**HKI Suaahara Data Import Manual**

* **Data Import Script Description:**

The script is written in JavaScript which uses different functions for importing data of different forms from commcare server to the dhis2 application.

We have compiled all the functions for different forms in one script which includes:

1. Register Household
2. Add Household members
3. Checklist 1,2 & 3
4. Fchv & Vmf checklist
5. Update Household Info
6. Update Household Member

These functions are written in such manner that they run one after the other traversing the commcare offset and date provided.

We have added these scripts in the standard reports of the dhis2 application and running it on production server using Node.js and PhantomJS.

We can run the script on daily basis which will import the data according to the functions written for different forms.

Suppose, if there is a condition like we only want to add only household member so what we can do is we can comment all other function calls and run the script providing the date and offset.

We have also added scripts for different functions by commenting all other function call and only calling those functions whose data which we need to import.

The scripts are available on the following link:

<https://github.com/hispindia/DHIS2-HKI>

* **Technologies used:**
* **Javascript**

Javascript is a scripting language developed by Netscape used to provide dynamic and interactive content on webpages. It is one of the first scripting languages in existence and today is the most popular scripting language on the web. Javascript works in all modern web browsers including Firefox, Netscape, Opera, Chrome and Internet Explorer.

* **Node.js**

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.

Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

For more information you can refer to nodejs documentation:

https://nodejs.org/en/docs/

* **PhantomJS**

PhantomJS is a new solution that provides headless testing of web applications. It is also a tool for dynamically capturing and rendering pages as images.

It allows you to programmatically manipulate page content to control and change it to different forms. It can scrape websites and save important information to files. It will also provide you network-level information of your page and site resources. These are just a few of the functions that PhantomJS can do for us. It provides a fresh and a whole new way for web designers, testers, and developers to perform and create browser-based solutions.

PhantomJS is not a library for NodeJS. It's a separate environment and code written for node is unlikely to be compatible. In particular PhantomJS does not expose a Common JS package loader.

For more information you can refer to PhantomJS documentation:

<http://phantomjs.org/documentation/>

* **Steps to run the Phantom scheduler**

The scheduler contains to files:

1. **package.json** : It provide all the dependencies to execute phantom js file
2. **phantomFile.js** : It contain the code to automatically execute and render the script.

**Steps:**

* Download nodejs from the link provided here <https://nodejs.org/en/download/>
* Download the scheduler folder containing the files mentioned above from the following link : <https://github.com/hispindia/DHIS2-HKI>

and copy them to local production server.

* Open command prompt to the location of the scheduler folder.
* Now run the following commands :

1. **npm install**

1. **npm start**

This schedular should be running all the time on the production server.