**Why Scraping?**

When there is a quick need for your apps to use a backend that does not expose web services and has data in form of a webpage. In such cases, you can go ahead with screen scraping for retrieving data from your website dynamically.

The main advantage with screen scraping is a reduction in development and maintenance cost of web services.

**Use Case Scenarios:**

Scraper Tool is useful in the following scenarios:

* When no web services are available as an alternative
* When there is a business logic embedded in the web page; scraping fetches the data after the business logic is executed so that the logic on the client-side is minimized even if there are web services available
* To scrape other web sites in conjunction with an application. For example, a hotel application might want to extract data from a third-party partner site (a restaurant for example)
* When you want to extract data from a web page
* When you want to pass the extracted data from a web page to another service call
* When you want to monitor a website constantly and extract some information at regular intervals or trigger an action based on a change. For example, polling a newspaper site to extract news items and display on a gadget.

**Web Scraping using Kony Fabric:**

In this blog we describe how we can use popular web scrapping NodeJS library like [nightmare](https://github.com/segmentio/nightmare) along with Kony Fabric’s [Logic services](http://docs.kony.com/konylibrary/mobilefabric/kony_mobilefabric_user_guide/Content/Logic.htm)  to achieve headless screen scraping .

**Prerequisites:**

* Fabric version > 7.2
* NightmareJS package is installed in kony Logic services .( Please raise a request to [kony support](https://support.kony.com)  for enabling logic service and installing the Nightmare JS package)

Once the pre requisites are installed you are ready to use nightmareJS for screen scrapping. I have created a sample script to show you the basic scrapping scenario. You can download the sample package which contains scrapping scripts from [github](https://github.com/kony/KonyFabric_ScreenScrapingWithNodeJS).

The sample code is a simple script which logs into kony fabric console and returns the user who is using the account, as shown below:

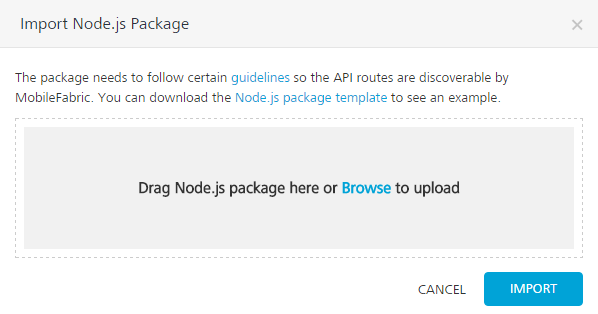


Follow below Steps to run the nightmare script:

1. Import and publish the nodeJS package(MFUser.zip) .
2. Import the Kony Fabric app(ScraperMFApp) to your kony fabric console.
3. Move to integration tab of the above App and give test values(your username and password for manage.kony.com account) for username and password parameters in header tab of request input of the getUser json service.
4. Now click “Fetch response” button you will get the user name of your manage.kony.com account

**Steps for publishing the package to Kony Fabric:**

1. In the Kony Fabric API Management > Logic, click the **IMPORT PACKAGE** button. The **Import Node.js Package** dialog appears.



1. In the **Import Node.js Package** dialog, drag the Node.js package from your local system and drop it to the dialog. You can also click **Browse** to select the package through the Microsoft Windows Open dialog. In the **Open** dialog, locate your exported package (for example, USERAPP.zip file), and select it. Click **Open**. In the Import Node.js Package dialog, the selected files from the package are added to the dialog.
2. Click **IMPORT**. The services from the package are imported to the **Logic** tab.
3. Click the **PUBLISH PACKAGE** button to publish the package to Node.js Runtime server.
4. Select the environment.
5. Click the **PUBLISH** button. The process of uploading the Node.js package to Node.js Runtime server begins.