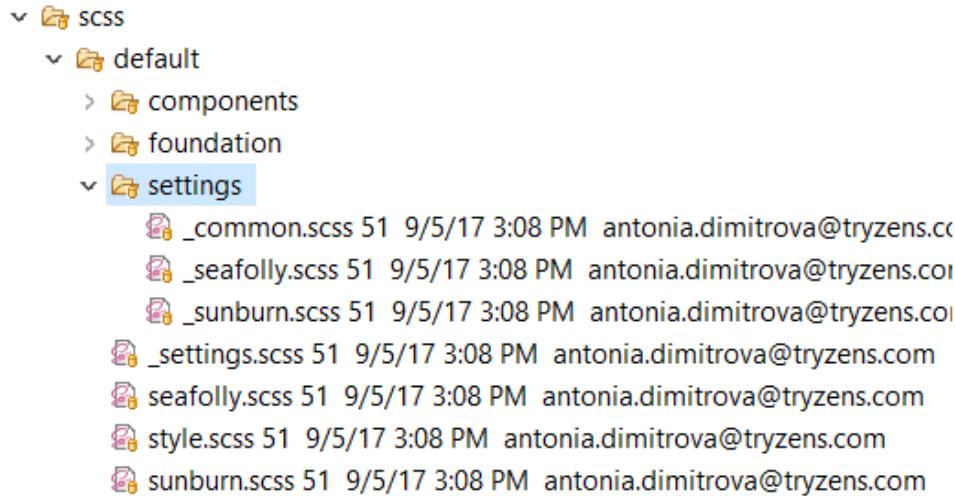
Release 1

Sass, CSS, JS, Sprites, Grunt

The css settings are divided into common & site specific ones:

- **partial css files** are located in the **settings** folder
- **main css files** are located directly into the **scss/default** folder

and form the following structure:



SVG files

How to make SVG sprites with grunt

grunt svg_sprite:dev

Using common variables, styles, and mixins for Seafolly / Sunburn

Note 1: The variables which are defined in Seafolly /_seafolly_variables.scss/ have to be repeated in Sunburn /_sunburn_variables.scss/ and the oposite.

The main colors for each site (\$sf- & \$sb-*) can be used only in the respective stylesheet i.e. :*

- **\$sf-*** variables can be used only within the **_seafolly_variables.scss** file
- **\$sb-*** variables can be used only within the **_sunburn_variables.scss** file.

They must be assigned to variables that are the ones to be used within all other files. For example, if you want to create variable for the mini bag background color **\$mini-bag-background** it must be defined as follows, and **\$mini-bag-background** can be used in all other files:

- **\$mini-bag-background: \$sf-grey;** (in **_seafolly_variables.scss**)
- **\$mini-bag-background: \$sb-green;** (in **_sunburn_variables.scss**)

Note 2: Button style includes hover, focus, and disabled status.

Note 3: SG styles redefine button styles in some cases. They need to be checked.

Note 4: Global style means that the style is already included. **Override style** means that the style can override the global style class.

Primary Button - global style

"button-primary" mixin is included:

button-primary

```
button {  
  @include .button-primary;  
}
```

Example:

```
vertical-align: top;  
}  
  
button,  
.button,  
.buttonstyle,  
input[type="button"] {  
  @include button-primary;  
  &.large {  
    font-size: 1.35em;  
    text-transform: uppercase;  
  }  
  &.raised {  
    border-width: 0;  
    box-shadow: 0 2px 5px 0 rgba(0, 0, 0, 0.16), 0 2px 10px 0 rgba(0, 0, 0, 0.12);  
    &:hover {  
      background-color: darken($citrus, 10%);  
      color: $white;  
    }  
  }  
  &.simple {  
    background-color: $white;  
  }  
}
```

Primary Button (Right)

"button-primary-right" mixin has to be included:

button-primary-right

```
button {  
  @include button-primary-right;  
}
```

Note: The style is only for **Seafolly** sites according the style guides.

Secondary Button

"button-secondary" mixin has to be included:

button-secondary

```
button {  
  @include button-secondary;  
}
```

Large Button

"button-large" mixin has to be included:

button-large

```
button {  
  @include button-large;  
}
```

Edit Button

"button-reset" and "button-edit" mixins have to be included:

button-edit

```
button {  
  @include button-reset;  
  @include button-edit;  
}
```

Remove Button

"button-reset" and "button-remove" mixins have to be included:

button-remove

```
button {  
  @include button-reset;  
  @include button-remove;  
}
```

Small Remove Button

"button-reset" and "button-remove-small" mixins have to be included:

button-remove-small

```
button {  
  @include button-reset;  
  @include button-remove-small;  
}
```

Input & Textarea With Label - global style

"input-with-label" mixin is included in ".field-wrapper" class:

input-with-label

```
.field-wrapper {  
  @include input-with-label;  
}
```

Input & Textarea Without Label

"input-with-label" mixin has to be included in ".field-wrapper" class:

input-without-label

```
.field-wrapper {  
  @include input-without-label;  
}
```

Mobile Input & Textarea With Label

"input-with-label-mobile" mixin has to be included in ".field-wrapper" class:

input-with-label-mobile

```
.field-wrapper {  
  @include input-with-label-mobile;  
}
```

Mobile Input & Textarea Without Label

"input-without-label-mobile" mixin has to be included in ".field-wrapper" class:

input-without-label-mobile

```
.field-wrapper {  
  @include input-without-label-mobile;  
}
```

Select - global style

"div-select" mixin is included in ".div-select" class and "select" mixin is included in select tag style:

select

```
.div-select {  
  @include div-select;  
}  
select {  
  @include select;  
}
```

Mobile Select

"select-mobile" mixin has to be included in select tag style:

select-mobile

```
select {  
  @include select-mobile;  
}
```

Note: The style is only for **Seafolly** sites according the style guides.

Radio - global style

In order for the radio button to accept the style correctly, it must be in the following HTML format and sequence:

Radio Button HTML

```
<input id="_some_id" type="radio" ... ><label for="_some_id">
```

The radio button must be followed by a label attached to it with an ID.

"button-radio" mixin is included:

button-radio

```
[type="radio"] {  
  @include button-radio;  
}
```

Note: In case we have one label tag attached to two radio buttons, they won't work correctly.

Checkbox - global style

In the same way like the radio button, in order for the checkbox button to accept the style correctly, it must be in the following HTML format and sequence:

Checkbox Button HTML

```
<input id="_some_id" type="checkbox" ... ><label for="_some_id">
```

The Checkbox must be followed by a label attached to it with an ID.

"button-checkbox" mixin is included:

button-checkbox

```
[type="checkbox"] {  
  @include button-checkbox;  
}
```

Note: One checkbox may have two label tags attached and it still going to work correctly.

Labels - global style

"labels" mixin is included:

labels

```
label {  
  @include labels;  
}
```

Error - global style

"errors" mixin is included:

errors

```
span.error,  
div.error {  
  @include errors;  
}
```

Normal Link

"link-normal" mixin has to be included:

link-normal

```
a {  
  @include link-normal;  
}
```

Reversed Link

"link-reversed" mixin has to be included:

link-reversed

```
a {  
  @include link-reversed;  
}
```

Special Link

"link-special" mixin has to be included:

link-special

```
a {  
  @include link-special;  
}
```

Paragraph - global style

"paragraph" mixin is included:

paragraph

```
p {  
  @include paragraph;  
}
```

Small Paragraph

"paragraph-small" mixin has to be included:

paragraph-small

```
p {  
  @include paragraph-small;  
}
```

H1 Heading - global style

"heading-h1" mixin is included:

heading-h1

```
h1 {  
  @include heading-h1;  
}
```

Special Heading

"heading-special" mixin has to be included:

heading-special

```
h1 {  
  @include heading-special;  
}
```

Note: The style is only for **Seafolly** sites according the style guides.

Mobile H1 Heading

"heading-h1-mobile" mixin has to be included:

heading-h1-mobile

```
h1 {  
  @include heading-h1-mobile;  
}
```

Mobile Special Heading

"heading-special-mobile" mixin has to be included:

heading-special-mobile

```
h1 {  
  @include heading-special-mobile;  
}
```

Note: The style is only for Seafolly sites according the style guides.

H2 Heading - global style

"heading-h2" mixin is included:

heading-h2

```
h2 {  
  @include heading-h2;  
}
```

Mobile H2 Heading

"heading-h2-mobile" mixin has to be included:

heading-h2-mobile

```
h2 {  
  @include heading-h2-mobile;  
}
```

H3 Heading - global style

"heading-h3" mixin is included:

heading-h3

```
h3 {  
  @include heading-h3;  
}
```

Mobile H3 Heading

"heading-h3-mobile" mixin has to be included:

heading-h3-mobile

```
h3 {  
  @include heading-h3-mobile;  
}
```

H4 Heading - global style

"heading-h4" mixin is included:

heading-h4

```
h4 {  
  @include heading-h4;  
}
```

Mobile H4 Heading

"heading-h4-mobile" mixin has to be included:

heading-h4-mobile

```
h4 {  
  @include heading-h4-mobile;  
}
```

Unordered list

"list-unordered" mixin has to be included:

list-unordered

```
ul {  
  @include list-unordered;  
}
```

Ordered list

"list-ordered" mixin has to be included:

list-ordered

```
ol {  
  @include list-ordered;  
}
```

&

Seafolly Integrations

- Overview
- Installation
- Modules List

Overview

This cartridge provides integration with Fusion Factory middleware.

Installation

Please add **int_seafolly** to cartridge path for all sites including Business Manager.

Requirements

Integration Framework Cartridge

This cartridge requires **bc_integrationframework** cartridge to be present in cartridge path for all sites including Business Manager.

Modules List

- Customers Export
- Customers Import
- Gift Vouchers Import
- Newsletter Export
- Stores Import
- Transfer To FTP

Customers Export

- Overview
- Requirements
- Scenario
- Job Step Parameters
- Meta Data
 - System Objects
 - Site Preferences
 - Jobs

Overview

This module handles export of customer objects to file.

Functional specifications agreed with Fusion Factory:

[Mailing list subscription and Customer data flows for Seafolly.docx](#)

Requirements

No requirements.

Scenario

In order to determine when user has changed certain parts of his profile there's time-stamped field **lastModifiedByCustomer** which is updated when the following actions occur:

1. New profile creation
2. Edit Account but not for Change Password action
3. Add/Change/Remove of any Address

When scheduled Job for export is executed it gets all customers which has **lastModifiedByCustomer > lastCustomerExport** site's preference. Then the job proceeds with exporting these objects into file. If any error occurs, the job stops otherwise it should log how many objects are found

and exported. If the job succeeds, **lastCustomerExport** site's preference is updated.

Job Step Parameters

1. **SiteID**: SiteID e.g. SeafollyEU
2. **ExportPath**: containing export path in format /FF/ + SiteID + /Customers e.g. /FF/SeafollyAU/Customers

Meta Data

System Objects

This field is required because when you import customers they receive lastModified to be current import time.

System Object Meta

```
<?xml version="1.0" encoding="UTF-8"?>
<metadata xmlns="http://www.demandware.com/xml/impe.../2006-10-31">
    ...
    <type-extension type-id="Profile">
        ...
        <custom-attribute-definitions>
            ...
            <attribute-definition attribute-id="lastModifiedByCustomer">
                <display-name xml:lang="x-default">Last modified by
Customer</display-name>
                <type>datetime</type>
                <mandatory-flag>false</mandatory-flag>
                <externally-managed-flag>false</externally-managed-flag>
            </attribute-definition>
        </custom-attribute-definitions>
        <group-definitions>
            <attribute-group group-id="Standard">
                ...
                <attribute attribute-id="lastModifiedByCustomer"/>
            </attribute-group>
        </group-definitions>
    </type-extension>
    ...
    <type-extension type-id="SitePreferences">
        <custom-attribute-definitions>
            ...
            <attribute-definition attribute-id="lastCustomerExport">
                <display-name xml:lang="x-default">Last Customer
Export</display-name>
                <type>datetime</type>
                <mandatory-flag>false</mandatory-flag>
                <externally-managed-flag>false</externally-managed-flag>
            </attribute-definition>
            ...
        </custom-attribute-definitions>
        <group-definitions>
            ...
            <attribute-group group-id="Integration_Seafolly">
                <display-name xml:lang="x-default">Integration
Seafolly</display-name>
                <attribute attribute-id="lastCustomerExport"/>
            </attribute-group>
            ...
        </group-definitions>
    </type-extension>
    ...
</metadata>
```

Site Preferences

Apply this for SeafollyAU, SeafollySG, SeafollyUS and Sunburn sites!

Site Preferences FOR EACH SITE

```
<?xml version="1.0" encoding="UTF-8"?>
<preferences
xmlns="http://www.demandware.com/xml/impex/preferences/2007-03-31">
...
<custom-preferences>
...
<development>
...
<preference
preference-id="lastCustomerExport">2017-01-01T00:00:00.000+0000</preference>
...
</development>
...
</custom-preferences>
</preferences>
```

Jobs

Jobs Meta Data

```
<?xml version="1.0" encoding="UTF-8"?>
<jobs xmlns="http://www.demandware.com/xml/impex/jobs/2015-07-01">
...
<job job-id="CustomersSeafollyAU" priority="0">
<description/>
<parameters/>
<flow>
    <context site-id="SeafollyAU"/>
    <step step-id="CustomersSeafollyAU" type="ExecutePipeline"
enforce-restart="false">
        <description/>
        <parameters>
            <parameter
name="ExecutePipeline.Pipeline">ExportCustomers-Start</parameter>
            <parameter name="SiteID">SeafollyAU</parameter>
            <parameter
name="ExportPath">/FF/SeafollyAU/Customers</parameter>
        </parameters>
        <rules>
            <on-exit status="ERROR">
                <stop-job/>
            </on-exit>
        </rules>
```

```

        </step>
        <step step-id="TransferSeafollyAU" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">TransferToFtp-Start</parameter>
                    <parameter name="HostURL">52.62.59.176</parameter>
                    <parameter name="UserID">seafolly</parameter>
                    <parameter name="Password">W2dReth$</parameter>
                    <parameter name="FilePattern">^.*\.\xml$</parameter>
                    <parameter
name="SourceFolder">/src/FF/SeafollyAU/Customers</parameter>
                    <parameter
name="TargetFolder">/root/dw_staging/upload/customer</parameter>
                    <parameter name="SecureFtp">SFTP</parameter>
                </parameters>
                <rules>
                    <on-exit status="ERROR">
                        <stop-job/>
                    </on-exit>
                </rules>
            </step>
        </flow>
    <rules/>
    <triggers>
        <run-recurring enabled="true">
            <recurrence>
                <date-from>2017-10-25Z</date-from>
                <start-time>16:15:00.000Z</start-time>
                <interval>1d</interval>
                <day-of-week/>
            </recurrence>
        </run-recurring>
    </triggers>
</job>

<job job-id="CustomersSeafollySG" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="SeafollySG"/>
        <step step-id="CustomersSeafollySG" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">ExportCustomers-Start</parameter>
                    <parameter name="SiteID">SeafollySG</parameter>
                    <parameter
name="ExportPath">/FF/SeafollySG/Customers</parameter>
                </parameters>
            <rules>

```

```

                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
        <step step-id="TransferSeafollySG" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">TransferToFtp-Start</parameter>
                <parameter name="HostURL">52.62.59.176</parameter>
                <parameter name="UserID">seafolly</parameter>
                <parameter name="Password">W2dReth$</parameter>
                <parameter name="FilePattern">^.*\.xml$</parameter>
                <parameter
name="SourceFolder">/src/FF/SeafollySG/Customers</parameter>
                <parameter
name="TargetFolder">/root/dw_staging/upload/customer</parameter>
                <parameter name="SecureFtp">SFTP</parameter>
            </parameters>
            <rules>
                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
    </flow>
    <rules/>
    <triggers>
        <run-recurring enabled="true">
            <recurrence>
                <date-from>2017-10-25Z</date-from>
                <start-time>16:15:00.000Z</start-time>
                <interval>1d</interval>
                <day-of-week/>
            </recurrence>
        </run-recurring>
    </triggers>
</job>

<job job-id="CustomersSeafollyUS" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="SeafollyUS"/>
        <step step-id="CustomersSeafollyUS" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">ExportCustomers-Start</parameter>
                <parameter name="SiteID">SeafollyUS</parameter>

```

```

        <parameter
name="ExportPath">/FF/SeafollyUS/Customers</parameter>
    </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
<step step-id="TransferSeafollyUS" type="ExecutePipeline"
enforce-restart="false">
    <description/>
    <parameters>
        <parameter
name="ExecutePipeline.Pipeline">TransferToFtp-Start</parameter>
        <parameter name="HostURL">52.62.59.176</parameter>
        <parameter name="UserID">seafolly</parameter>
        <parameter name="Password">W2dReth$</parameter>
        <parameter name="FilePattern">^.*\.xml$</parameter>
        <parameter
name="SourceFolder">/src/FF/SeafollyUS/Customers</parameter>
        <parameter
name="TargetFolder">/root/dw_staging/upload/customer</parameter>
        <parameter name="SecureFtp">SFTP</parameter>
    </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
</flow>
<rules/>
<triggers>
    <run-recurring enabled="true">
        <recurrence>
            <date-from>2017-10-25Z</date-from>
            <start-time>16:15:00.000Z</start-time>
            <interval>1d</interval>
            <day-of-week/>
        </recurrence>
    </run-recurring>
</triggers>
</job>

<job job-id="CustomersSunburn" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="Sunburn"/>
        <step step-id="CustomersSunburn" type="ExecutePipeline"
enforce-restart="false">
            <description/>

```

```
<parameters>
    <parameter
name="ExecutePipeline.Pipeline">ExportCustomers-Start</parameter>
        <parameter name="SiteID">Sunburn</parameter>
            <parameter
name="ExportPath">/FF/Sunburn/Customers</parameter>
        </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
<step step-id="TransferSunburn" type="ExecutePipeline"
enforce-restart="false">
    <description/>
    <parameters>
        <parameter
name="ExecutePipeline.Pipeline">TransferToFtp-Start</parameter>
            <parameter name="HostURL">52.62.59.176</parameter>
            <parameter name="UserID">seafolly</parameter>
            <parameter name="Password">W2dReth$</parameter>
            <parameter name="FilePattern">^.*\.xml$</parameter>
        <parameter
name="SourceFolder">/src/FF/Sunburn/Customers</parameter>
        <parameter
name="TargetFolder">/root/dw_staging/upload/customer</parameter>
            <parameter name="SecureFtp">SFTP</parameter>
        </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
</flow>
<rules/>
<triggers>
    <run-recurring enabled="true">
        <recurrence>
            <date-from>2017-10-25Z</date-from>
            <start-time>16:15:00.000Z</start-time>
            <interval>1d</interval>
            <day-of-week/>
        </recurrence>
    </run-recurring>
</triggers>
```

```
</job>  
...  
</jobs>
```

Customers Import

- Overview
- Requirements
- Scenario
- Job Step Parameters
- Meta Data
 - Customer Groups

Overview

This module handles import of customer from uploaded file(s).

Functional specifications agreed with Fusion Factory:

[Mailing list subscription and Customer data flows for Seafolly.docx](#)

Requirements

All importing customers must have **Migrated** customer group assigned so they can be handled alter.

Scenario

FusionFactory uploads a file or files with pattern **FilePattern** into **FolderPath**. The Job executes ImportManager-ImportFiles pipeline which retrieve all files under **FolderPath** by **FilePattern**. These files are looped to ImportCustomers-Start pipeline where using bc_integrationframework they gets imported. If any error occurs, the job is canceled. Files with errors are moved into **ErrorPath**.

Job Step Parameters

1. **ImportPipeline**: ImportStores-Start (this should not be changed)
2. **FilePattern**: RegExp pattern for matching which files you want to be exported e.g. Seafolly_\d+.xml
3. **FolderPath**: Folder where your files was uploaded by FF e.g. /out/Seafolly/Customers
4. **ErrorPath**: Folder where your files with error will be moved e.g. /out/Seafolly/Customers/Errors

Meta Data

Customer Groups

Customer Group

```
<?xml version="1.0" encoding="UTF-8"?>
<customer-groups
  xmlns="http://www.demandware.com/xml/impex/customergroup/2007-06-30">
  ...
  <customer-group group-id="Migrated">
    <description>Represents all customers which were migrated before
    site launch.</description>
    <custom-attributes/>
  </customer-group>
  ...
</customer-groups>
```

Gift Vouchers Import

- Overview
- Requirements
- Scenario
- Job Step Parameters
- How to test Base64
- Example XML Data
- Meta Data
 - Jobs

Overview

This module processes xml files with valid gift voucher structure and send email to customer.

Requirements

No additional requirements.

Scenario

Job process all files in **FolderPath** matching **FilePattern**. Each of the files is processed, extracting voucher data into useful objects. Then an email is sent to the customer for each of the vouchers.

If all vouchers are valid and there are no validation errors, current file is being deleted.

If there are some vouchers which are not valid, job process only valid ones and current file is being moved into **ErrorPath**.

If there's a fatal error during the job execution, no vouchers will be processed and current file is being moved into **ErrorPath**.

Job Step Parameters

1. **ImportPipeline**: ImportGiftVouchers-ImportXMLFromFile (this should not be changed)
2. **FilePattern**: RegExp pattern for matching which files you want to be exported e.g. ^SeafollyAU_\d+\.xml\$
3. **FolderPath**: Folder where your files was uploaded by FF e.g. /in/giftvouchers/SeafollyAU
4. **ErrorPath**: Folder where your files with error will be moved e.g. /in/giftvouchers/SeafollyAU/Errors

How to test Base64

To encode/decode Base64 strings, please use this online tool:

Base64 Decode: <https://www.base64decode.org/>

Base64 Encode: <https://www.base64encode.org/>

Example XML Data

SeafullyAU_1295382748.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<vouchers>
    <voucher>
        <order-id>00018413</order-id>
        <status>new</status>
        <voucher-pin>MDAyMTYwMzAxNDc=</voucher-pin>
        <code>MzAxNDcwMA==</code>
        <amount>25.30</amount>
        <expiry-date>10/22/2017</expiry-date>
    </voucher>
    <voucher>
        <order-id>00003101</order-id>
        <status>refunded</status>
        <voucher-pin>MDAyMTYwMzAxNDg=</voucher-pin>
        <code>MzAxNDgwMA==</code>
        <amount>10.00</amount>
        <expiry-date>11/25/2017</expiry-date>
    </voucher>
</vouchers>
```

Meta Data

Jobs

Jobs Meta Data

```
<?xml version="1.0" encoding="UTF-8"?>
<jobs xmlns="http://www.demandware.com/xml/impex/jobs/2015-07-01">
    ...
    <job job-id="GiftVouchersImportSeafollyAU" priority="0">
        <description/>
        <parameters/>
        <flow>
            <context site-id="SeafollyAU"/>
            <step step-id="GiftVouchersImportSeafollyAU"
type="ExecutePipeline" enforce-restart="false">
                <description/>
                <parameters>
                    <parameter
name="ExecutePipeline.Pipeline">FileManager-ImportFiles</parameter>
                    <parameter
name="ImportPipeline">ImportGiftVouchers-ImportXMLFromFile</parameter>
                    <parameter
name="FilePattern">^SeafollyAU_\d+\.\xml$</parameter>
                    <parameter
name="FolderPath">/in/giftvouchers/SeafollyAU</parameter>
```

```

        <parameter
name="ErrorPath">/in/giftvouchers/SeafollyAU/Errors</parameter>
    </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
</flow>
<rules/>
<triggers>
    <run-recurring enabled="true">
        <recurrence>
            <date-from>2017-10-25Z</date-from>
            <start-time>16:15:00.000Z</start-time>
            <interval>1d</interval>
            <day-of-week/>
        </recurrence>
    </run-recurring>
</triggers>
</job>

<job job-id="GiftVouchersImportSeafollySG" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="SeafollySG"/>
        <step step-id="GiftVouchersImportSeafollySG"
type="ExecutePipeline" enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">FileManager-ImportFiles</parameter>
                <parameter
name="ImportPipeline">ImportGiftVouchers-ImportXMLFromFile</parameter>
                <parameter
name="FilePattern">^SeafollySG_\d+\.\xml$</parameter>
                <parameter
name="FolderPath">/in/giftvouchers/SeafollySG</parameter>
                <parameter
name="ErrorPath">/in/giftvouchers/SeafollySG/Errors</parameter>
            </parameters>
            <rules>
                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
    </flow>
    <rules/>
    <triggers>
        <run-recurring enabled="true">

```

```

        <recurrence>
            <date-from>2017-10-25Z</date-from>
            <start-time>16:15:00.000Z</start-time>
            <interval>1d</interval>
            <day-of-week/>
        </recurrence>
    </run-recurring>
</triggers>
</job>

<job job-id="GiftVouchersImportSeafollyUS" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="SeafollyUS"/>
        <step step-id="GiftVouchersImportSeafollyUS"
type="ExecutePipeline" enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">FileManager-ImportFiles</parameter>
                <parameter
name="ImportPipeline">ImportGiftVouchers-ImportXMLFromFile</parameter>
                <parameter
name="FilePattern">^SeafollyUS_\d+\.xml$</parameter>
                <parameter
name="FolderPath">/in/giftvouchers/SeafollyUS</parameter>
                <parameter
name="ErrorPath">/in/giftvouchers/SeafollyUS/Errors</parameter>
            </parameters>
            <rules>
                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
    </flow>
    <rules/>
    <triggers>
        <run-recurring enabled="true">
            <recurrence>
                <date-from>2017-10-25Z</date-from>
                <start-time>16:15:00.000Z</start-time>
                <interval>1d</interval>
                <day-of-week/>
            </recurrence>
        </run-recurring>
    </triggers>
</job>

<job job-id="GiftVouchersImportSunburn" priority="0">
    <description/>
    <parameters/>

```

```
<flow>
    <context site-id="Sunburn"/>
    <step step-id="GiftVouchersImportSunburn"
type="ExecutePipeline" enforce-restart="false">
        <description/>
        <parameters>
            <parameter
name="ExecutePipeline.Pipeline">FileManager-ImportFiles</parameter>
            <parameter
name="ImportPipeline">ImportGiftVouchers-ImportXMLFromFile</parameter>
            <parameter
name="FilePattern">^Sunburn_\d+\.xml$</parameter>
            <parameter
name="FolderPath">/in/giftvouchers/Sunburn</parameter>
            <parameter
name="ErrorPath">/in/giftvouchers/Sunburn/Errors</parameter>
        </parameters>
        <rules>
            <on-exit status="ERROR">
                <stop-job/>
            </on-exit>
        </rules>
    </step>
</flow>
<rules/>
<triggers>
    <run-recurring enabled="true">
        <recurrence>
            <date-from>2017-10-25Z</date-from>
            <start-time>16:15:00.000Z</start-time>
            <interval>1d</interval>
            <day-of-week/>
        </recurrence>
    </run-recurring>
</triggers>
```

```
</job>
...
</jobs>
```

Newsletter Export

- Overview
- Requirements
- Scenario
- Job Step Parameters
- Meta Data
 - Jobs

Overview

This module handles export of newsletter custom objects to file.

Functional specifications agreed with Fusion Factory:

[Mailing list subscription and Customer data flows for Seafolly.docx](#)

Requirements

This module requires **Meta Data** from [Mailing list subscription](#).

Scenario

Exporting all EmailSubscription Custom Objects into a file, then remove them. Check [Mailing list subscription](#) Scenarios section to see when custom object is created.

Job Step Parameters

1. **SiteID**: SiteID e.g. SeafollyEU
2. **ExportPath**: containing export path in format /FF/ + SiteID + /Newsletters e.g. /FF/SeafollyAU/Newsletters

Meta Data

Jobs

Jobs Meta Data

```
<?xml version="1.0" encoding="UTF-8"?>
<jobs xmlns="http://www.demandware.com/xml/impex/jobs/2015-07-01">
    ...
    <job job-id="NewsletterSeafollyAU" priority="0">
        <description/>
        <parameters/>
        <flow>
            <context site-id="SeafollyAU"/>
            <step step-id="NewsletterSeafollyAU" type="ExecutePipeline"
enforce-restart="false">
                <description/>
                <parameters>
                    <parameter>
```

```

name="ExecutePipeline.Pipeline">ExportNewsletters-Start</parameter>
    <parameter name="SiteID">SeafollyAU</parameter>
    <parameter
name="ExportPath">/FF/SeafollyAU/Newsletters</parameter>
    </parameters>
    <rules>
        <on-exit status="ERROR">
            <stop-job/>
        </on-exit>
    </rules>
</step>
<step step-id="TransferSeafollyAU" type="ExecutePipeline"
enforce-restart="false">
    <description/>
    <parameters>
        <parameter
name="ExecutePipeline.Pipeline">TransferToFtp-Start</parameter>
            <parameter name="HostURL">52.62.59.176</parameter>
            <parameter name="UserID">seafolly</parameter>
            <parameter name="Password">W2dReth$</parameter>
            <parameter name="FilePattern">^.*\.xml$</parameter>
            <parameter
name="SourceFolder">/src/FF/SeafollyAU/Newsletters</parameter>
            <parameter
name="TargetFolder">/root/dw_staging/upload/newsletter</parameter>
                <parameter name="SecureFtp">SFTP</parameter>
            </parameters>
            <rules>
                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
    </flow>
    <rules/>
    <triggers>
        <run-recurring enabled="true">
            <recurrence>
                <date-from>2017-10-23Z</date-from>
                <start-time>16:00:00.000Z</start-time>
                <interval>1d</interval>
                <day-of-week/>
            </recurrence>
        </run-recurring>
    </triggers>
</job>

<job job-id="NewsletterSeafollySG" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="SeafollySG"/>
        <step step-id="NewsletterSeafollySG" type="ExecutePipeline">

```

```

enforce-restart="false">
    <description/>
    <parameters>
        <parameter
name="ExecutePipeline.Pipeline">ExportNewsletters-Start</parameter>
            <parameter name="SiteID">SeafollySG</parameter>
            <parameter
name="ExportPath">/FF/SeafollySG/Newsletters</parameter>
        </parameters>
        <rules>
            <on-exit status="ERROR">
                <stop-job/>
            </on-exit>
        </rules>
    </step>
    <step step-id="TransferSeafollySG" type="ExecutePipeline"
enforce-restart="false">
        <description/>
        <parameters>
            <parameter
name="ExecutePipeline.Pipeline">TransferToFtp-Start</parameter>
                <parameter name="HostURL">52.62.59.176</parameter>
                <parameter name="UserID">seafolly</parameter>
                <parameter name="Password">W2dReth$</parameter>
                <parameter name="FilePattern">^.*\.xml$</parameter>
            <parameter
name="SourceFolder">/src/FF/SeafollySG/Newsletters</parameter>
            <parameter
name="TargetFolder">/root/dw_staging/upload/newsletter</parameter>
                <parameter name="SecureFtp">SFTP</parameter>
            </parameters>
            <rules>
                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
    </flow>
    <rules/>
    <triggers>
        <run-recurring enabled="true">
            <recurrence>
                <date-from>2017-10-23Z</date-from>
                <start-time>16:00:00.000Z</start-time>
                <interval>1d</interval>
                <day-of-week/>
            </recurrence>
        </run-recurring>
    </triggers>
</job>

<job job-id="NewsletterSeafollyUS" priority="0">
    <description/>

```

```

<parameters/>
<flow>
    <context site-id="SeafollyUS"/>
    <step step-id="NewsletterSeafollyUS" type="ExecutePipeline"
enforce-restart="false">
        <description/>
        <parameters>
            <parameter
name="ExecutePipeline.Pipeline">ExportNewsletters-Start</parameter>
            <parameter name="SiteID">SeafollyUS</parameter>
            <parameter
name="ExportPath">/FF/SeafollyUS/Newsletters</parameter>
        </parameters>
        <rules>
            <on-exit status="ERROR">
                <stop-job/>
            </on-exit>
        </rules>
    </step>
    <step step-id="TransferSeafollyUS" type="ExecutePipeline"
enforce-restart="false">
        <description/>
        <parameters>
            <parameter
name="ExecutePipeline.Pipeline">TransferToFtp-Start</parameter>
            <parameter name="HostURL">52.62.59.176</parameter>
            <parameter name="UserID">seafolly</parameter>
            <parameter name="Password">W2dReth$</parameter>
            <parameter name="FilePattern">^.*\.xml$</parameter>
            <parameter
name="SourceFolder">/src/FF/SeafollyUS/Newsletters</parameter>
            <parameter
name="TargetFolder">/root/dw_staging/upload/newsletter</parameter>
            <parameter name="SecureFtp">SFTP</parameter>
        </parameters>
        <rules>
            <on-exit status="ERROR">
                <stop-job/>
            </on-exit>
        </rules>
    </step>
</flow>
<rules/>
<triggers>
    <run-recurring enabled="true">
        <recurrence>
            <date-from>2017-10-23Z</date-from>
            <start-time>16:00:00.000Z</start-time>
            <interval>1d</interval>
            <day-of-week/>
        </recurrence>
    </run-recurring>
</triggers>

```

```

    </job>

<job job-id="NewsletterSunburn" priority="0">
    <description/>
    <parameters/>
    <flow>
        <context site-id="Sunburn"/>
        <step step-id="NewsletterSunburn" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">ExportNewsletters-Start</parameter>
                <parameter name="SiteID">Sunburn</parameter>
                <parameter
name="ExportPath">/FF/Sunburn/Newsletters</parameter>
            </parameters>
            <rules>
                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
        <step step-id="TransferSunburn" type="ExecutePipeline"
enforce-restart="false">
            <description/>
            <parameters>
                <parameter
name="ExecutePipeline.Pipeline">TransferToFtp-Start</parameter>
                <parameter name="HostURL">52.62.59.176</parameter>
                <parameter name="UserID">seafolly</parameter>
                <parameter name="Password">W2dReth$</parameter>
                <parameter name="FilePattern">^.*\.\xml$</parameter>
                <parameter
name="SourceFolder">/src/FF/Sunburn/Newsletters</parameter>
                <parameter
name="TargetFolder">/root/dw_staging/upload/newsletter</parameter>
                <parameter name="SecureFtp">SFTP</parameter>
            </parameters>
            <rules>
                <on-exit status="ERROR">
                    <stop-job/>
                </on-exit>
            </rules>
        </step>
    </flow>
    <rules/>
    <triggers>
        <run-recurring enabled="true">
            <recurrence>
                <date-from>2017-10-23Z</date-from>
                <start-time>16:00:00.000Z</start-time>
                <interval>1d</interval>
            </recurrence>
        </run-recurring>
    </triggers>
</job>

```

```
    <day-of-week/>
  </recurrence>
</run-recurring>
</triggers>
```

```
</job>  
...  
</jobs>
```

Stores Import

- Overview
- Requirements
- Scenario
- Job Step Parameters
- Meta Data

Overview

This module handles import of stores from uploaded file(s).

Requirements

No additional requirements.

Scenario

FusionFactory uploads a file or files with pattern **FilePattern** into **FolderPath**. The Job executes ImportManager-ImportFiles pipeline which retrieve all files under **FolderPath** by **FilePattern**. These files are looped to ImportStores-Start pipeline where using bc_integrationframework they gets imported. If any error occurs, the job is canceled. Files with errors are moved into **ErrorPath**.

Job Step Parameters

1. **ImportPipeline**: ImportStores-Start (this should not be changed)
2. **FilePattern**: RegExp pattern for matching which files you want to be exported e.g. Seafolly_\d+.xml
3. **FolderPath**: Folder where your files was uploaded by FF e.g. /out/Seafolly/Customers
4. **ErrorPath**: Folder where your files with error will be moved e.g. /out/Seafolly/Customers/Errors

Meta Data

No additional requirements.

Transfer To FTP

- Overview
- Requirements
- Jobs
- Job Step Parameters

Overview

This module handles transfer files to FTP server.

Requirements

Integration Framework Cartridge

This cartridge requires **bc_integrationframework** cartridge to be present in cartridge path for all sites including Business Manager.

Jobs

This must be implemented as a Job Step after an export.

Job Step Parameters

1. **HostURL:** FTP HostURL
2. **UserID:** FTP UserID
3. **Password:** FTP Password
4. **FilePattern:** RegExp pattern for matching which files you want to be exported e.g. ^.*\.xml\$
5. **SourceFolder:** folder where files you want to upload are. Usually the value is /src + ExportPath
6. **TargetFolder:** folder located on remote FTP server where the files will be sent to
7. **SecureFtp:** whenever to user normal or secure FTP. Possible values are **FTP** or **SFTP**

Search Details

Page : Search-ShowContent

Content Asset ID : content-search-refinebar

Description: Show the refine search menu Items.

"DRESS"

Products Found (0) | Articles Found (2)

ID: * content-search-refinebar

REFINE SEARCH

- New Arrivals
- Womens
- Mens
- Top Sellers

About Us

It all started with a series of observations: eCommerce merchandising and marketing innovation is what generates revenue, yet most operations are spending 80% of their budgets simply maintaining current infrastructure. The pace of ecommerce accelerates daily, but most operations are scrambling only to stand still. Merchandisers and marketers are supremely frustrated, spending more time chasing outsourced providers and internal IT organizations than actually merchandising and marketing their own businesses. There had to be a better way...dresses [Read More](#)

Terms & Conditions of Sale

These terms of use set forth the basis on which you are permitted to access and use this website, salesforce.com dresses [Read More](#)

Page: No results search page :

Content Slot ID : no-hits-shop-by-category

Slot Template : slots/category/no-hits-shop-by-category.isml

Description : Bottom content below no hits search form - Category carousel

TRY A NEW SEARCH

Popular Search Terms: Bikini, Tankini, High neck, One piece...

ID: * no-hits-shop-by-category

Enabled: Yes

Default:

Description:

Slot Content

Content Type: * Category

Category: womens
newarrivals
mens
electronics

Add

Template: * slots/category/no-hits-shop-by-category.isml

Callout:

MENS >

ELECTRONICS >

TOP SELLERS >

NEW ARRIVALS >

Content Slot ID : noresult-banner

Content Assest : noresults-help

Content Assest : noresults-bottom

Single Page Checkout

Content assets

The SPC has the following content assets:

- **checkout-terms** - positioned bellow the "Place Order" button ([terms-secure.png](#))
- **secure-checkout** - GoDaddy Secured image (next to "Place Order" button) ([terms-secure.png](#))
- **checkout-footer**
- **checkout-paypal** - the PayPal confirmation text in the payments tab ([paypal.png](#))
- **checkout-afterpay** - the AfterPay confirmation text in the payments tab ([afterpay.png](#))
- **checkout-suscribe** - subscription text for unregistered user. Different for Seafolly & Sunburn
- **checkout-security-code** - the "What's this" popup in the credit card payments tab ([secure-code.png](#))

The used style sheet for the SPC and the confirmation pages is [_spcheckout.scss](#)

Other functionality

- Address Fields Constraints
- Shipping Countries
- Shipping Methods Preferences
- Tax Preferences

Address Fields Constraints

Site Preference setting called "**SPC address fields constraints**" is defined in **BM > Merchant Tools > Site Preferences > Custom preferences > Seafolly Configurations**.

The setting must be stored as a **JSON string** that consists of key-value pair defined as bellow:

```
{  
    "constraints": {  
        "firstName": 12,  
        "lastName": 18,  
        "address1": 30,  
        "address2": 30,  
        "city": 24,  
        "stateCode": 19,  
        "postalCode": 6  
    }  
}
```

The possible field names are:

- firstName
- lastName
- address1
- address2
- city
- stateCode
- postalCode

If the setting is empty or some of the address fields names are omitted - the constraint won't be applied and the field will be stored with the initial value.

Shipping Countries

The available shipping countries are defined as a site preference and can be found under **BM > Merchant Tools > Site Preferences > Custom preferences > Seafolly Configurations**

The setting is a JSON string in the following format, where the country ID is the ISO-2 code :

```
{ "countryID": [ "SG", "AU" ] }
```

If no countries are defined here, then there will be no countries in the SPC shipping address dropdown!

Shipping Methods Preferences

Shipping methods are defined in **BM > Merchant Tools > Ordering > Shipping Methods**.

There is one additional configuration in Site Preferences that is responsible for the visualization of the appropriate shipping methods on the basket & checkout pages. The merchant can define as many shipping methods as they like, but only the ones listed in the Site Preferences will be visible to the customers.

This additional setting "**Shipping Method Preferences**" can be found under **BM > Merchant Tools > Site Preferences > Custom preferences > Seafolly Configurations**.

The setting must be stored as a **JSON string** & the merchant can define different rules that are applicable for the already defined and configured shipping methods.

```
{
    "basketRule1": {
        "basketTotal": "100",
        "methodID": "002"
    },
    "basketRule2": {
        "basketTotal": "0,99.99",
        "methodID": "001,003"
    }
}
```

```
{
    "basketRule1": {
        "basketTotal": "min,max",
        "methodID": "ID1"
    },
    "basketRule2": {
        "basketTotal": "min,max",
        "methodID": "ID2,ID3"
    },
    "basketRule3": {
        "basketTotal": "min",
        "methodID": "ID4"
    }
}
```

Each rule consists of “**basketTotal**” & “**methodID**” property.

The “**basketTotal**” property can be a single float number or a comma separated list of 2 float numbers. They represent the “min” & “min and max values” of the basket total respectively. **If no “max” value is present in the configuration (i.e. only one value) it is expected that there is no limit and the method will be visible if the other rule is true.**

The “**methodID**” can be comma delimited list of shipping method IDs. **One shipping method ID must be used in only one rule.**

If there are no defined rules, then all available shipping methods will be returned to the customer!

Tax Preferences

Business Manager

Sites with gross type of taxation (**GST**) use the default tax functionality and have the following settings in **Merchant Tools > Ordering > Taxation Settings**. **The previously defined custom preference "Tax Percentage" in Seafolly Configurations is no longer used!**

General:

- Policy: Gross
- Map Address To: Jurisdiction ID
- Address Mapping 1: Country
- Address Mapping 2: Not used (this mapping is not applicable for mapping of type "Jurisdiction ID")

Tax Classes:

- exempt, standard

Tax Jurisdictions:

The defined jurisdictions are per country as follows:

- For Seafolly AU & Sunburn - AU, NZ
- For Seafolly SG - SG

See [tax-jurisdictions.png](#), [tax-jurisdictions-edit.png](#)

Tax Rates: the rates for that country. See [tax-rates.png](#)

Code changes

On the checkout it is presented as follows:

- Product prices - shown **with** the tax
- Subtotal - the amount of products in the cart **without** the tax
- GST - the amount of the **items only tax**

For net taxation AvaTax is used and the prices shown on the SPC are not reduced.

Site Switcher

- General
- Meta Data
 - Seafolly Shared Library

General

Meta Data

Seafolly Shared Library

Seafolly Shared Library

```
<?xml version="1.0" encoding="UTF-8"?>
<library xmlns="http://www.demandware.com/xml/impex/library/2006-10-31"
library-id="SeafollySharedLibrary">
    ...
    <content content-id="site-switch-content-footer">
        <display-name xml:lang="x-default">Site switch content
        footer</display-name>
        <online-flag>true</online-flag>
        <searchable-flag>false</searchable-flag>
        <page-attributes/>
        <custom-attributes>
            <custom-attribute attribute-id="body"
            xml:lang="x-default">&lt;p&gt;We only ship to limited
            countries.&lt;/p&gt;&#13;
            &lt;p&gt;Seafolly uses cookies to allow you to make full use of our
            website. View our &lt;a href="$url('Page-Show', 'cid',
            'privacy-policy')$"&gt;Privacy and Cookie
            Policy&lt;/a&gt;.&lt;/p&gt;</custom-attribute>
        </custom-attributes>
    </content>

    <content content-id="site-switch-content-header">
        <display-name xml:lang="x-default">Site switch content
        header</display-name>
        <online-flag>true</online-flag>
        <searchable-flag>false</searchable-flag>
        <page-attributes/>
        <custom-attributes>
            <custom-attribute attribute-id="body"
            xml:lang="x-default">&lt;h2&gt;Select your
            country&lt;/h2&gt;</custom-attribute>
        </custom-attributes>
    </content>
    ...
</library>
```

Storefront Homepage[Seafolly/Sunburn]

1. Homepage Main Carousel:

Slot ID : home-main

Content Type: Content Asset

Content Asset ID: homepage-main-carousel (here we adding the owl carousel slides)

- Text with in the images can be managed from BM
 - Using images of various dimensions for mobile, tab and desktop with foundation interchanger plugin
 -

Slot content template : slots/content/homepage/homepagemaincarousel.ism

Demo : <https://owlcarousel2.github.io/OwlCarousel2/demos/animate.html>

If only one slide is present, there is no navigation to next slide (No Arrows and Dots present).

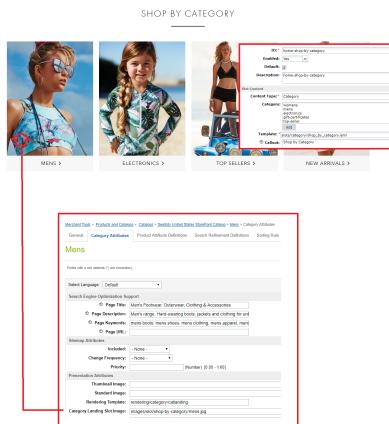
2. Shop by Category:

Slot ID : home-shop-by-category

Content Type: Category

Slot content template : slots/category/shop_by_category.isml

For category image configuration pls see below screenshot.



3. Shop New In:

Slot ID : home-shop-new-in

Content Type: Product

Slot content template : slots/product/product-shop-new-in.isml

4. Shop Social:

Slot ID : home-shop-social

Content Type: Content

Content Asset ID : home-shop-social

Slot Template : slots/content/homepage/home-shop-social.isml

5. Shop By Brand:

Slot ID: home-shop-by-brand

Content Type: Content Asset

Content Type: Content Asset
Content Asset ID : home-shop-by-brand

Content Asset ID : home-shop-by-brand
Slot Template : slots/content/homepage/home-shop-by-brand.isml

6. **Bottom Site Features Slot** (Last 3 column content with gray BG):

Slot ID: home-bottom-features-slot

Content Type: Content Asset

Content Asset IDs :

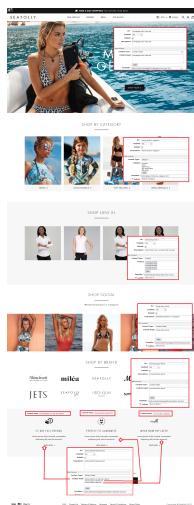
home-bottom-31-day-full-refund

home-bottom-perfect-fit

home-bottom-afterpay

Slot Template : slots/content/homepage/home-bottom-features-slot.isml

Positions and slot details :



Storefront Inline Editor (SIE) integration

I. Import META data

Upload to **SANDBOX/DEVELOPMENT/STAGING Storefront Inline Editor (SIE) META XML file `sie_system_object_type_extensions.xml`** from `int_inlineeditor/sites/sie_site_template/meta` and import it.

II. Import Content Block Templates in Libraries - MaajiSharedLibrary and SeafollySharedLibrary

Upload to **SANDBOX/DEVELOPMENT/STAGING files `sie_content_block_templates_Maaji.xml` and `sie_content_block_templates_Seafolly.xml`** from `int_inlineeditor/sites/sie_site_template/library` and import them.

III. Add to the end of path for each site you will use **Storefront Inline Editor (SIE)** - `int_inlineeditor`

IV. Update **Storefront Inline Editor (SIE)** preferences for each site. See screenshot below:

Salesforce Sandbox - dev02 Maaji Colombia

Merchant Tools Administration Storefront Toolkit Beta Cache Status OFF

Cancel Apply to Other Sites Save

Name	Value	Default Value	
Storefront Inline Editor Status	Yes	No	Edit Across Sites
Storefront Inline Editor Host*	dev02-ap01-seafolly.demandware.net		Use your SANDBOX or DEVELOPMENT or STAGING host here
Storefront Inline Editor OCAPI Client ID*	bfa0b2a1-dcf1-4c2a-b19d-cb2541042af9		Edit Across Sites
Storefront Inline Editor OCAPI Client Secret*	ff_team_2017		Edit Across Sites
OCAPI Version used by Storefront Inline Editor	18.1		Edit Across Sites
Root Images folder used by Slots	images		Edit Across Sites

© 2018 salesforce.com, inc. All Rights Reserved. Maaji Colombia Time Zone: Colombia Time | Instance Time Zone: Coordinated Universal Time | Version: 18.5 , last updated Apr 30, 2018 (Compatibility Mode: 16.2)

V. Update OCAPI configuration in Administration > Site Development > Open Commerce API Settings. Open file **ocapi_settings.json** in **int_inlineeditor/sites/sie_site_template/OCAPI** and copy/paste content as shown below:

Administration > Site Development > Open Commerce API Settings

Open Commerce API Settings

This page allows you to make client application-specific configurations of Open Commerce API resources, i.e. manage resource access privileges, attribute read/write permissions, or configure client application-specific response headers. You can select the actual type of the API that you want to configure and specify whether your settings are site-specific or global (organization-wide). Please note that due to caching, changes may take up to three minutes to become effective. You can browse the Open Commerce API here [API Explorer](#).

Select type: Data Select context: Maaji Colombia

```
{
  "_v": "18.1",
  "clients": [
    {
      "client_id": "bfa0b2a1-dcf1-4c2a-b19d-cb2541042af9",
      "resources": [
        {
          "resource_id": "libraries/content",
          "methods": ["get", "patch"],
          "read_attributes": "(**)",
          "write_attributes": "(*)"
        },
        {
          "resource_id": "sites/slots/slot_configurations",
          "methods": ["get", "patch", "delete"],
          "read_attributes": "(**)",
          "write_attributes": "(*)"
        }
      ]
    }
  ]
}
```

Save Reset

© 2018 salesforce.com, inc. All Rights Reserved. Maaji Colombia Time Zone: Colombia Time | Instance Time Zone: Coordinated Universal Time | Version: 18.5 , last updated Apr 30, 2018 (Compatibility Mode: 16.2)

VI. Configure slots to edit in storefront

To be able to edit slot with slot configuration of type HTML you have to:

1. Go to **Merchant Tools > Online Marketing > Content Slots**
2. Find the slot(s) that you want to be able to edit in storefront
3. Open it
4. Open every active slot configuration and update the custom attribute **Slot Configuration ID** to have the same value as slot configuration system attribute **ID**.

Salesforce - Sandbox - dev02
Merchant Tools Administration Storefront Toolkit Beta Cache Status OFF

Merchant Tools > Online Marketing > Content Slots > Slot 'header-banner' - 0

Slot Configurations

Slot Configuration(s)

Slot above the site header. Click New to create a new slot configuration, and click a specific slot configuration to edit that slot configuration. To delete a specific slot configuration, click the red deletion icon. To copy a specific slot configuration, select the slot configuration and click the Copy button.

ID	Description	Default	Status	Type	Preview
free2dayshiporderover300 (default)	HTML slot for Free 2 Day Shipping	True	Active	HTML	

New Copy

Slot Configuration - free2dayshiporderover300

The Description field is for an internal description, and the Callout field is for a storefront message. If Default is checked, the slot configuration is displayed when nothing else is scheduled. The Schedule defines when the slot configuration is displayed; the Rank decides which slot configuration displays if multiple slot configurations are running on the same schedule. If multiple slot configurations are scheduled and have the same Rank, the last edited slot configuration is displayed. The Content Type field enables you to define the type of content shown in the slot. The Product type is for specific products; Category is for all category products; Content Asset is for a specific content asset; and HTML is for plain text or HTML code.

Fields with a red asterisk (*) are mandatory.

Select Language: Default

ID*: free2dayshiporderover300

Enabled: Yes

Default:

Description: HTML slot for Free 2 Day Shipping

Content Type: HTML

HTML:

```
<div class="grid-x header-promotion-vr-spacer">
<div class="cell auto header-promotion-lft">Over 360 style combinations</div>
<div class="cell auto header-promotion">Free shipping on orders over $100 us</div>
<div class="cell auto header-promotion-right">Enjoy 31 day returns</div>
</div>
```

Template: slots/html/html[slotcontainer.isml]

Callout:

Custom

Slot Configuration ID: free2dayshiporderover300

Schedule/Qualifiers/Compatibility

Add Schedule

ID	Description	Type	Schedule	Customer Groups	Rank
Slot Default Schedule	Default	Site	Continuously		

<< Back to List

© 2018 salesforce.com, inc. All Rights Reserved. Maaji Colombia Time Zone: Colombia Time | Instance Time Zone: Coordinated Universal Time | Version: 18.5 , last updated Apr 30, 2018 (Compatibility Mode: 16.2)

Test it in storefront!

Store locator

Meta Data

The following custom attributes are added to **Store** system object under [Refinements] attribute group:

	ID	Name	Value
1	category	Category	Swimwear Overswim Kids
2	storeType	Type	Seafolly Stores DFO Stores Stockists
3	brand	Brand	Seafolly Milea JETS Maaji Sunseeker Camilla Miracle Suit Heidi Klum Swim Sunnyle Duskii Bondi Bathers Holster Sunsoaked Beach Candy Seafolly Girls Bondi Born

Configuration

New values can be added in **Business Manager**: Administration > Site Development > System Object Types > Store - Attribute Definitions > category, storeType and brand attribute.

Site preferences:

All custom preferences related to Store locator are located under Store Locator Configurations group.

ID	Name	Value
googleMapsApiKey	Google Maps API Key	api key
storeLookupUnit	Store Lookup Unit	miles kilometers
storeLookupMaxDistance	Store Lookup Max Distance	50 75 100 150 200
storeLookupExtendedMaxDistance	Store Lookup Extended Max Distance	number
maxNumberOfStoresIfNoResults	Number of suggestions(stores) to be shown if no results	number
maxNumberOfStores	Max number of stores	number
enableStoreLocatorFilterStores	Enable "Filter stores by..." option	Yes No
enableStoreLocatorCountryFilter	Enable "List by country" filter	Yes No
storeLocatorEnabledFilters	Enable only specific filters	storeType category brand
storeLookupDefaultLongitude	Store locator Default Longitude	default longitude for current site
storeLookupDefaultLatitude	Store locator Default Latitude	default latitude for current site

All custom attributes used as refinements must be added to **storeLocatorEnabledFilters** ([Enum of Strings](#)) site preference (Administration > Site Development > System Object Types > Site Preferences - Attribute Definitions > **storeLocatorEnabledFilters**).

Each preference value must be the same as the relevant custom attribute ID.

Display value is recommended to be the same as the relevant custom attribute Display Name.

After everything is configured as expected you can select only required refinements for specific site.

Workflow

List by county dropdown options are populated with data from countryCode system attribute.

Filter Stores dropdown options are populated with data from custom attributes under [Refinements] attribute group.

Filters can be enabled/disabled via the following preferences - **enableStoreLocatorFilterStores** and **enableStoreLocatorCountryFilter**.

storeLookupDefaultLongitude and **storeLookupDefaultLatitude** preferences are used for displaying default google map - map that is displayed when customer requests Store locator without performing any search.

Store locator provides the following types of search:

Search by user input or geolocation ("Find a store near me"):

1. When user searches by location(city/postal code) or clicks on "Find a store near me" - [**maxNumberOfStores**] stores within [**storeLookupMaxDistance**] [**storeLookupUnit**] radius should be displayed.
2. When user searches by location(city/postal code) or clicks on "Find a store near me" and there are no stores within [**storeLookupMaxDistance**] [**storeLookupUnit**] radius, then [**maxNumberOfStoresIfNoResults**] stores within [**storeLookupExtendedMaxDistance**] radius should be displayed even if it is not in [**storeLookupMaxDistance**] [**storeLookupUnit**] radius.
3. Filtering result – if user select some filter options a new request is made with selected filter and the result is updated.

Search by Country:

1. When user searches by country all stores located in the selected country are displayed.
2. User can select some filter options.

The working time is generated from the back-end according the following data format which will be provided by the middleware:

09:00 am - 5:30 pm, Mon - Wed
09:00 am - 9:00 pm, Thu
09:00 am - 5:30 pm, Fri
09:00 am - 5:00 pm, Sat
10:00 am - 5:00 pm, Sun

or

09:00 am - 5:30 pm, Mon - Sun

or

10:00 am - 6:00 pm, Mon - Thu
10:00 am - 9:00 pm, Fri-Sun

and so on.

TradeState Integration

Trade State has one cartridge: **int_tradestate_core**. It works with the custom objects and system objects. Read more information in the attachment "[TradeState Integration Document](#)".

[TradeState Integration Document.docx](#)