

# Git Search Project Initiation

Github API: *Latest:* <https://developer.github.com/v3/search/> *Alternate:* </v3/gists/>

Purpose: A need to Search Authors of Git Repositories has surfaced.

Start with: <https://github.com/TerbiumLabs/flask-developer-challenge/blob/master/README.md>

The core functionality will support a generic search: **search( query)**

DEFAULTS :

Order:	<b>Date, Newest First</b>	Oldest:	<b>2012-01-01</b>
Result Size:	<b>30 ??</b> MAX?	Page Size:	<b>30</b>

The GIT API will return up to 1000 paginated results. There is no way to limit the results to a particular size. So as many search options as possible should be used to keep the speed usable and to keep the results relevant.

An API wrapper to the core search function will serve to customize the search to our existing *and future* needs:

**searchAuthor( author, pattern, \*resultSize, \*pageSize)** ( *\*optional*)

**??** need "OrderBy" argument?

The Git Search API default order is by popular: RANK, which may not currently apply to our purposes.

**??** need "Oldest" argument? [&q=pushed:myDate](#)

"PageSize": [&per\\_page=myPageSize](#)

"ResultSize": if resultSize <= PageSize, then only the 1<sup>st</sup> page is stored -no pagination needed.

<https://help.github.com/articles/searching-repositories>

Only the existing needs warrant verification the first release of this service, however expansion is expected.

Error Handling: <https://developer.github.com/v3/#client-errors>

In keeping with MVC efficiency, arguments must be validated in the client view before this service is called. The client view argument validation is part of this project and will be contained in a separate file:

**gitSearch.js** (*Front-End / Client-Side*)

However, this service can also perform crude argument validation with a return status and a **NULL** result set. A NULL result set is also returned when the search criteria is not permissible or not found.

An error will be written to a server log in the case of invalid calls (*that should never happen*) to signal rate limit issues or to trigger the improvement of the client-side validation.

API Call Limits: <https://developer.github.com/v3/#rate-limiting>

- Not covered in this demo
- **??** Use an account with OAUTH2 to increase limits?
- **??** Use special headers so repeat calls for unchanged results will not subtract from our limit?

User Interface: *TBD ??*

Use an object from the Service Results that will fall into place within the Client-Side Form with minimal processing. A form with elements named the same as the result keys would be ideal.

Contrarily, the first demo (*without pagination*) used extra processing to format chosen values for each repository:

### GIT-SEARCH Demo

One Page of up to: 30 Repositories    Oldest Update: 2012-01-01    Order-By: Push Date, Newest First

Author:     Pattern:         Total Count: 2

```
[ID] 78130027    [NAME] hab-pkg-viz    [FULL-NAME] smith/hab-pkg-viz
[DESCRIPTION] Habitat Package Dependency Visualizer
[LAST-UPDATE] 2017-01-09T09:45:40Z
[URL] https://api.github.com/repos/smith/hab-pkg-viz
[LANGUAGE] Shell    [SIZE] 3    [WATCHERS] 1    [HAS-WIKI] true
```

```
[ID] 80471540    [NAME] hab-pkg-deb    [FULL-NAME] smith/hab-pkg-deb
[DESCRIPTION] Debian package exporter for Habitat artifacts
[LAST-UPDATE] 2017-01-30T22:47:18Z
[URL] https://api.github.com/repos/smith/hab-pkg-deb
[LANGUAGE] Shell    [SIZE] 17    [WATCHERS] 0    [HAS-WIKI] true
```

*Note: above draft shown for the demonstration of data results only*

Idea: Use a configurable mapping table with priorities assigned to key names to determine what to include.

Idea: An Excel Spreadsheet type plugin could be considered for styling

Pagination: <https://github.com/michaelbromley/angularUtils/blob/master/src/directives/pagination/>

Since all of the GIST API calls handle pagination in order to keep its own database hits under control, it may seem like there is not much benefit to utilize our own database for the same purpose. However, angular js, dirpagination will be used for the following reasons:

- Each row returned by GIST API has so many more values than we need. Passing this to the client is undesirable.
- Each repeat hit to successive pages deducts from our call limit.
- Its faster to use a local database in the case where the client navigates to a previously fetched page.
- The angular solution requires the most minimal amount of custom processing on the client side.

GIT Search API Pagination Handling: <https://developer.github.com/v3/#pagination>  
<https://developer.github.com/guides/traversing-with-pagination/>

```
Link:
<https://api.github.com/user/repos?page=1 >; rel="first",
<https://api.github.com/user/repos?page=3 >; rel="prev",
<https://api.github.com/user/repos?page=5 >; rel="next",
<https://api.github.com/user/repos?page=50 >; rel="last"
```

### Custom Database Handling:

If saved search results will not be reused, then all such rows should be deleted from the database at such time as this can be determined. The same request made after a *?? certain* time period will become stale automatically – whether it is known that the data has changed or not --because finding out if the data has changed would require the same hit to the GIT as fetching it in the first place.

Guidelines: Only the first page will be fetched and stored until the client initiates successive results. Each page will only be fetched from the external GIT API host and stored locally once.

# Github Search API V3 JSON data returned for the first repository in: response.items[0]

```
{
  "total_count": 2,
  "incomplete_results": false,
  "items": [
    {
      "id": 6047567,
      "name": "mbp51xx-tools",
      "full_name": "justdionysus/mbp51xx-tools",
      "owner": {
        "login": "justdionysus",
        "id": 2153161,
        "avatar_url": "https://avatars.githubusercontent.com/u/2153161?v=3",
        "gravatar_id": "",
        "url": "https://api.github.com/users/justdionysus",
        "html_url": "https://github.com/justdionysus",
        "followers_url": "https://api.github.com/users/justdionysus/followers",
        "following_url": "https://api.github.com/users/justdionysus/following{/other_user}",
        "gists_url": "https://api.github.com/users/justdionysus/gists{/gist_id}",
        "starred_url": "https://api.github.com/users/justdionysus/starred{/owner}/{repo}",
        "subscriptions_url": "https://api.github.com/users/justdionysus/subscriptions",
        "organizations_url": "https://api.github.com/users/justdionysus/orgs",
        "repos_url": "https://api.github.com/users/justdionysus/repos",
        "events_url": "https://api.github.com/users/justdionysus/events{/privacy}",
        "received_events_url": "https://api.github.com/users/justdionysus/received_events",
        "type": "User",
        "site_admin": false
      },
      "private": false,
      "html_url": "https://github.com/justdionysus/mbp51xx-tools",
      "description": "Utilities for reversing and modifying the MBP5120 and MBP5130 Blu-ray Player's firmware",
      "fork": false,
      "url": "https://api.github.com/repos/justdionysus/mbp51xx-tools",
      "forks_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/forks",
      "keys_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/keys{/key_id}",
      "collaborators_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/collaborators{/collaborator}",
      "teams_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/teams",
      "hooks_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/hooks",
      "issue_events_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/issues/events{/number}",
      "events_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/events",
      "assignees_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/assignees{/user}",
      "branches_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/branches{/branch}",
      "tags_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/tags",
      "blobs_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/git/blobs{/sha}",
      "git_tags_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/git/tags{/sha}",
      "git_refs_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/git/refs{/sha}",
      "trees_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/git/trees{/sha}",
      "statuses_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/statuses/{sha}",
      "languages_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/languages",
      "stargazers_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/stargazers",
      "contributors_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/contributors",
      "subscribers_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/subscribers",
      "subscription_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/subscription",
      "commits_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/commits{/sha}",
      "git_commits_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/git/commits{/sha}",
      "comments_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/comments{/number}",
      "issue_comment_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/issues/comments{/number}",
      "contents_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/contents/{+path}",
      "compare_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/compare/{base}...{head}",
      "merges_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/merges",
      "archive_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/{archive_format}/{ref}",
      "downloads_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/downloads",
      "issues_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/issues{/number}",
      "pulls_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/pulls{/number}",
      "milestones_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/milestones{/number}",
      "notifications_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/notifications{?since,all,participating}",
      "labels_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/labels{/name}",
      "releases_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/releases{/id}",
      "deployments_url": "https://api.github.com/repos/justdionysus/mbp51xx-tools/deployments",
      "created_at": "2012-10-02T16:11:16Z",
      "updated_at": "2013-12-12T09:04:17Z",
      "pushed_at": "2012-10-18T17:58:21Z",
      "git_url": "git://github.com/justdionysus/mbp51xx-tools.git",
      "ssh_url": "git@github.com:justdionysus/mbp51xx-tools.git",
      "clone_url": "https://github.com/justdionysus/mbp51xx-tools.git",
      "svn_url": "https://github.com/justdionysus/mbp51xx-tools",
      "homepage": null,
      "size": 120,
      "stargazers_count": 1,
      "watchers_count": 1,
      "language": null,
      "has_issues": true,
      "has_downloads": true,
      "has_wiki": true,
      "has_pages": false,
      "forks_count": 0,
      "mirror_url": null,
      "open_issues_count": 0,
      "forks": 0,
      "open_issues": 0,
      "watchers": 1,
      "default_branch": "master",
      "score": 1.0
    },
    ...
  ]
}
```

## Testing:

In keeping with IntroMalls development side testing standards:

- python unit tests ?? other unit tests? Code validation?
- cross-browser consistency: FireFox, Chrome, iOS, Android, IE, Edge, Safari and Opera. ?? versions?
- Browser console must be clean with FireFox javascript "strict mode" ON.
- Clean Server log files
- Friendly Reminder: Be sure to turn off Debug Mode in all sources before checking in.

## Load Testing:

This feature is considered to be a nice-to-have and thus will not be load tested. More importantly, it hits an external server which is not under our control. Load testing would not only be rude but would also run the risk of getting our host black-listed.

However, periodic executions of this service should be run during a load test of our main server to verify the affects to our server during a heavy load.

A new administrative variable: **gitSearchOn** default value: 1

can be flipped in an emergency situation to take effect immediately without restarting the server. A polite "*please try again later*" message will be displayed when this feature is attempted during the OFF state.

At a known internet speed comparable to the slower side of our average customer, manual tests should be performed in order to determine: MAX resultSize.

## Localization:

This demo has no multiple language support. Error messages are hard-coded in the javascript. Result sets are assumed to be in English. ?? Requirement? Gist API support?

Status: a Flask Stack Demo has been previewed by ?? and at least adequately serves as a **Proof-Of-Concept**.

No Stack –Browser Only Running On: <http://fdc.factorsmith.com/nostack/>

Tested POC Production Stack:

Service: Ubuntu 16.04 Linux, Flask, Python 2.7

UI: Ubuntu 16.04 Linux, Nginx, MySQL, HTML5 & Javascript (jQuery, Angular)

Running On: <http://fdc.factorsmith.com/>

Tested POC Development Stack:

Service: Windows XP SP3, Flask (eclipse plugin), Python 2.7

UI: Windows XP SP3, AMPPS (Apache), MySQL, HTML5 & Javascript (jQuery, Angular)

Final Production Stack: TBD ?? Remove the need for Flask:

- If Apache will be the Web Server, then FastCGI or SCGI can be used for scalability of Python scripts. It is recommended to avoid mod\_python ??  
How does mod\_wsgi scale ??
- Nginx can work with: uWSGI to serve python scripts --this may be ideal ??
- Keep in mind -most importantly, the existing **security issues** of the **mysql** statements utilized by default in the angular dirpagination package. Instead replace with the use of a relational database or better mysql handling, ie: PDO -if PHP will be used for front-end processing.

## Extra Setup:

```
$ pip install -U flask-cors
```

## Nginx Configuration:

/etc/nginx/conf.d/default.conf

```
server {
    listen 80;
    server_name flask.factorsmith.com;

    location / {
        proxy_pass http://127.0.0.1:8000;
    }
}
server {
    listen 80;
    root /var/www/html/fdc;
    server_name fdc.factorsmith.com;

    index index.html;
    access_log Off;
    error_log /var/www/html/fdc/error-fdc.log;

    location / {
        if (-f $request_filename) {
            expires 365d;
            break;
        }

        if (!-e $request_filename) {
            return 404;
        }
    }
}
```

File Structure: ( /include should be separated into: /img, /css, /js )

/var/www/html/fdc/index.html

/var/www/html/fdc/favicon.ico

/var/www/html/fdc/flask/**gistapi.py** (back-end service)

/var/www/html/fdc/include/demo.css

/var/www/html/fdc/include/asyncWait.gif

/var/www/html/fdc/include/**gitSearch.js** (user handling)

/var/www/html/fdc/include/jquery-2.1.4.min.js

/var/www/html/fdc/include/jquery-ui.css,

/var/www/html/fdc/include/jquery-ui.min.js

/var/www/html/fdc/include/jquery.highlighttextarea.css

/var/www/html/fdc/include/jquery.highlighttextarea.js

## Start Service:

python -m /var/www/html/fdc/flask/gistapi