ReGov GUI: Testing Manual

Task 1: Creating a new pod.

1. Head to the Pod creation screen from the Welcome Page.
2. Register a new pod by entering a pod name, type, default pod location, and public & private keys of the first ganache account.
3. Ideally, this should create a pod successfully.
4. If not, make a note of steps taken, inputs given, and error messages in the terminal for the Feedback report.
5. Repeat the process and test out the module with different inputs that may result in any possible error.

Task 2: Create a new resource.

1. Copy and Paste any image file inside a pod folder. This will be a resource for this pod.
2. Head to the Pod Manager screen, by opening the existing pod.
3. Head to the Register New Resource page and fill out the form to register the image resource to the pod.
4. Ideally, this should create a resource successfully.
5. If not, make a note of steps taken, inputs given, and error messages in the terminal for the Feedback report.
6. Create multiple resources for the same or different pods and test out the module with different inputs that may result in any possible error.

Task 3: Retrieve a pod’s resource successfully via an authorized API call.

1. Head to view the resource details screen and try retrieving the resource.
2. HTTP servers with desired port numbers can be turned on/off at the **Pod Management** screen(default port: *8080*) and try again to retrieve the resource.
3. Public keys can be whitelisted by adding them to the Access Control List.
4. Make sure your system is connected to the internet and select the name of the country from where you are accessing the app (most probably **Italy**) in the **Country Obligation** section.
5. Once all these steps are done, Click on the ‘*Get Resource*’ button to call the API.
6. Upon successful authorization, a new pop-up window will open with the resource image and its obligation rules.
7. If not, make a note of steps taken, inputs given, and error messages in the terminal for the Feedback report.

8. Repeat the process and test the service by giving different country names, port numbers, or anything else.

Task 4: Start resource usage policy monitoring and view/export logs

1. *Start Monitoring* to initiate the usage policy monitoring routine.
2. This will initiate a new monitoring and add a new entry to the log table.
3. Create more logs if desired and head to the Resource Logs screen.
4. A log table with entries for each monitoring routine can be seen. *View Detailed Logs* helps you to examine any particular log in detail.
5. To export any logs, select one or more rows and right-click and select Export > Export current selection, to export logs as a CSV file in the desired location.
6. If not, make a note of steps taken, inputs given, and error messages in the terminal for the Feedback report.
7. Repeat the process and test out the service with different inputs that may result in any possible error.

Task 5: Test all four resource obligations

1. Test out all four resource obligations of the pod or resource by giving different input values to each.
2. Ideally, the form validations must hold good and prompt appropriate error messages.
3. If not, make a note of steps taken, inputs given, and error messages in the terminal for the Feedback report.

General Note:

Upon the completion of the tasks please test out the whole system, and try to find any possible scenarios that may crash, poor error handling, etc. Test out different inputs for all 4 resource obligations for both pod and resource, create multiple pods and multiple resources for each pod, start multiple monitoring routines, test filtering or exporting logs, etc.

Feedback From Link: https://forms.gle/aY78h2LcmctYqM2y8

FYI :

- A resource without any resource obligation inherits its pod’s resource obligation.
- In Resource obligation, ***Unrestricted*** means that there are no restrictions or conditions for the resource or pod.
  For example: If both pod and resource had their country obligation as ***Unrestricted*** then no check is made on which country the service is being used.