Google Apps Script Debugging using GAPPS

Tuesday, August 2, 2016 6:06 PM

This is a proposed workflow/framework for development of Google APPs scripts here at Dialog

Prerequisites

1. Not required, however all of the testing for this environment was performed on Chrome.
2. Node version .12.x or later installed on your machine
3. Node module for integrating with Google Drive installed

   ```
   npm install -g node-google-apps-script
   ```

4. `The documentation for node-google-apps-script is located at:`

   https://github.com/danthareja/node-google-apps-script

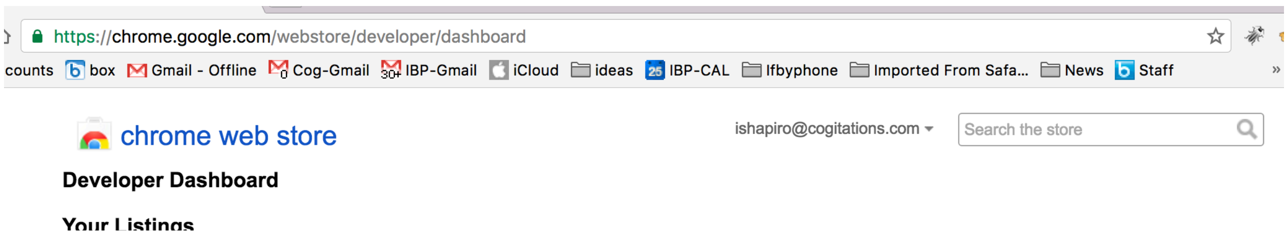
Getting Started

1. Set up your Google Developer Console environment

   https://chrome.google.com/webstore/developer/dashboard

2. Before accessing this link from Chrome you need to sign Chrome into a gmail accounts.

3. The first time you access this link you will be prompted to pay a $5 registration fee. You to gain access to all of the resources you will need to create a Google Add-On and use the


https://chrome.google.com/webstore/developer/dashboard

counts box Gmail - Offline Cog-Gmail IBP-Gmail iCloud ideas IBP-CAL Ifbyphone Imported From Safa... News Staff

chrome web store ishapiro@cogitations.com Search the store

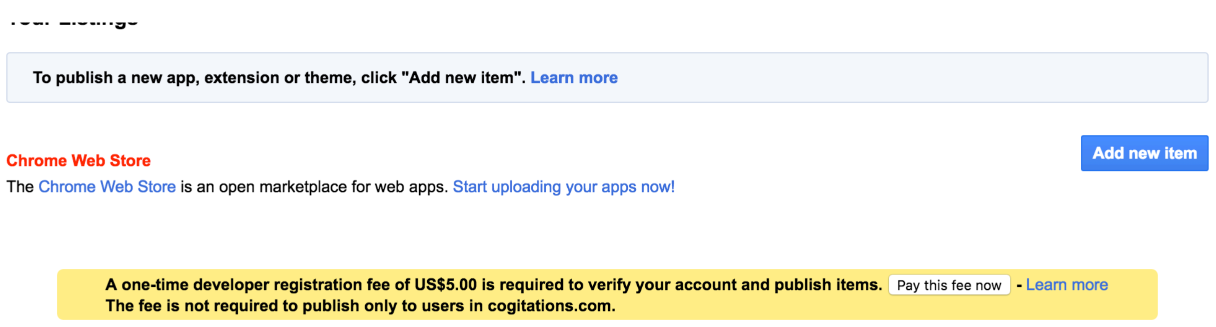
Developer Dashboard

Your Listings

Tech.

.

need to do this
e Google APIs.


To publish a new app, extension or theme, click "Add new item". Learn more

Chrome Web Store

The Chrome Web Store is an open marketplace for web apps. Start uploading your apps now!

Add new item

A one-time developer registration fee of US$5.00 is required to verify your account and publish items. Pay this fee now - Learn more
The fee is not required to publish only to users in cogitations.com.

4. Create a Google Developer Console Project

   https://console.developers.google.com/iam-admin/projects

   I named my project DialogTech Google Drive Access

5. Under the Library Tab -- add google drive API access. Click on enable.

6. Now click on Credentials on the left side of the page.
   VERY IMPORTANT. Click on Oauth Consent Screen on the top of the page.
   DO NOT CLICK ON create credentials.

7. The only field you need on the Oauth page is the email address. Then click on SAVE.

8. Now click on Create Credentials from the main screen.
   Select Oauth Client ID.
   Select other as the type.
   Given the credentials a name.
   Click on create.

9. Download the client secret file to your Google Scripts Directory as a json file. You cannot
   file without starting over so do not lose it.

Authenticating gapps (gapps is the node to google application)

1. Open a terminal window in your Google Scripts Directory on your Mac. (I use the "Go2Sh
   finder extension to make this easy to do.

2. Execute the following unix command (where the client_secret … is the file you download

led)

t recreate this

hell.app" OSX

led)

gapps auth client_secret_
130360900166-8g4u8u6d4br0bcpvu0qnf34mmfka7csr.apps.googleusercontent.com.jso

3. Copy the URL displayed and open it up in Chrome. This will display an Oauth security pa

4. Click on allow.

   OK. You are done the hard part.

Now Start/Initialize Your First Project

1. Navigate to your Apps Script project from Google Drive (must be a standalone script)

2. Get your project ID from the address bar, located after /d/ and before /edit.
   - For example, '//script.google.com/a/google.com/d/abc123-xyz098/edit?usp=drive

3. Navigate to a directory where your Apps Script project will live

4. Run gapps init <fileId> within your project directory.

   For example, gapps init abc123-xyz098

5. This will have create a .json configuration file and also a src directory. All of your module
   src directory renamed from .gs to .js. When you upload the files back to google they will
   back to .gs.

Ready for Coding Locally

1. Edit the local copy of your code in the src directory that was created by the gapps init co

2. When you are ready to update the Google Drive copy just type from the root directory n
   directory. The gapps app looks for the json config file in the current directory.

   gapps upload

3. Like magic your copy of the script in Google Drive will be updated.

n

ge.

e_web'

es will be in the
be renamed

mmand.

ot from the src

Usage Notes

1. The gapps tools does not currently have the ability to copy files back down from a google
   update them with the Google Apps UI. There is a pull request from a couple of months a
   feature but I have not investigated if it is stable.

2. Therefore once you start using gapps you must do ALL of your editing locally, run gapps u
   in the Google APPs Script editor interface. If you find a bug, make sure you go back and e
   re-upload.

   This is simular to using a DEV server when you would edit locally, ftp to test, and then ed
   can think of the Google APPs Script editor is the DEV server.

e drive if you
go for this

upload, then test
edit locally and

it locally. You