

Sprint 2 Testing

Back-end

Test ID		1
Test Case Title		User front-end is able to get all active modules
Test Type		Functional
Objective		Verify that user front-end is able to get all active modules
Pro-condition		1. The admin front-end had created modules 2. The admin front-end should make these modules active.
Steps		1. Send the request to the server
Excepted Outcome		User front-end can get the list of all active modules
Acceptance Criteria	Given	Make the request
	When	I parse the request to the Eratos server
	Then	I should get the list of all active modules.
Test result		PASS
Notes		
tester		En Wen Tsai
Date		2021-5-26

Test ID		2
Test Case Title		user front-end is able to create a task
Test Type		Functional
Objective		Verify that the user front-end is able to create a task after finishing the payment.
Pro-condition		1. Login to get their user uri 2. Get the resource uri 3. Get the resource name 4. Get the payment ID & price from front-end 5. Get the geometry info from front-end
Steps		1. Take priority to this task 2. Send the request with these parameters
Excepted Outcome		Output success message with TaskID
Acceptance Criteria	Given	Received user's data
	When	I create a task based on provided data
	Then	I should get the success message.
Test result		PASS

Notes	
tester	En Wen Tsai
Date	2021-5-26

Test ID		3.1
Test Case Title		User/admin front-end is able to sync the task information
Test Type		Functional
Objective		Verify that the user front-end syncs the task information in our database with Eratos.
Pro-condition		1. the task status is incomplete (Queued or Processing)
Steps		1. Connect to back-end database 2. check the status is "Queued" or "Processing" 3. send the request to Eratos server.
Excepted Outcome		Output the success message "Database is up to date"
Acceptance Criteria	Given	check the task status and connect to the database
	When	I send the request to Eratos server
	Then	I can get the success message.
Test result		PASS
Notes		
tester		En Wen Tsai
Date		2021-5-26

Test ID		3.2
Test Case Title		User/admin front-end is able to get the task information
Test Type		Functional
Objective		Verify that the users can get the latest task status.
Pro-condition		<div>1. Sync the task info</div> <div>2. Get the user uri</div>
Steps		<div>1. Find the task information by user uri</div>
Excepted Outcome		Users can get the task information including task id, priority, state, type, resource order id and so on.
Acceptance Criteria	Given	User uri
	When	I use the user uri as parameter to find task information
	Then	I can get the string of all tasks information.
Test result		PASS
Notes		

tester	En Wen Tsai
Date	2021-5-26

Test ID		4.1
Test Case Title		User/admin front-end is able to sync the order information
Test Type		Functional
Objective		Verify that the users sync the order information in our database with Eratos.
Pro-condition		1. the order status is incomplete (Queued or Processing)
Steps		1. Connect to back-end database 2. check the order status is "Queued" or "Processing" 3. send the request to Eratos server.
Excepted Outcome		Output the success message "Database is up to date"
Acceptance Criteria	Given	check the order status and connect to the database
	When	I send the request to Eratos server
	Then	I can get the success message.
Test result		PASS
Notes		
tester		En Wen Tsai
Date		2021-5-26

Test ID		4.2
Test Case Title		User/admin front-end is able to get the order history
Test Type		Functional
Objective		Verify that the users can get the latest order history.
Pro-condition		1. Sync the order info 2. Get the user uri
Steps		1. Find the order information by user uri
Excepted Outcome		Users can get the order information including order id, price, status, order time, user id and payment id.
Acceptance Criteria	Given	User uri
	When	I use the user uri as parameter to find order information in the database
	Then	I can get the string of all order information.
Test result		PASS
Notes		
tester		En Wen Tsai

Date		2021-5-26
Test ID		5
Test Case Title		User/admin front-end is able to register as a new user
Test Type		Functional
Objective		Register new users for our back-end database
Pro-condition		1. get user data from front-end
Steps		1. insert these data to the back-end database.
Excepted Outcome		output the success message with username.
Acceptance Criteria	Given	user data from front end
	When	I get the user info from the front-end and then insert it to our database.
	Then	I should get the successful message.
Test result		PASS
Notes		
tester		En Wen Tsai
Date		2021-5-26

Test ID		6.1
Test Case Title		User front-end is able to the user information
Test Type		Functional
Objective		Verify that user front-end can get their user details from the database
Pro-condition		1. Get the user uri
Steps		1. Query with user uri to their user information.
Excepted Outcome		Output the success message with a string including EratosUserID, Email, Auth0ID, UserID, and name.
Acceptance Criteria	Given	Login to get user uri
	When	I query with valid user uri to the database
	Then	I should get the message about the user details.
Test result		PASS
Notes		
tester		En Wen Tsai
Date		2021-5-26

Test ID		6.2
---------	--	-----

Test Case Title		Admin front-end is able to all user information
Test Type		Functional
Objective		Admin front-end can get all user information
Pro-condition		1. Start index of user table 2. End index of user table 3. The range between start and end must be less than 100.
Steps		1. Query with start point and end point of the user ids.
Excepted Outcome		Output the success message with a string including eratosUserId, email, Auth0ID, UserID, and name. Each user is separated by “,”.
Acceptance Criteria	Given	Put the given start and end point
	When	I send the query to the database
	Then	I should get the message about specific user details.
Test result		PASS
Notes		[1] If an admin wants to get more than a hundred user information, he/she should query more than one time.
tester		En Wen Tsai
Date		2021-5-26

Test ID		7
Test Case Title		User/admin front-end is able to update their user information
Test Type		Functional
Objective		Users and admins can update user information
Pro-condition		1. Had already registered as an user/admin.
Steps		1. Make some changes in their personal information. ex: name, email, info or isAdmin.
Excepted Outcome		Output the success message “The user info is up to date.”.
Acceptance Criteria	Given	Make some changes
	When	I update these changes to the database
	Then	I should get the success message.
Test result		PASS
Notes		[1] Only admins can change the isAdmin parameter to true or false.
tester		En Wen Tsai
Date		2021-5-26

Test ID		8.1
Test Case Title		Admin front-end is able to create a resource/modules

Test Type		Functional
Objective		Verify that admins are able to create a new resource for users.
Pro-condition		
Steps		<ol style="list-style-type: none"> 1. Create a module name 2. Create a module schema 3. Make this module is active or not
Excepted Outcome		Get the success message
Acceptance Criteria	Given	Put needed parameter
	When	I send these parameters to the server
	Then	I should get the success message from the server
Test result		PASS
Notes		
tester		En Wen Tsai
Date		2021-5-26

Test ID		8.2
Test Case Title		Admin front-end is able to modify a resource/modules
Test Type		Functional
Objective		Verify that admins are able to modify the existing resource.
Pro-condition		<ol style="list-style-type: none"> 1. admin had already created a resource
Steps		<ol style="list-style-type: none"> 1. Change the module name 2. Change the module schema 3. Change the module status to inactive.
Excepted Outcome		Get the success message
Acceptance Criteria	Given	Put changed parameters
	When	I send these parameters to the server
	Then	I should get the success message from the server
Test result		PASS
Notes		
tester		En Wen Tsai
Date		2021-5-26

Test ID		9
Test Case Title		Admin front-end is able to get all available modules
Test Type		Functional

Objective		Verify that admin front-end is able to get the all available modules
Pro-condition		1. Start index of module id 2. End index of module id 3. The range between start and end must be less than 100.
Steps		1. Query with start point and end point of module id
Excepted Outcome		Output the success message with a string including Module id, Module name, Module schema, and IsActive or not. Each module is separated by “,”
Acceptance Criteria	Given	Put the given start and end point
	When	I send the query to the database
	Then	I should get the message about available modules' details.
Test result		PASS
Notes		[1] If an admin wants to get more than a hundred user information, he/she should query more than one time.
tester		En Wen Tsai
Date		2021-5-26