Users and Developers Manual for Enabling the system with new CLIENT ID:

To enable the google forms and Script API follow the steps given below,

1. Copy the files from ~/nbextensions/ma/client/qresources/ as Template.xlsx and CreateForm.json into your Google Drive.
2. Now open the documents in the drive by using Google Sheets and Apps Script application, this will generate their respective converted files.
3. In case you don't have App Script enabled in your drive, please enable it to execute the API.
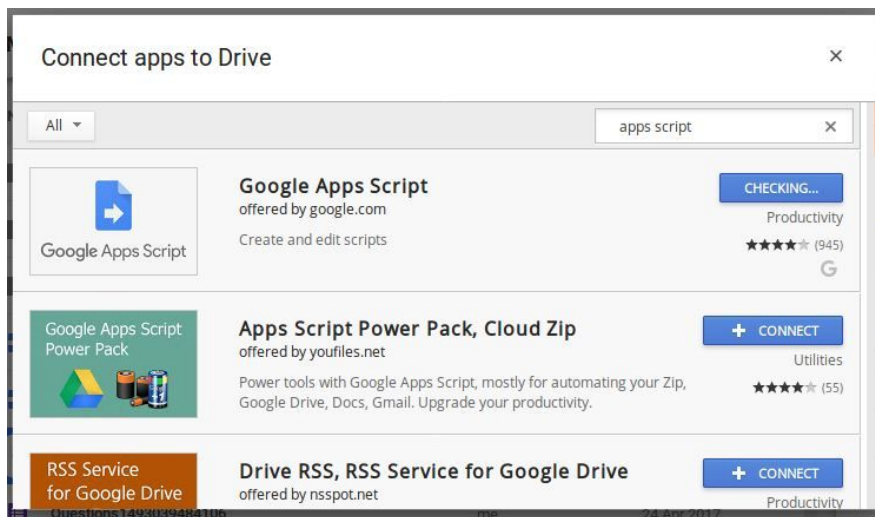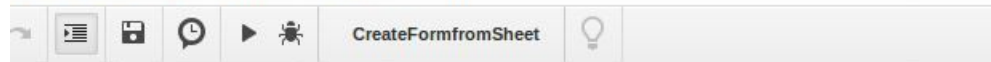

4.
5. Now as you have your documents running, copy and paste the Sheets ID and Script ID in
6. File ~/nbextensions/ma/client/common/js/googledrive/gdapi.js
7. SheetID:
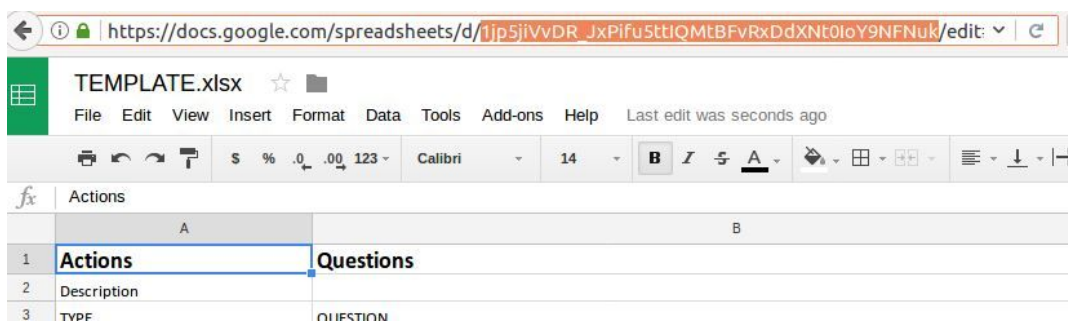
8. ScriptID:

9. Once it is done now update the Google API to run the the ColData environment on your system. Add resources to the Script to have seamless communication beetween the system and API. Select "Advanced Google Services".

10. Now Click on the highlighted field "Google API Console". It will redirect you to the library page of Google console and you can add the required libraries there.
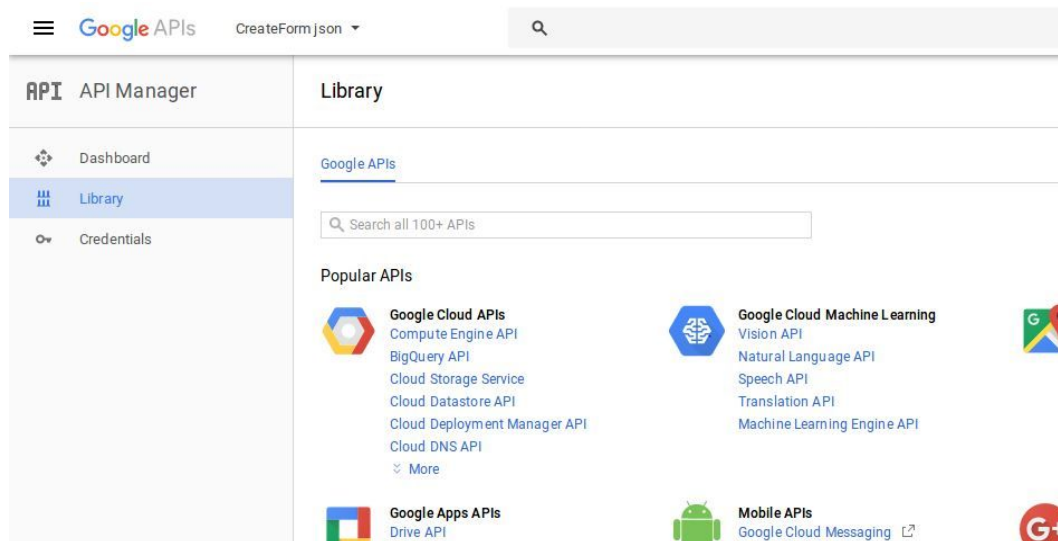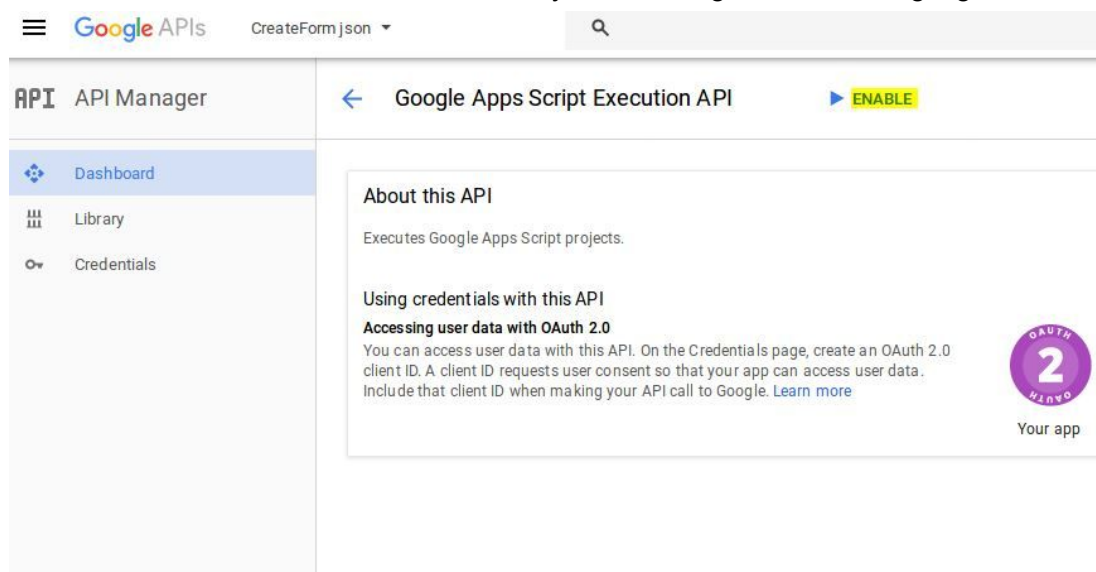
11. To run the whole application we need Drive API, Gmail API, and Google Apps Script Execution
   API.

12. Please   enable   the   API,   by   clicking   on   highlighted   field.

13. After you have added all the three required APIs you can verify those listed on the
    overview page (you can revisit the page to see the calls and responses, incase of any
    issue   with   API   integration).

14. On left pane of the page, click on "Credential" button to get the CLIENT ID for the entire API collection. For seamless communication and correct frame exchange between the system and Google, edit the Oauth id by clicking the edit button on right.
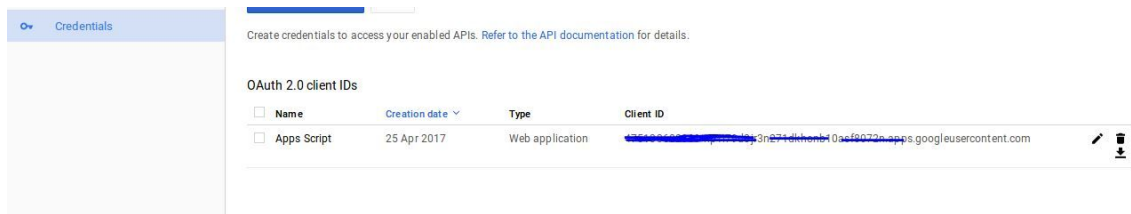
15. Add the Authorized JavaScript origin as "http://localhost:8888", and save it. Now update CLIENT ID in files: "~/nbextensions/ma/client/common/js/gmail/gmailapi.js" "~/nbextensions/ma/client/common/js/googledrive/gdapi.js" and into the folder on the local machine named "Git": "/home/developer/git/jupyter-drive/jupyterdrive/gdrive/gapiutils.js"

16. Now go to the Apps Script and deploy it as API(whenever you make any changes in the script deploy it as new version, it helps to revert back in future).

17. Now run the file to check if all services are added on automatic prompt select "allow". Do the same while running the ColData system for the first time. It will ensure that you have exception free system.
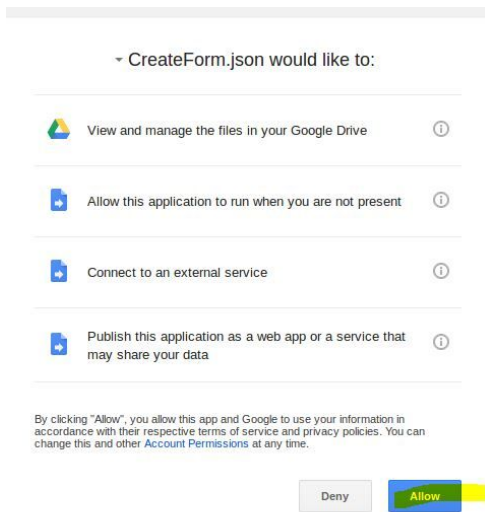
18. Keep the files intact in the Google Drive to use the functionality from the system using user friendly environment and a simple click.
19. If you have deleted the files from drive it is always possible to start fresh.