

Estore Documentation

Ebarle, Roselle

Maglasang, Catherine

Yee, Mar Rynner

Esin, Dexter

April 22, 2016

Contents

1	Charter	1
1.1	Abstract	1
1.2	Vision	1
1.3	Mission	1
1.4	Objectives	1
1.5	Principles	2
1.6	Features	2
1.7	Business Rules	3
2	Stories	4
2.1	Roles	4
2.2	Role Attributes	4
2.3	Persona	4
2.3.1	User	4
2.3.2	Admin	4
2.4	User Stories	5
3	Use Cases	6
4	Test Cases	13

Chapter 1

Charter

1.1 Abstract

e-store is an open-source, python-based ecommerce platform for online merchants. It provides the power to grow your web business, reach more customers and sell more products and services. It enables businesses to experience an integrated workflow for their business - Accounting, Inventory and Order Management and Customer Service under one platform.

1.2 Vision

To be the most trusted integrated e-commerce platform for merchants

1.3 Mission

To develop a platform with utmost regards in security and customer satisfaction in the e-commerce industry

1.4 Objectives

1. Within 2 weeks after project initiation, the developers will already have a design and plan for the project's interface and architecture. The developers will come up with an initial product backlog and be able to commence the first sprint planning.
2. Within 7 weeks after planning, the team will finish the first sprint. And come up with 50 % of the planned major features and interface.
 - Within 1 week, the admin dashboard interface and functionalities such as adding products is created.
 - Then, Within 1 week, the customers are now able to login and register in the site.
 - Then, Within 1 week, the customers can now view the products and the product detail page.
 - Then, Within 2 weeks, the search feature is already implemented.
 - Then, Within 2 weeks, the add to cart functionality is created.
3. Within 5 weeks after the first sprint, the second sprint will be completed. The system will complete at least 90 % of its proposed major features.

- Within 1 week, the customers are now able to add product feedback (or ratings)
 - Then, within 1 week, the social sharing, add to favorites features are now implemented.
 - Then, within 1 week, the related products section and contact admin features are now implemented.
 - Then, within 1 week, the recommendation feature is already added.
 - Then, within 1 week, the currency converter feature is already added.
4. Within 1 week after the second sprint the main features decided by the team would be completed and tested correctly by the client.

1.5 Principles

Secure Transactions Scalability and Speed

1.6 Features

1. Admin Logout
2. Admin Login
3. Customer Login
4. Customer Logout
5. Customer Signin

Customer Dashboard

6. View my Orders
7. View my Wishlist
8. View my Reviews
9. Re-order items
10. Update Settings

Admin Dashboard

11. Add Staff
12. Update Staff
13. Revoke Staff Access
14. Add Order
15. View Customer Orders
16. Add Products
17. View Products
18. Add Category

19. View Category
20. Add Supplier
21. View Supplier
22. Add Customer
23. View Customers
24. View Sales Report
25. Add Promotion
26. Add Static page
27. Add Content Block

Store Catalog / Store Page

28. Filter Products
29. Product Social Sharing
30. Add to Wishlist
31. Search Product
32. Add Feedback
33. View Related Products
34. Customer Behaviour Tracking
35. Abandoned Cart
36. Checkout Payment
37. Automated Fulfillment

1.7 Business Rules

1. When signing up, the customer can only use the alphanumeric characters and the password has a required field of 6-22 valid characters
2. The email/username is unique to every account that is recorded in the database
3. When signing up, the information inputted must be complete such as the email/username, password, contact info, and confirm password
4. When filling up the form, the password and the confirm password must be the same
5. The username and password must be recorded on the database for the customer to be logged in
6. The customer and admin has different access rights
7. The customers not logged in will not be able to checkout their cart
8. When the admin adds a product all the information of the product must be complete
9. feedback for a product is limited to only 500 characters

Chapter 2

Stories

2.1 Roles

Customer - The customer browses and interacts within the store page

Admin - The admin is responsible in handling and managing orders and inventory in the admin dashboard.

2.2 Role Attributes

Customer:

Frequency of Use: Everyday

Domain Expertise: Moderate

Computer Expertise: Good

General Goals: Browse and buy products he need in the site.

Admin:

Frequency of Use: Everyday

Domain Expertise: Excellent

Computer Expertise: Excellent

General Goals: Manage orders and inventory

2.3 Persona

2.3.1 User

Akira loves shopping. Akira loves buying some rock magazines, manga, novels, gadgets, accessories and even the latest ones. Sometimes Akira is having a hard time dealing laziness because Akira doesn't like to go to a mall for shopping but he wants to buy some products from a mall. Akira found S-List. After finding S-List, Akira found out that buying products online is easier and Akira found out that there is no need to go to a mall for shopping.

2.3.2 Admin

Jane was a blogger who likes shopping. The owner of the website hired her to manage the website. Her job is to add a product, make a product be featured in the home page, add new categories for the

products, update the information of a product, and delete a product

2.4 User Stories

1. As an admin, I want to login to the website
2. As an admin, I want to be able to logout of the website
3. As a customer, I want to login to the website
4. As a customer, I want to be able to logout of the website
5. As a customer, I want to register to the website
6. As the customer, I want to be able to view the orders i've made so that I could track the status of my order.
7. As the customer, I want to view the items in my wishlist
8. As the customer, I want to view the reviews I submitted for the products I bought
9. As the customer, I want to be able to re-order products i've ordered in the past
10. As the customer, I want to update my profile settings
11. As an admin, I want to view the orders of my customers
12. As an admin, I want to add products
13. As an admin, I want to view all the products
14. As an admin, I want to view all my customers
15. As an admin, I want to view my sales report
16. As a customer, I want to filter the product by its brand and its price
17. As a customer, I want to share the product to social media sites
18. As a customer, I want to add the product to my wishlist
19. As a customer I want to search a particular product
20. As a customer, I want to add a product feedback
21. As a customer, I want to view the related products

Chapter 3

Use Cases

Use Case Element	Description
ID	1
Name	Get all products
Description	Get all product details from database
Primary Actors	Admin
Pre - Condition	User is logged in the system and has administrator privileges
Post - Condition	The product details are outputted in the product table in admin dashboard
Main Course	<ol style="list-style-type: none">1. The user accesses the product page in admin dashboard2. The client fetches the api resource url ‘/api/v1/products/’3. The api backend returns a json to the client containing the product details4. The client processes the json result from the api backend and outputs it in5. the product table6. The user sees the product and product details in the product table
Alternate Flows	
Exceptions	<ol style="list-style-type: none">3a. The api returns an empty array of products<ol style="list-style-type: none">1. The client will output a message saying ‘No products found’2. The user sees the message that no products are found
ID	2
Name	Get Single Product
Description	Get the details of a single product
Primary Actors	Admin
Pre - Condition	The user is logged in and the user has administrator privileges
Post - Condition	The user should see the details of the product in the product details page
Main Course	<ol style="list-style-type: none">1. The user accesses the product page2. The user clicks on the product id he wants to view3. The client fetches the api resource url ‘/api/v1/products/2’4. The api backend returns a json to the client containing the product details5. The client processes the json result from the api backend and outputs it in the table6. The user sees the product details in the product detail page
Alternate Flows	
Exceptions	<ol style="list-style-type: none">4a. The api backend returns no existing product<ol style="list-style-type: none">1. The client outputs a message saying that the product doesn’t exist

ID	3
Name	Create a Product
Description	Create a product in inventory / catalogue
Primary Actors	Admin
Pre - Condition	The user is logged in and the user has administrator privileges
Post - Condition	The product should appear in the product table
Main Course	<ol style="list-style-type: none"> 1. The user accesses the product page 2. The user clicks on the add product button 3. The user fills up the form in add new product page 4. The user clicks on submit 5. The client processes the form input and values 6. The client posts to the product resource url ‘/api/v1/products/’ and adds the form inputs as a data parameter in json form 7. The api backend processes the post request from the client 8. The api saves the new product in database 9. The client redirects to the product page 10. The user sees the new product in the product table
Alternate Flows	
Exceptions	<ol style="list-style-type: none"> 9a. The product already exists in the database <ol style="list-style-type: none"> 1. The api returns the error to the client 2. A message in the form should appear indicating that the product already exists
ID	4
Name	Update a Product
Description	Update a product’s details in inventory / catalogue
Primary Actors	Admin
Pre - Condition	The user is logged in and the user has administrator privileges
Post - Condition	The new product details should appear in the product table
Main Course	<ol style="list-style-type: none"> 1. The user accesses the product page 2. The user clicks on the edit button in the product row 3. The system redirects the user to the Edit Product Page 3. The user re-fills up the form and update the details as needed 4. The user clicks on submit 5. The client processes the form input and values 6. The client sends a put request to the product resource url ‘/api/v1/products/’ and adds the form inputs as a data parameter in json form 7. The api backend processes the ‘put’ request from the client 8. The api updates the product in database 9. The user is redirected to the product page 10. The user sees the new product details in the product table
Alternate Flows	
Exceptions	

ID	5
Name	Add a Supplier
Description	Add a supplier for the product
Primary Actors	Admin
Pre - Condition	The supplier to be added has a name, address, phone number, fax and email.
Post - Condition	The supplier is added in the database and can be viewed on the supplier list page
Main Course	<ol style="list-style-type: none"> 1. The admin is on the supplier page. 2. The admin Clicks on the add Supplier tab and is redirected to the add supplier form 3. The admin enters the name, address, phone number, fax, and email of the supplier 4. The Admin clicks on the submit button 5. The supplier is added and is saved in the database 6. There is a pop up message that says “the supplier was successfully added” 7. Exit
Alternate Flows	
Exceptions	<ol style="list-style-type: none"> 4.a There is a blank input box. <ol style="list-style-type: none"> 1.1 The admin will redirected to the add supplier form 1.2 There is a message that tells the admin to fill in all input boxes 1.3 Back to step 4 4b: Supplier already exists <ol style="list-style-type: none"> 2.1 There is a pop up message that says “Supplier already exists” 2.1 Back to step 3.
ID	6
Name	Update Supplier
Description	Update Supplier Information
Primary Actors	Admin
Pre - Condition	
Post - Condition	
Main Course	<ol style="list-style-type: none"> 1. The admin is on the supplier page. 2. The admin Clicks on the update Supplier button and is redirected to the update supplier form 3. The admin enters the name, address, phone number, fax, and email of the supplier 4. The Admin clicks on the submit button 5. The supplier is updated and is saved in the database 6. There is a pop up message that says “the supplier was successfully updated” 7. Exit
Alternate Flows	
Exceptions	<ol style="list-style-type: none"> 4a There is a blank input box. <ol style="list-style-type: none"> 1.1 The admin will redirected to the add supplier form 1.2 There is a message that tells the admin to fill in all input boxes 1.3 Back to step 4

ID	7
Name	View all Supplier
Description	View all supplier on the list
Primary Actors	Admin
Pre - Condition	The admin is on the Admin dashboard page
Post - Condition	The list of all suppliers is shown
Main Course	<ol style="list-style-type: none"> 1. The admin clicks on the supplier tab and will then be redirected to the supplier page. 2. The admin Clicks on the view Supplier tab and is redirected to the supplier list page 3. The Supplier list is shown 4. Exit
Alternate Flows	
Exceptions	
ID	8
Name	Add Order
Description	Add Customer Order
Primary Actors	Admin
Pre - Condition	The order to be added has a customer id, payment id, transaction date, shipping date and time stamp.
Post - Condition	The order is added in the database and can be viewed on the order list page
Main Course	<ol style="list-style-type: none"> 1. The admin is on the order page. 2. The admin Clicks on the add order tab and is redirected to the add order form 3. The admin enters the customer id, payment id, transaction date, shipping date and time stamp of the order 4. The Admin clicks on the submit button 5. The order is added and is saved in the database 6. There is a pop up message that says “the order was successfully added” 7. Exit
Alternate Flows	
Exceptions	<p>4a There is a blank input box.</p> <ol style="list-style-type: none"> 1.1 The admin will redirected to the add order form 1.2 There is a message that tells the admin to fill in all input boxes 1.3 Back to step 4 <p>4b: Order already exists</p> <ol style="list-style-type: none"> 2.1 There is a pop up message that says “Order already exists” 2.1 Back to step 3.

ID	9
Name	View all Orders
Description	View all orders on the list
Primary Actors	Admin
Pre - Condition	The admin is on the Admin dashboard page
Post - Condition	The list of all orders is shown
Main Course	<ol style="list-style-type: none"> 1. The admin clicks on the order tab and will then be redirected to the order page. 2. The admin Clicks on the view orders tab and is redirected to the supplier list page 3. The Supplier list is shown 4. Exit
Alternate Flows	
Exceptions	
ID	10
Name	Add Order Item
Description	Add an Order Item
Primary Actors	Admin
Pre - Condition	The order item to be added has an order id, product id, unit price, discount and quantity.
Post - Condition	The order item is added in the database and can be viewed on the order item list page
Main Course	<ol style="list-style-type: none"> 1. The admin is on the order page. 2. The admin Clicks on the add order item button and is redirected to the add order item form 3. The admin enters the order id, product id, unit price, discount and quantity 4. The admin clicks on the submit button 5. The order item is added and is saved in the database 6. There is a pop up message that says “the order item was successfully added” 7. Exit
Alternate Flows	
Exceptions	<ol style="list-style-type: none"> 4a. There is a blank input box. <ol style="list-style-type: none"> 1.1 The admin will redirected to the add supplier form 1.2 There is a message that tells the admin to fill in all input boxes 1.3 Back to step 4 4b: Order Item is already exists <ol style="list-style-type: none"> 2.1 There is a pop up message that says “Order item already exists” 2.1 Back to step 3.

ID	11
Name	Add Cart
Description	Add new Cart
Primary Actors	Admin
Pre - Condition	The cart has a session id, date created and customer id
Post - Condition	The new cart is added and saved in the database and can be viewed on the cart page
Main Course	<ol style="list-style-type: none"> 1. The admin is on the cart page 2. The admin clicks on the add cart tab and is redirected on the add cart form page 3. The admin inputs the cart session id, date created and customer id. 4. The admin clicks on the submit button 5. The new cart is added and saved in the database and can be viewed on the cart page 6. Exit

Alternate Flows

Exceptions	<ol style="list-style-type: none"> 4a. There is a blank input box. <ol style="list-style-type: none"> 1.1 The admin will redirected to the add supplier form 1.2 There is a message that tells the admin to fill in all input boxes 1.3 Back to step 4
------------	---

ID	12
Name	View All Carts
Description	View All Carts on the List
Primary Actors	Admin
Pre - Condition	The admin is on the Admin dashboard page
Post - Condition	The list of all carts is shown
Main Course	<ol style="list-style-type: none"> 1. The admin clicks on the cart tab and will then be redirected to the cart page. 2. The admin Clicks on the view carts tab and is redirected to the cart list page 3. The cart list is shown 4. Exit

Alternate Flows

Exceptions

ID	13
Name	View Wishlist
Description	View Products that was added in the Wishlist
Primary Actors	Customer
Pre - Condition	The customer is logged in and is in the customer dashboard
Post - Condition	The customer will be able to view the products he/she added to wishlist
Main Course	<ol style="list-style-type: none"> 1. The use case begins when the customer logs in the customer dashboard. 2. The user clicks on "My Wishlist" link in the navigation section. 3. The system displays all items in the wishlist of the user. 4. The user sees all items he added in wishlist. 5. The use case exits.

Alternate Flows

Exceptions

ID	14
Name	Add to wishlist
Description	Adding a product to the customer's wishlist
Primary Actors	Customer
Pre - Condition	The customer is logged in and is in the customer dashboard
Post - Condition	The specified product will be added to the costumer's wishlist
Main Course	<ol style="list-style-type: none"> 1. The customer clicks the heart icon on the product page. 2. The specified product will be reflected on the customer's wishlist. 3. The use case exits.
Alternate Flows	
Exceptions	<ol style="list-style-type: none"> 2a. If the specified product is already in the customer's wishlist <ol style="list-style-type: none"> 1. it shows an error message 2. Exit.
ID	15
Name	View Categories
Description	View all Categories
Primary Actors	Customer
Pre - Condition	The customer is logged in and is in the customer dashboard
Post - Condition	The customer will be able to view all categories
Main Course	<ol style="list-style-type: none"> 1. The use case begins when the customer logs in the customer dashboard. 2. The user clicks on "Categories" link in the navigation section. 3. The system displays all categories. 4. The user sees all the categories. 5. The user clicks the category button. 6. The user sees the product/s that is under a certain category. 7. The use case exits.
Alternate Flows	
Exceptions	
ID	16
Name	Add Category
Description	Add a new Category
Primary Actors	Admin
Pre - Condition	The admin is in the admin dashboard
Post - Condition	The category will be added in the category list
Main Course	<ol style="list-style-type: none"> 1. The admin is in the admin dashboard. 2. The admin clicks the add category button and will be displayed in the category page. 3. The admin clicks on the add button. 4. The category is added.
Alternate Flows	
Exceptions	<ol style="list-style-type: none"> 3a. If the category exists <ol style="list-style-type: none"> 1. it show an error message 2. Exit.

Chapter 4

Test Cases

ID	1
Scenario	Add Attribute
Given	I have the following data attribute_name validation default default
When	I save the data
Then	I get a "201" response
And	I get a field "status" containing "ok"
And	I get a field "message" containing "ok"
ID	2
Scenario	Update Attribute
Given	I have a resource with the id "1"
And	I want to update its data to the following data attribute_name validation default default
When	I update the data
Then	I get a "200" response
And	I get a field "status" containing "ok"
And	I get a field "message" containing "ok"
ID	3
Scenario	Create Cart Item
Given	I have the following data id cart_id product_id quantity time_stamp 1 1 1 1 2016-03-15 11:49:17
When	I Post the cart item to resource url '/api/v1/carts/1/items/'
Then	I should get response '200'
And	I should get "status" 'ok'
And	I should get "message" 'OK'

ID 4

Scenario Create duplicate cart item

Given I have the following data
 | id | cart_id | product_id | quantity | time_stamp |
 | 1 | 1 | 1 | 1 | 2016-03-15 11:49:17 |

When I Post the cart item to resource url '/api/v1/carts/1/items/'

Then I should get response '200'

And I should get "status" 'ok'

And I should get "message" 'ID EXISTS'

ID 5

Scenario Get cart item

Given cart item '1' is in the system

When I retrieve the cart item '1'

Then I should get response '200'

AND the following cart item details are returned:
 | cart_id | product_id | quantity | time_stamp |
 | 1 | 1 | 1 | 2016-03-15 11:49:17 |

ID 6

Scenario Get a cart item that doesn't exist

Given I retrieve the cart item '2'

When i retrieve JSON result

Then I should get response '200'

And I should get "status" 'ok'

And I should get a message containing 'No entries found'

And it should have a field "count" 0

And it should have an empty field " entries "

ID 7

Scenario Create cart

Given I have the following data
 |id | session_id | date_created| customer_id| is_active |
 |1 | 1 | 2016-03-15 | 1 | True |

When I Post the cart to resource url '/api/v1/carts/'

Then I should get response '200'

And I should get "status" 'ok'

And I should get "message" 'OK'

ID 8

Scenario Create Duplicate Cart

Given I have the following data
 |id | session_id | date_created| customer_id| is_active |
 |1 | 1 | 2016-03-15 | 1 | True |

When I Post the cart to resource_url '/api/v1/carts/'

Then I should have a status code '200'

And I should get a status 'ok'

And I should get a message 'ID EXISTS'

ID	9
Scenario	Get cart
Given	cart '1' is in the system
When	I retrieve the cart '1'
Then	I should have a status code '200'
And	the following cart details are returned : session_id date_created customer_id is_active 1 2016-03-15 1 True
ID	10
Scenario	Get a Cart that Doesn't Exist
Given	I retrieve the cart '2'
When	i retrieve a JSON result
Then	I should have a status code '200'
And	I should get a status 'ok'
And	it should have a field "message" 'No entries found'
And	it should have a field "count" 0
And	it should have an empty field " entries "
ID	11
Scenario	Get Customer
Given	customer id '1' is in the system
When	I retrieve the customer id '1'
Then	I get the customer '200' response
And	the following customer details are shown: id first_name last_name address city state postal_code country phone email user_id billing_address shipping_address date_created 1 first1 last1 address1 city1 state1 postalcode1 country1 phone1 test@estore.com 1 baddress1 saddress1 2016-03-11 11:49:17
ID	12
Scenario	Get Customer not in the Database
Given	I access the customer url '/api/v1/customers/2/'
When	I retrieve the customer JSON result
Then	I get the customer '200' response
And	it should have a customer field 'status' containing 'ok'
And	it should have a customer field 'message' containing 'No entries found'
And	it should have a customer field 'count' containing '0'
And	it should have an empty customer field 'entries'
ID	13
Scenario	Create Customer
Given	I have the following data id first_name last_name address city state postal_code country phone email user_id billing_address shipping_address date_created 9 first9 last9 address9 city9 state9 postalcode9 country9 phone9 test9@estore.com 9 baddress9 saddress9 2016-03-11 11:49:17
When	I POST to the customer url '/api/v1/customers/'
Then	I get the create customer '201' response
And	I should get a customer field 'status' containing 'ok'
And	I should get a customer field 'message' containing 'ok'

ID	14
Scenario	Create Duplicate Customer
Given	I have the following data id first_name last_name address city state postal_code country phone email user_id billing_address shipping_address date_created 9 first9 last9 address9 city9 state9 postalcode9 country9 phone9 test9@estore.com 9 baddress9 saddress9 2016-03-11 11:49:17
When	I POST to the customer url '/api/v1/customers/'
Then	I get the create customer '201' response
And	I should get a customer field 'status' containing 'ok'
And	I should get a customer field 'message' containing 'CUSTOMER EXISTS'
ID	15
Scenario	Create Customer with Missing Details
Given	I have the following data id first_name last_name address city state postal_code country phone email user_id billing_address shipping_address date_created 10 address9 city9 state9 country9 phone9 9 2016-03-11 11:49:17
When	I POST to the customer url '/api/v1/customers/'
Then	I get the create customer '201' response
And	I should get a customer field 'status' containing 'ok'
And	I should get a customer field 'message' containing 'error'
ID	16
Scenario	Add Image
Given	I have the following data item_id image_url caption 1 google.com hi
When	I save the data
Then	I get a "201" response
And	I get a field "status" containing "ok"
And	I get a field "message" containing "ok"
ID	17
Scenario	Update Image
Given	I have a resource with the id "1"
And	I want to update its data to the following data image_id item_id image_url caption 1 1 google.com hi
When	I update the data
Then	I get a "200" response
And	I get a field "status" containing "ok"
And	I get a field "message" containing "ok"

ID 18
 Scenario Add Item Attribute
 Given I have the following data
 | attribute_id | item_id | attribute_value |
 | 1 | 1 | Default |
 When I save the data
 Then I get a "201" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"

ID 19
 Scenario Update Item Attribute
 Given I have a resource with the id "1"
 And I want to update its data to the following data
 | attribute_id | item_id | attribute_value |
 | 1 | 1 | New Default |
 When I update the data
 Then I get a "200" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"

ID 20
 Scenario Add Item Variation
 Given I have the following data
 | item_id | option_id | stock_on_hand | unit_cost | re_order_level | re_order_quantity | is_active |
 | 1 | 1 | 100.00 | 10.00 | 100.00 | 100.00 | true |
 When I save the data
 Then I get a "201" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"

ID 21
 Scenario Update Item Variation
 Given I have a resource with the id "1"
 And I want to update its data to the following data
 | item_id | option_id | stock_on_hand | unit_cost | re_order_level | re_order_quantity | is_active |
 | 1 | 1 | 100.00 | 10.00 | 100.00 | 100.00 | true |
 When I update the data
 Then I get a "200" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"

ID 22
 Scenario Add Item
 Given I have the following data
 | name | description | date_added | date_updated | is_active |
 | name | description | 2001-1-1 1:1:1 | 2001-1-1 1:1:1 | true |
 When I save the data
 Then I get a "201" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"

ID 23
 Scenario Update Item
 Given I have a resource with the id "1"
 And I want to update its data to the following data
 | name | description | date_added | date_updated | is_active |
 | name | description | 2001-1-1 1:1:1 | 2001-1-1 1:1:1 | true |
 When I update the data
 Then I get a "200" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"

ID 24
 Scenario Add Location
 Given I have the following data
 | location_name |
 | name |
 When I save the data
 Then I get a "201" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"

ID 25
 Scenario Update Location
 Given I have a resource with the id "1"
 And I want to update its data to the following data
 | location_name |
 | new name |
 When I update the data
 Then I get a "200" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"

ID	26
Scenario	Add Option Group
Given	I have the following data option_group_name default
When	I save the data
Then	I get a "201" response
And	I get a field "status" containing "ok"
And	I get a field "message" containing "ok"
ID	27
Scenario	Update Option Group
Given	I have a resource with the id "1"
And	I want to update its data to the following data option_group_name default
When	I update the data
Then	I get a "200" response
And	I get a field "status" containing "ok"
And	I get a field "message" containing "ok"
ID	28
Scenario	Add Option
Given	I have the following data option_group_id option_value 1 default
When	I save the data
Then	I get a "201" response
And	I get a field "status" containing "ok"
And	I get a field "message" containing "ok"
ID	29
Scenario	Update Option
Given	I have a resource with the id "1"
And	I want to update its data to the following data option_group_id option_value 1 default
When	I update the data
Then	I get a "200" response
And	I get a field "status" containing "ok"
And	I get a field "message" containing "ok"

ID	30
Scenario	Create order item
Given	I have the following data id order_id item_id unit_price discount quantity 1 1 1 100.0 0.1 20
When	I Post the order item to resource_url '/api/v1/orders/1/items/'
Then	I should have a response '200'
And	I should have a "status" containing 'ok'
And	I should have a "message" containing 'OK'
ID	31
Scenario	Create a duplicate order item
Given	I have the following data id order_id item_id unit_price discount quantity 1 1 1 100.0 0.1 20
When	I Post the order item to resource_url '/api/v1/orders/1/items/'
Then	I should have a response '200'
And	I should have a "status" containing 'ok'
And	I should have a "message" containing 'ID EXISTS'
ID	32
Scenario	Create order item with incomplete details
Given	I have the following order item details id order_id item_id unit_price discount quantity 2 1 100.00 0.1 20
When	I Post the order item to resource_url '/api/v1/order_items/'
Then	I should have a response '200'
And	I should have a "status" containing 'ok'
And	I should have a "message" containing 'error'
ID	33
Scenario	Get an order item
Given	order item id '1' is in the system
When	I retrieve the order item '1'
Then	I should have a response '200'
And	the following order item details are returned: id order_id item_id unit_price discount quantity 1 1 1 100.0 0.1 20

ID	34
Scenario	Get an order item that doesn't exist
Given	I retrieve the order item '2'
When	I retrieve JSON result
Then	I should have a response '200'
And	I should have a "status" containing 'ok'
And	It should have a field "message " 'No entries found'
And	It should have a field "count " 0
And	It should have an empty field " entries "
ID	35
Scenario	Create order
Given	I have the following data id customer_id payment_id transaction_date shipping_date time_stamp transaction_status total 1 1 1 2016-03-11 2016-03-11 2016-03-11 11:49:17 Pending 100.0
When	I Post the order to resource_url '/api/v1/orders/'
Then	I should get a status of '200'
And	I should get a "status" 'ok'
And	I should get a "message" 'OK'
ID	36
Scenario	Create a duplicate order
Given	I have the following data id customer_id payment_id transaction_date shipping_date time_stamp transaction_status total 1 1 1 2016-03-11 2016-03-11 2016-03-11 11:49:17 Pending 100.0
When	I Post the order to resource_url '/api/v1/orders/'
Then	I should get a status of '200'
And	I should get a "status" 'ok'
And	I should get a "message" 'ID EXISTS'
ID	37
Scenario	Create an order with incomplete details
Given	I have the following data id customer_id payment_id transaction_date shipping_date time_stamp transaction_status total 2 2 2 2016-03-11 2016-03-11 2016-03-11 11:49:17 100.0
When	I Post the order to resource_url '/api/v1/orders/'
Then	I should get a status of '200'
And	I should get a "status" 'ok'
And	I should get a "message" 'error'

ID 38
 Scenario Get Order
 Given Order id '1' is in the system
 When I retrieve the order '1'
 Then I should get a status of '200'
 And the following orders are returned:
 | customer_id | payment_id | transaction_date | shipping_date | time_stamp |
 transaction_status | total |
 | 1 | 1 | 2016-03-11 | 2016-03-11 | 2016-03-11 11:49:17 |
 Pending | 100.0 |

ID 39
 Scenario Get an order that doesn't exist
 Given I retrieve the order '2'
 When I retrieve a JSON result
 Then I should get a status of '200'
 3 And I should get a "status" 'ok'
 And It should have a "message" "No entries found"
 And It should have a field "count" 0
 And It should have an empty field " entries "

ID 40
 Scenario Create Supplier
 Given I have the following data
 | id | name | address | phone | fax | email | is_active |
 | 1 | supplier1 | address1 | 221-2277 | 063-221-2277 | supplier1@estore.com | True |
 When I Post the supplier to resource_url '/api/v1/suppliers/'
 Then I should get a response '200'
 And I should get a "status" containing 'ok'
 And I should get a "message" containing 'OK'

ID 41
 Scenario Create duplicate supplier
 Given I have the following data
 | id | name | address | phone | fax | email | is_active |
 | 1 | supplier1 | address1 | 221-2277 | 063-221-2277 | supplier1@estore.com | True |
 When I Post the supplier to resource_url '/api/v1/suppliers/'
 Then I should get a response '200'
 And I should get a "status" containing 'ok'
 And I should get a "message" containing 'SUPPLIER EXISTS'

ID	42
Scenario	Create supplier with incomplete details
Given	I have the following data id name address phone fax email is_active 2 supplier1@estore.com True
When	I Post the supplier to resource_url '/api/v1/suppliers/'
Then	I should get a response '200'
And	I should get a "status" containing 'ok'
And	I should get a "message" containing 'error'
ID	43
Scenario	Get a supplier
Given	supplier '1' is in the system
When	I retrieve the supplier '1'
Then	I should get a response '200'
And	the following supplier details are returned: id name address phone fax email is_active 1 supplier1 address1 221-2277 063-221-2277 supplier1@estore.com True
ID	44
Scenario	Get a supplier that doesn't exist
Given	I retrieve the supplier '2'
When	I get the JSON result
Then	I should get a response '200'
And	I should get a "status" containing 'ok'
And	It should have a field "message" 'No entries found'
And	It should have a field "count" 0
And	It should have an empty field "entries"
ID	45
Scenario	Get User
Given	user id '1' is in the system
When	I retrieve the user '1'
Then	I get the '200' response
And	the following user details are shown: user_id username email password date_created is_admin 1 user9 user9@estore.com user9 1/1/1 1:1:1 true

ID	46
Scenario	Get User not in the Database
Given	I access the user id '2'
When	I retrieve the user JSON result
Then	I get the '200' response
And	it should have a user field 'status' containing 'ok'
And	it should have a user field 'message' containing 'No entries found'
And	it should have a user field 'count' containing '0'
And	it should have an empty field 'entries'
ID	47
Scenario	Create User
Given	I have the following user details: user_id username email password date_created is_admin 1 user9 user9@estore.com user9 1/1/1 1:1:1 true
When	I POST to the user url '/api/v1/users/'
Then	I get the create '201' response
And	I should get a user field 'status' containing 'ok'
And	I should get a user field 'message' containing 'OK'
ID	48
Scenario	Create Duplicate User
Given	I have the following user details: user_id username email password date_created is_admin 1 user9 user9@estore.com user9 1/1/1 1:1:1 true
When	I POST to the user url '/api/v1/users/'
Then	I get the create '201' response
And	I should get a user field 'status' containing 'ok'
And	I should get a user field 'message' containing 'USER EXISTS'
ID	49
Scenario	Create User with missing Details
Given	I have the following user details: user_id username email password date_created is_admin 1 user9@estore.com user9 1/1/1 1:1:1 true
When	I POST to the user url '/api/v1/users/'
Then	I get the create '201' response
And	I should get a user field 'status' containing 'ok'
And	I should get a user field 'message' containing 'error'

ID 50
 Scenario Create Wishlist Item
 Given I have the details of wishlist items
 | wishlist_item_id | wishlist_id | item_id | time_stamp |
 | 2 | 1 | 3 | 2016-04-14 |
 When I POST to url '/api/v1/wishlist_items/' the wishlist items
 Then I should get status code response '200'
 And I should get 'ok' status
 And I should get 'OK' message

ID 51
 Scenario Create a duplicate wishlist item
 Given I have the details of wishlist items
 | wishlist_item_id | wishlist_id | item_id | time_stamp |
 | 2 | 1 | 3 | 2016-04-14 |
 When I POST to url '/api/v1/wishlist_items/' the wishlist items
 Then I should get status code response '200'
 And I should get 'ok' status
 And I should get 'ERROR' message for duplication

ID 52
 Scenario Create an invalid wishlist item
 Given I have the details of wishlist items
 | wishlist_item_id | wishlist_id | item_id | time_stamp |
 | d | x | t | r |
 When I POST to url '/api/v1/wishlist_items/' the wishlist items
 Then I should get status code response '200'
 And I should get 'error' status for invalid details

ID 53
 Scenario Create an incomplete wishlist
 Given I have the details of wishlist items
 | wishlist_item_id | wishlist_id | item_id | time_stamp |
 | | 1 | | 2016-04-14 |
 When I POST to url '/api/v1/wishlist/' the wishlist
 Then I should get status code response '200'
 And I should get 'ok' status
 And I should get 'ERROR' message for incomplete details

ID 54
 Scenario Get Wishlist Item
 Given wishlist item '2' is in the system
 When I retrieve the wishlist '2'
 Then I should have a status code response '200'
 And the following details are returned :
 | wishlist_item_id | wishlist_id | item_id | time_stamp |
 | 2 | 1 | 3 | 2016-04-14 |

ID 55
 Scenario Get a wishlist item that doesn't exist
 Given I retrieve a wishlist item with id '4'
 When I retrieve the wishlist item JSON result
 Then I should have a status code response '200'
 And I should get the status says 'ok'
 And it should have a field message saying 'No entries found'
 And it should have a field count '0'
 And it should have an empty field 'entries'

ID 56
 Scenario Add Wishlist
 Given I have the following data
 | wishlist_id | wishlist_name |
 | 1 | default |
 When I save the data
 Then I get a "201" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"

ID 57
 Scenario Update Wishlist
 Given I have a resource with the id "1"
 And I want to update its data to the following data
 | wishlist_id | wishlist_name |
 | 1 | default |
 When I update the data
 Then I get a "200" response
 And I get a field "status" containing "ok"
 And I get a field "message" containing "ok"