

Test case for Story 1:

Story ID	1.1	
Task	Check valid user's privilege/permission	
Test Type	Functional	
Objective	Safeguard user's activity in accordance to his/her authorized action	
Setup	1. Get a valid token from the Eratos server	
Expected Outcome	The function should return user details for a valid token.	
Acceptance Criteria	Given	a valid token
	When	I pass the token to the authorisation function
	Then	I should get the user details successfully
Test result	PASS	
Notes	The story 1 is separated into 2 sub-stories (valid users and invalid users), and this task is for testing valid users.	
Tester	Yan Dai	
Date	05/05/2021	

Story ID	1.2	
Task	Check invalid user's privilege/permission	
Test Type	Functional	
Objective	Prevent the random token from invalid users	
Setup	1. Generate a random token	
Expected Outcome	The function returns an error for a randomly generated token.	
Acceptance Criteria	Given	a randomly generated token
	When	I pass the token to the authorisation function
	Then	I should get an authorisation error
Test result	PASS	
Notes	The story 1 is separated into 2 sub-stories (valid user and invalid user), and this task is for testing invalid users.	
Tester	Deyou Zou	
Date	05/05/2021	

Test case for Story 2:

story ID	2.1	
Task	Create a resource	
Test Type	Functional	
Objective	Create resources for further operation	
Setup	<ol style="list-style-type: none">1. Get a valid token from the Eratos server2. Start the server and wait for request from the front-end3. Parse the request from front-end into valid type and name for resources	
Expected Outcome	Get details of newly created resources from the Eratos server	
Acceptance Criteria	Given	received a request from the front-end
	When	I parse the request and send it to the Eratos server
	Then	I should get the information about the newly created resource
Test result	PASS	
Notes	The story 2 is separated into 3 sub-stories. 2.2 and 2.3 will be done in sprint 2.	
Tester	En-Wen TSAI	
Date	05/05/2021	