



Endpoint

`/cms/getSkillsByAuthor.json`

```
▼ object {3}
  ▼ session {1}
    ▼ identity {3}
      type : host
      name : 172.68.146.136_ee0f4403
      anonymous : ☒ true
    accepted : ☒ true
  message : Author hasn't created any Skill yet.
```

Fetching Skills created by an author using author email

The Dashboard on the [SUSI Skill CMS](#) is the page where all user specific Skill information is being displayed. It also has a section called “My Skills” which displays the Skills created by the user himself. The data for the “My Skills” section has to be fetched from the Skill metadata of each Skill. Hence, this blog post explains how the Skills created by an author are fetched using an API which takes author’s email as the only parameter.

Why is such an API required?

The task of displaying the “My Skills” section cannot be done entirely on the Client side with the use of existing APIs, as that would result in a lot of AJAX calls to the Server, because for each Skill, there would have to be an AJAX call for each user. Hence, we had to create an endpoint which could directly return the list of Skills created by an author using his email. This would prevent multiple AJAX calls to the Server.

The endpoint used is:

```
"https://api.susi.ai/cms/getSkillsByAuthor.json?author_email="+author_email
```

Here, author_email is the email of the author who published the particular Skill, which is fetched from the Skill metadata object which looks like this:

```
"Antonyms": {
  // skill metadata
  "author_email": akjn11@gmail.com,
  // skill metadata
}
```

Working of the API

The code to fetch the Skills created by an author is described in the [GetSkillsByAuthor.java](#) file. This API accepts either the author's name, or the author's email as the parameter. In our

We get all folders within a folder using the following method:

```
private void listFoldersForFolder(final File folder, ArrayList<String> fileList) {
    File[] filesInFolder = folder.listFiles();
    if (filesInFolder != null) {
        Arrays.stream(filesInFolder).filter(fileEntry -> fileEntry.isDirectory() &&
!fileEntry.getName().startsWith(".")).forEach(fileEntry->fileList.add(fileEntry.getName(
) + ""));
    }
}
```

We use this method to get list of folders of all the group names for all the different Skills, and store them in a variable 'folderList' as per the following code:

```
File allGroup = new File(String.valueOf(model));
ArrayList<String> folderList = new ArrayList<String>();
listFoldersForFolder(allGroup, folderList);
```

For every folder corresponding to a group name, we traverse inside it and then further traverse inside the language folders to reach the place where all the Skill files are stored.

This is the code for the method to get all files within a folder:

```
private void listFilesForFolder(final File folder, ArrayList<String> fileList) {
    File[] filesInFolder = folder.listFiles();
    if (filesInFolder != null) {
        Arrays.stream(filesInFolder).filter(fileEntry -> !fileEntry.isDirectory() &&
!fileEntry.getName().startsWith(".")).forEach(fileEntry->fileList.add(fileEntry.getName(
) + ""));
    }
}
```

We need to get all Skill files inside the 'languages' folder, which itself is in 'group' folder. This is the code required to implement the same:

```
for (String temp_group_name : folderList){
    File group = new File(model, temp_group_name);
    File language = new File(group, language_name);
    ArrayList<String> fileList = new ArrayList<String>();
    listFilesForFolder(language, fileList);

    for (String skill_name : fileList) {
        // get skill metadata of each skill iteratively and put it in
        result JSONObject accordingly
    }
}
```

```
}
```

This is how the Skill metadata is fetched from the `getSkillMetadata()` method of [SusiSkill](#) class:

```
skill_name = skill_name.replace(".txt", "");
JSONObject skillMetadata = SusiSkill.getSkillMetadata(model_name,
temp_group_name, language_name, skill_name);

if(skillMetadata.get("author_email").equals(author_email)) {
    skillMetadata.put("skill_name", skill_name);
    authorSkills.put(skillMetadata);
}
```

`getSkillMetadata()` method takes model of the Skill, group of the Skill, language of the Skill, Skill name and also duration (for usage statistical purposes) as the method parameters, and returns the metadata for the Skill in the format as shown below:

```
"Antonyms": {
  "model": "general",
  "group": "Knowledge",
  "language": "en",
  "developer_privacy_policy": null,
  "descriptions": "A skill that returns the antonyms for a word",
  "image": "images/antonyms.png",
  "author": "akshat jain",
  "author_url": "https://github.com/Akshat-Jain",
  "author_email": "akjn11@gmail.com",
  "skill_name": "Antonyms",
  "protected": false,
  "terms_of_use": null,
  "dynamic_content": true,
  "examples": ["Antonym of happy"],
  "skill_rating": {
    "negative": "0",
    "bookmark_count": 0,
    "positive": "0",
    "stars": {
      "one_star": 0,
      "four_star": 0,
      "five_star": 0,
      "total_star": 0,
      "three_star": 0,
      "avg_star": 0,
      "two_star": 0
    }
  },
  "feedback_count": 0
},
"usage_count": 0,
"skill_tag": "Antonyms",
"creationTime": "2018-05-06T03:05:22Z",
"lastAccessTime": "2018-06-23T18:48:36Z",
"lastModifiedTime": "2018-05-06T03:05:22Z"
}
```

This is how the Skills created by an author are fetched using this API. The JSON response of this endpoint shows the metadata of all Skills which have 'author_email' value the same as that given in the endpoint's query parameter.

Resources

- List files and folders in a directory in JAVA - <http://www.codejava.net/java-se/file-io/how-to-list-files-and-directories-in-a-directory>
- For-each loop in JAVA - <https://www.geeksforgeeks.org/for-each-loop-in-java/>
- ArrayList in JAVA - <https://www.geeksforgeeks.org/arraylist-in-java/>

Tags:

FOSSASIA, SUSI.AI, Tutorial, GSoC, Skills, Development, API, Dashboard