

Google Form Extension: Git File Upload

This is a Google Forms extension that uploads documents to a repository on submission of a form. The user should fill out the form and include their documents and upon submission the information and the document will be version controlled on an external Github repository

Table of Contents

- [Google Form Extension: Git File Upload](#)
 - [Table of Contents](#)
 - [Google Form Extension Developer Setup Guide](#)
 - [Prerequisites](#)
 - [Setup](#)
 - [Test Run](#)
 - [Important Commands](#)
 - [Important Links:](#)
 - [This is the link to the form where the code is deployed to](#)
 - [This is where the logs are viewable](#)
 - [This is where the deployed code resides](#)
 - [This is where you create OAuth 2.0 credentials](#)

Google Form Extension Developer Setup Guide

This guide will walk you through the setup process for creating a Google Form extension using Google Apps Script and CLASP. The extension allows you to add custom functionality and automation to your Google Forms.

Prerequisites

Before you begin, make sure you have the following prerequisites installed:

- **Node.js:** You can download and install Node.js from the official website: [Node.js](#)
- **CLASP:** CLASP (Command Line Apps Script Projects) is a command-line tool used for managing Google Apps Script projects. Install CLASP globally by running the following command:

```
npm install -g @google/clasp
```

- Google Clasp allows you to deploy the project as a google form extension

Setup

1. Clone the project from github [git@github.com:SociallyResponsibleComputing/Form-Extension.git](https://github.com/SociallyResponsibleComputing/Form-Extension.git)
2. Login to <https://console.cloud.google.com/apis/credentials?authuser=0&project=socially-responsible-computing> and create new oAuth2.0 credentials:

Start your Free Trial with \$300 in credit. Don't worry—you won't be charged if you run out of credits. [Learn more](#)

DISMISS [START FREE](#)

Google Cloud [Socially Responsible Computing](#) Search (/) for resources, docs, products, an... Search

APIs & Services

- Enabled APIs & services
- Library
- Credentials**
- OAuth consent screen
- Page usage agreements

Credentials [+ CREATE CREDENTIALS](#) [DELETE](#) [RESTORE DELETED CREDENTIALS](#)

Create credentials to access your enabled APIs. [Learn more](#)

API Keys

<input type="checkbox"/>	Name	Creation date ↓	Restrictions	Actions
No API keys to display				

OAuth 2.0 Client IDs

<input type="checkbox"/>	Name	Creation date ↓	Type	Client ID	Actions
<input type="checkbox"/>	MSI - Alex	Jun 28, 2023	Desktop	798789045867-anm6...	
<input type="checkbox"/>	Apps Script	Feb 8, 2023	Web application	798789045867-32mf...	
<input type="checkbox"/>	Apps Script	Feb 8, 2023	Web application	798789045867-jsgg...	
<input type="checkbox"/>	SRC form extension	Feb 8, 2023	Desktop	798789045867-6fki...	
<input type="checkbox"/>	Apps Script	Jan 11, 2023	Web application	798789045867-5eh3...	
<input type="checkbox"/>	Apps Script	Jan 11, 2023	Web application	798789045867-p2jg...	

Service Accounts [Manage service accounts](#)

Choose desktop Application when it asks you for the type of credentials

- Download the credentials as JSON and save it to the root directory of the project. Rename the credentials file to creds.json
- Inside the root directory there is a .clasp.json file, make sure the root directory points to the path of your cloned directory
- Next login using the freshly downloaded creds.json `clasp login --creds creds.json`

You should now be ready for development. Let's do a test to make sure everything is working

Test Run

- Open a terminal inside the project and run 'clasp run HelloWorld' and you should see the following output: "Hello World!"
- If that succeeds, you have successfully setup your development environment
- Check the logs to verify that the code ran:
https://script.google.com/u/0/home/projects/1EPqaSsc9UfkhVfQ_C6K4TjRY7PQYY0bvJq-37hcRZB2zg11DcwUzn4cP/executions

Important Commands

- clasp push** - This pushes the code to the google apps script project which effectively deploys the code
- clasp run <function>** - This runs the code on the google apps script project. It is used for testing functions in the app prior to deployment. You must specify a function to run as defined in the code

Important Links:

This is the link to the form where the code is deployed to

<https://forms.gle/v9kitcKe45XMq4SA6>

This is where the logs are viewable

https://script.google.com/u/0/home/projects/1EPqaSsc9UfkhVfQ_C6K4TjRY7PQYY0bvJq-37hcRZB2zg11DcwUzn4cP/executions

This is where the deployed code resides

https://script.google.com/u/0/home/projects/1EPqaSsc9UfkhVfQ_C6K4TjRY7PQYY0bvJq-37hcRZB2zg11DcwUzn4cP/edit?pli=1

This is where you create OAuth 2.0 credentials

<https://console.cloud.google.com/apis/credentials?authuser=0&project=socially-responsible-computing>