# Customer Review Case Study

Overview 1

REST endpoints 1

Repository Design And Sequence Diagram 2

Files Description 2

Sample Input/Output 3

Testing Tools 4

1. Overview :

This module implements product review feature using Oracle Commerce Restful API. Many customers can review the product and also a customer can review many products. The ProductReviewActor is implements this feature using below actor chains.

* getProductReviewsByCustomer : Get Product reviews for current logged in customer.
* getProductReviewsForProduct : Get Product reviews for product in request.
* addProductReview : Add new product review.

1. REST endpoints :

* getProductReviewsForProduct :

http://localhost:8080/rest/model/neo/productReviews/actors/ProductReviewActor/getProductReviewsForProduct

* getProductReviewsByCustomer :

http://localhost:8080/rest/model/neo/productReviews/actors/ProductReviewActor/getProductReviewsByCustomer

* addProductReview :

http://localhost:8080/rest/model/neo/productReviews/actors/ProductReviewActor/addProductReview

1. Repository Design:

ReviewDetails

Product

User

Diagram 1 : Repository Design

* ReviewDetails contains reviews details [productId, profileId and comment text]. This item descriptor is available in CustomerReviewRepository [Core Schema].
* User item have one to many relationship with ReviewDetails. A user can review multiple products. To achieve this reviews property is added to user item descriptor.
* Product item have one to many relationship with ReviewDetails. A product can be reviewed by multiple users. To achieve this reviews property is added to product item descriptor.

Sequence Diagram:

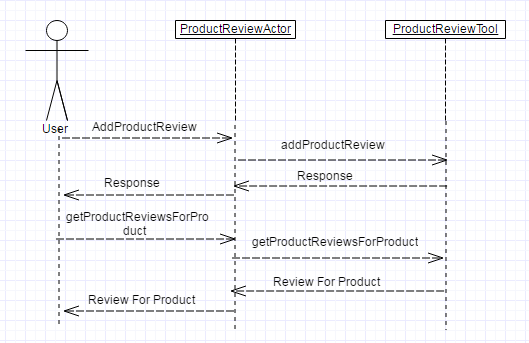


Diagram 2 : Add Review And Display Review For Product

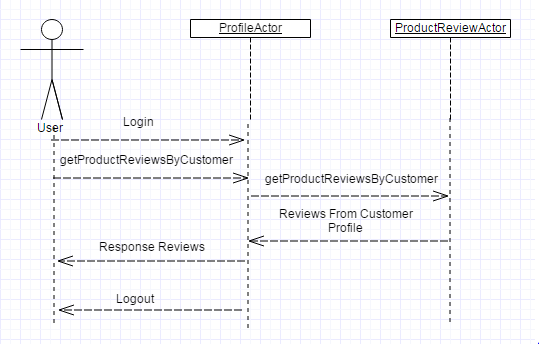


Diagram 3 : Logged In User Review

1. Files Description:

|  |  |  |
| --- | --- | --- |
| SN | File Name | Description |
| 1. | /CaseStudy/src/com/neo/review/constant/ProductReviewConstants.java | Contains constants used in this module |
| 2. | /CaseStudy/src/com/neo/review/tools/ProductReviewTool.java | Implements the core functionality [get/add reviews] |
| 3. | /CaseStudy/META-INF/MANIFEST.MF | Module Manifest file. |
| 4. | /CaseStudy/config/atg/commerce/catalog/custom/customCatalog.xml | Extended the product item-descriptor to include reviews. |
| 5. | /CaseStudy/config/atg/dynamo/service/filter/bean/beanFilteringConfiguration.xml | REST filter to control output of review. |
| 6. | /CaseStudy/config/atg/rest/registry/ActorChainRestRegistry.properties | Added custom chains in the Actor Chain registry [Login/logout and create user chains added for demonstration purpose only]. |
| 7. | /CaseStudy/config/atg/userprofiling/userProfile.xml | Extended the user item-descriptor to include reviews. |
| 8. | /CaseStudy/config/com/neo/review/tools/ProductReviewTool.properties | Component of /CaseStudy/src/com/neo/review/tools/ProductReviewTool.java. This is used in actor chains. |
| 9. | /CaseStudy/config/com/neo/review/CustomerReviewRepository.properties | New repository on core/production schema contains user/product review data. |
| 10. | /CaseStudy/config/com/neo/review/customerReviewRepository.xml | Definition file for customer review repository [mentioned above]. |
| 11. | /CaseStudy/config/neo/productReviews/actors/ProductReviewActor.properties | Product review Actor component. |
| 12. | /CaseStudy/config/neo/productReviews/actors/productReviewActor.xml | Defines actor chains for above component. |

1. Sample Input/Output:

## Get Product Reviews by Customer

The end point will return all the product reviews for the current logged in customer.

Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET** | <http://localhost:8080/rest/model/neo/productReviews/actors/ProductReviewActor/getProductReviewsByCustomer> |

Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | {"productReviews":[  { "productId": "xprod1013", "profileId": "user00001","comments": "Nice Product must buy" },  { "productId": "xprod2500", "profileId": "user00001", " comments ": "Good Product" } ] } |

## Get Product Reviews For Product

The end point will return all the product reviews for the productId passed.

Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | <http://localhost:8080/rest/model/neo/productReviews/actors/ProductReviewActor/getProductReviewsForProduct> |
|  | {"productId": "xprod1013" } |

Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | {  "productReviews":[  { "productId": "xprod1013", "profileId": "user00001", "comments": "Nice Product must buy" },  { "productId": "xprod1013", "profileId": "user00002", " comments ": "Didn’t like it " } ]  } |

## Add Product Review For Product

The end point will add a review for passed product

Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | <http://localhost:8080/rest/model/neo/productReviews/actors/ProductReviewActor/addProductReview> |
|  | { "productId": "xprod1013", "profileId": "user00001", "comments": "Nice Product must buy" } |

Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | {} |

1. Testing Tools: This feature can be tested/demonstrated either using HTTP Requester or even using curl.