JMeter Scenarios for Platform

# CUSTOMER

Test Plan:

|  |  |  |
| --- | --- | --- |
| Scenario | API Endpoints | Method |
| Customer Creation/Registration | /stores/jdsportsuk/customers | POST |
| Get customer details | /stores/jdsportsuk/customers/${customerId} | GET |
| Customer Edit Profile | /stores/jdsportsuk/customers/${customerId} | PUT |
| Customer Login | /stores/jdsportsuk/customers/login | POST |

**Configuration:**

|  |  |  |
| --- | --- | --- |
| Number of Threads | Ramp up period | Loop Count |
| 1 | 1 | 1 |

Before going through request we can set the Base URL in HTTP Request Defaults.

Base URL : <http://commerce.mesh.mx.local>

Port: 80

Protocol : http

We can set the header related information of request in HTTP Header Manager.

x-api-key: 66666666666666666666666666666661

Loop controller:

**Configuration:**

Loop count: 1

**Scenario:**

**Customer Creation/Registration:**

PAYLOAD:

{

"email": "rajmacbook4@example.com",

"firstName": "raja",

"lastName": "rao",

"password": "raja8099"

}

Note:

* Only these fields are mandatory.
* Here email is unique.

We can extract the customer id using JSON Extractor for later purpose.

Configuration:

Names of created variables : customerId

JSON path expressions : $..ID

Match case : 1

**Get Customer Details:**

We can extract the email using JSON Extractor for later purpose:

Names of created variables : emailId

JSON path expressions : $.. email

**Update Customer Details:**

PAYLOAD:

{

"ID": "${customerId}",

"clientID": null,

"isActive": true,

"isGuest": false,

"gender": null,

"title": "",

"firstName": "Raja",

"lastName": "raoooo",

"email": "${emailId}",

"phone": "",

"mobile": "",

"otherPhoneNumber": "",

"preferredContactMethod": "phone",

"enrolledForEmailMarketing": false,

"enrolledForSMSMarketing": false,

"enrolledForPostageMarketing": false,

"enrolledForThirdPartyMarketing": false,

"registrationGiftcardSent": false,

"FBID": null,

"addresses": [],

"dateLastPurchased": "2018-09-28T11:59:36+0000",

"dateOfBirth": null,

"clientDateAdded": "2018-09-28T11:59:35+0000",

"isRegistered": true,

"canSetPrimaryAddress": true,

"canSetPrimaryBillingAddress": true,

"canEditEmailAddress": true

}

Note:

* Only these fields are mandatory.

**Customer Login :**

Payload :

{

"username" : "${emailId}",

"password" : "raja8099"

}

Results of Customer creation and login test plan(PHP 5.6 platform code)

No. of Threads : 1 user  
Ramp-up period(seconds) : 1  
Loop count : 1

Results :  
total samples : 4 (4 http requests, 1 user)  
Average response time : 1847 (milliseconds)  
Error(%) : 0.00%  
Average Throughput : 32.5/min

# CART

Test Plan:

|  |  |  |
| --- | --- | --- |
| Scenario | API Endpoints | Method |
| Guest Customer Creation/Registration | /stores/sizede/customers | POST |
| Create cart | /stores/sizede/carts | POST |
| Associate guest customer id to cart | /stores/sizede/carts/${cartId} | PUT |
| Get cart details | /stores/sizede/carts/${cartId} | GET |
| Add products to cart | /stores/sizede/carts/${cartId}/addProducts | PUT |
| Increase quantity of products in cart | /stores/sizede/carts/${cartId}/addProducts | PUT |
| Get cart details | /stores/sizede/carts/${cartId} | GET |
| Remove products from cart | /stores/sizede/carts/${cartId}/246002\_sizede | DELETE |
| Applying promotion code to cart | /stores/sizede/carts/${cartId}/discountCodes/OFFER\_15 | PUT |

**Configuration:**

|  |  |  |
| --- | --- | --- |
| Number of Threads | Ramp up period | Loop Count |
| 1 | 1 | 1 |

Before going through request we can set the Base URL in HTTP Request Defaults.

Base URL : <http://commerce.mesh.mx.local>

Port: 80

Protocol : http

We can set the header related information of request in HTTP Header Manager.

x-api-key: 66666666666666666666666666666661

Loop controller:

**Configuration:**

Loop count: 1

**Scenario:**

**Guest User Creation/Registration:**

PAYLOAD:

{

"firstName":"Test",

"lastName":" JMeter Customer",

"email":"LoadTest@a.co.uk",

"isGuest":true,

"phone":"00000000001",

"enrolledForEmailMarketing":true,

"addresses":

[

{

"title": "Mr",

"firstName": "Test",

"lastName": "Jmeter customer",

"address1": "OneIota",

"address2": "Pembrokeshire Technology",

"town": "Pembroke Dock",

"locale":"gb",

"email": "a@a.com",

"postcode":"SA72 6UN",

"phone": "000000000001",

"isPrimaryAddress": true,

"isPrimaryBillingAddress": true

}

]

}

We will extract the guest id using JSON Extractor for later purpose.

Configuration:

Names of created variables : guestId

JSON path expressions : $..ID

Match case : 1

**Cart Creation :**

We have to insert this query in stockListings table.

INSERT INTO `e2e\_commerce`.`stockListings` (`SKU`, `storeID`, `quantity`) VALUES ('799467', '43', '100');

We have to insert this query in toggles table.

INSERT INTO `e2e\_commerce`.`toggles` (`name`, `value`, `allowsOverrides`) VALUES ('USE\_OLD\_CART\_FUNCTIONALITY', '0', '1');

Payload:

{

"contents" :

[

{

"$schema":"http://commerce.mesh.mx/schemas/base/common/schema/CartProduct",

"SKU" : "285610\_sizede",

"quantity" : 1

}

]

}

We will extract the cart id using JSON Extractor for later purpose.

Configuration:

Names of created variables : cartId

JSON path expressions : $..ID

Match case : 1

**Associate guest user to cart :**

Payload:

{

"customer": {

"id": "http://commerce.mesh.mx.local/stores/sizede/customers/${guestId}"

}

}

**Get cart details:**

**Add Products to cart:**

Payload:

[

{

"SKU" : "246002\_sizede",

"quantity" : 1

}

]

**Increase Quantity of products in cart:**

Payload:

[

{

"SKU" : "246002\_sizede",

"quantity" : 1

}

]

**Get cart details:**

**Remove products from cart:**

**Apply promotion code to cart:**

Payload:

{

"user": null,

"promotionCode": "OFFER\_15",

"customerID": null

}

Promotion code OFFER\_15 applied to cart. we can check the cart details. Before going to apply any promotion code first we have to create the promotion code. Link: <http://mesh-platform-docs.k8s.jdgroupmeshtest.cloud/routes/POST_stores_:storeName_promotions.html>

Results of Cart creation with products test plan(PHP 5.6 platform code)

No. of Threads : 1 user  
Ramp-up period(seconds) : 1  
Loop count : 1

Results :  
total samples : 11 (8 http requests, 1 user)  
Average response time : 4562 (milliseconds)  
Error(%) : 0.00%  
Average Throughput : 13.1/min

# DEIVERY METHODS

Test Plan:

|  |  |  |
| --- | --- | --- |
| Scenario | API Endpoints | Method |
| Create a delivery method for a fascia | /stores/sizede/deliverymethods | POST |
| Create cart | /stores/sizede/carts | POST |
| Choose the created delivery method into the created cart. | /stores/sizede/carts/${cartId}/delivery | PUT |
| Get customer details | /stores/sizede/carts/${cartId} | GET |

**Configuration:**

|  |  |  |
| --- | --- | --- |
| Number of Threads | Ramp up period | Loop Count |
| 5 | 5 | 1 |

Before going through request we can set the Base URL in HTTP Request Defaults.

Base URL : <http://commerce.mesh.mx.local>

Port: 80

Protocol : http

We can set the header related information of request in HTTP Header Manager.

x-api-key: 66666666666666666666666666666661

Loop controller:

**Configuration:**

Loop count: 1

**Scenario:**

**Created Delivery method:**

Payload:

{

"displayName": "Standard Delivery",

"clientID": "DEUEXP",

"locale": "de",

"eligibleForFreeDelivery": true,

"price": {

"amount": "8.99",

"currency": "EUR"

},

"type": "",

"storeWide": true,

"isBookable": false,

"isPremium": true,

"allowsCrossLocaleDelivery": false,

"isExclusivelyCrossLocale": false,

"leadTimeForDispatch": null,

"isNominated": false,

"metapackServiceCode": null,

"name": "Next Day Delivery",

"costPrice": null,

"freeDeliveryThreshold": null,

"modifiers": [],

"bookingOptions": null,

"channels": [],

"allowableDeliveryLocales": []

}

We can extract the delivery method id using JSON Extractor for later purpose.

Configuration:

Names of created variables : deliveryMethodId

JSON path expressions : $..ID

Match case : 1

**Create Cart:**

Payload:

{

"contents" :

[

{

"$schema":"http://commerce.mesh.mx/schemas/base/common/schema/CartProduct",

"SKU" : "285610\_sizede",

"quantity" : 1

}

]

}

We can extract the cart id using JSON Extractor for later purpose.

Configuration:

Names of created variables : cartId

JSON path expressions : $..ID

Match case : 1

**Associate Delivery method to Cart:**

Payload:

{

"deliveryMethodID": "${deliveryMethodId}"

}

**Get Cart Details:**

Results of Delivery methods test plan(PHP 5.6 platform code)

No. of Threads : 5 users  
Ramp-up period(seconds) : 5  
Loop count : 1

Results :  
total samples : 20 (4 http requests, total 5 users)  
Average Response time : 6435 (milliseconds)  
Average Error(%) : 0.00%  
Average Throughput : 42.1/min

# ORDER TRACKING

Test Plan:

|  |  |  |
| --- | --- | --- |
| Scenario | API Endpoints | Method |
| Customer login | /stores/ark/customers/login | POST |
| Load the customer orders | /stores/ark/orders/2 | GET |
| Choose a specific order and check the order status | /stores/ark/orders/track?orderID=${OrderId}&email=${customerEmailId} | GET |

**Configuration:**

|  |  |  |
| --- | --- | --- |
| Number of Threads | Ramp up period | Loop Count |
| 5 | 5 | 1 |

Before going through request we can set the Base URL in HTTP Request Defaults.

Base URL : <http://commerce.mesh.mx.local>

Port: 80

Protocol : http

We can set the header related information of request in HTTP Header Manager.

x-api-key: 66666666666666666666666666666661

Loop Controller:

**Configuration:**

Loop count: 1

**Scenario:**

**Customer login :**

Payload:

{

"username" : "anemailaddress@test.com",

"password" : "raja8099"

}

Note:

* We have already registered with that email.

We can extract the customer email id using JSON Extractor for later purpose.

Configuration:

Names of created variables : customerEmailId

JSON path expressions : $..email

Match case : 1

**Get Customer order details:**

Before going to get order details we have to place an order through Behat e2e test case scenario. Here I have run the scenario line 25 in OrderTrackingStatus.feature. I have placed an order and order id is 2. Now I can get the order details of order id with 2.

We can extract the order id using JSON Extractor for later purpose.

Configuration:

Names of created variables : OrderId

JSON path expressions : $..ID

Match case : 1

**Order tracking status:**

Results of Order tracking test plan(PHP 5.6 platform code)

No.of Threads : 1 user  
Ramp-up period(seconds) : 1  
Loop count : 3

Results :  
total samples : 9 (3 http requests, 1 user )  
Average response time : 1222 (milliseconds)  
Error(%) : 0.00%  
Throughput : 49.1/min