* Info story or who is the user and why is it important to them

The user of this product is the recruiters looking for ideal employees in tech.然而，他们发现如果直接查看一个candidate的github profile，很难快速地得出analysis insight, 并快速地理解这个programmer的skill level。尤其是如果他们

正在寻找在某些programming area，比如擅长rust的programmer，将很难通过浏览这个prorgammer的repo，很快地总结出来。

而在有限的时间内找到合适的candidates对于recruiter来说非常重要，只有这样他们才能够deliver results for their company，并取得业绩上的成功和公司的更好发展。

因此，information that can help them 快速地分析一个candidate’s github account, providing details about their programming skills，并发现他们需要的ideal candidate，对于他们非常的重要，并且能够帮助他们achieve their goals and solve their problems,同时，这也是the purpose and motivation of the information.

因此，我希望design一个information product where recruiters可以enter the candiates’ github account name and 直接获得the account’s analysis, 帮助the recuiter 作出对于candidate的recruit决定

* Existing structure and FAIR assessment

Define the existing information structure:

通过make api calls to two GitHub endpoints for user inforamtion (https://api.github.com/users/EveWangUW) and user repo information (https://api.github.com/users/EveWangUW/repos), using the github account ‘EveWangUW’ as example, the existing information structure can be retrieved: (See the folder ‘exisiting information structure’ for the original user and user repo json files)

How is the existing information structured:

The user information contains details such as ‘login’,‘id’, ‘followers’, reflecting the general information about the user, 然而, 这个information structure也涵盖了很多不必要的信息，比如‘avatar\_url’等, that可能会让recruiter find it hard to understand the information, 感到informaion overload, 降低recruiter寻找优秀candidates的速度，不利于他们实现objective.

The user repo information存在着相似的问题. It contains details of each public repository of the user by presenting an array of different repositories. 虽然在每一个repo中都有比较有用的details比如‘language, created\_at’,‘updated\_at’, 但是recruiter只能一个一个repository地观察the candidates’ performance, 和直接在GitHub account page观察，差别并不大，缺少了对于the candidate programming skills的summary和更深的insights

User information structure:

{

"login": "EveWangUW",

"id": 142384748,

"node\_id": "U\_kgDOCHyebA",

"avatar\_url": "https://avatars.githubusercontent.com/u/142384748?v=4",

"gravatar\_id": "",

"url": "https://api.github.com/users/EveWangUW",

"html\_url": "https://github.com/EveWangUW",

"followers\_url": "https://api.github.com/users/EveWangUW/followers",

"following\_url": "https://api.github.com/users/EveWangUW/following{/other\_user}",

"gists\_url": "https://api.github.com/users/EveWangUW/gists{/gist\_id}",

"starred\_url": "https://api.github.com/users/EveWangUW/starred{/owner}{/repo}",

"subscriptions\_url": "https://api.github.com/users/EveWangUW/subscriptions",

"organizations\_url": "https://api.github.com/users/EveWangUW/orgs",

"repos\_url": "https://api.github.com/users/EveWangUW/repos",

"events\_url": "https://api.github.com/users/EveWangUW/events{/privacy}",

"received\_events\_url": "https://api.github.com/users/EveWangUW/received\_events",

"type": "User",

"site\_admin": false,

"name": "EveWang",

"company": null,

"blog": "",

"location": null,

"email": null,

"hireable": null,

"bio": null,

"twitter\_username": null,

"public\_repos": 25,

"public\_gists": 0,

"followers": 0,

"following": 2,

"created\_at": "2023-08-16T07:12:54Z",

"updated\_at": "2024-05-03T01:02:37Z"

}

user repo information structure:

[

{

"id": 793294266,

"node\_id": "R\_kgDOL0i1ug",

"name": "ActiveRecall-StudyBestFriend",

"full\_name": "EveWangUW/ActiveRecall-StudyBestFriend",

"private": false,

"owner": {

"login": "EveWangUW",

"id": 142384748,

"node\_id": "U\_kgDOCHyebA",

"avatar\_url": "https://avatars.githubusercontent.com/u/142384748?v=4",

"gravatar\_id": "",

"url": "https://api.github.com/users/EveWangUW",

"html\_url": "https://github.com/EveWangUW",

"followers\_url": "https://api.github.com/users/EveWangUW/followers",

"following\_url": "https://api.github.com/users/EveWangUW/following{/other\_user}",

"gists\_url": "https://api.github.com/users/EveWangUW/gists{/gist\_id}",

"starred\_url": "https://api.github.com/users/EveWangUW/starred{/owner}{/repo}",

"subscriptions\_url": "https://api.github.com/users/EveWangUW/subscriptions",

"organizations\_url": "https://api.github.com/users/EveWangUW/orgs",

"repos\_url": "https://api.github.com/users/EveWangUW/repos",

"events\_url": "https://api.github.com/users/EveWangUW/events{/privacy}",

"received\_events\_url": "https://api.github.com/users/EveWangUW/received\_events",

"type": "User",

"site\_admin": false

},

"html\_url": "https://github.com/EveWangUW/ActiveRecall-StudyBestFriend",

"description": null,

"fork": false,

"url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend",

"forks\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/forks",

"keys\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/keys{/key\_id}",

"collaborators\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/collaborators{/collaborator}",

"teams\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/teams",

"hooks\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/hooks",

"issue\_events\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/issues/events{/number}",

"events\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/events",

"assignees\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/assignees{/user}",

"branches\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/branches{/branch}",

"tags\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/tags",

"blobs\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/git/blobs{/sha}",

"git\_tags\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/git/tags{/sha}",

"git\_refs\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/git/refs{/sha}",

"trees\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/git/trees{/sha}",

"statuses\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/statuses/{sha}",

"languages\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/languages",

"stargazers\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/stargazers",

"contributors\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/contributors",

"subscribers\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/subscribers",

"subscription\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/subscription",

"commits\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/commits{/sha}",

"git\_commits\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/git/commits{/sha}",

"comments\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/comments{/number}",

"issue\_comment\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/issues/comments{/number}",

"contents\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/contents/{+path}",

"compare\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/compare/{base}...{head}",

"merges\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/merges",

"archive\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/{archive\_format}{/ref}",

"downloads\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/downloads",

"issues\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/issues{/number}",

"pulls\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/pulls{/number}",

"milestones\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/milestones{/number}",

"notifications\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/notifications{?since,all,participating}",

"labels\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/labels{/name}",

"releases\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/releases{/id}",

"deployments\_url": "https://api.github.com/repos/EveWangUW/ActiveRecall-StudyBestFriend/deployments",

"created\_at": "2024-04-28T23:51:14Z",

"updated\_at": "2024-04-28T23:52:01Z",

"pushed\_at": "2024-04-28T23:51:24Z",

"git\_url": "git://github.com/EveWangUW/ActiveRecall-StudyBestFriend.git",

"ssh\_url": "git@github.com:EveWangUW/ActiveRecall-StudyBestFriend.git",

"clone\_url": "https://github.com/EveWangUW/ActiveRecall-StudyBestFriend.git",

"svn\_url": "https://github.com/EveWangUW/ActiveRecall-StudyBestFriend",

"homepage": null,

"size": 358,

"stargazers\_count": 0,

"watchers\_count": 0,

"language": "JavaScript",

"has\_issues": true,

"has\_projects": true,

"has\_downloads": true,

"has\_wiki": true,

"has\_pages": false,

"has\_discussions": false,

"forks\_count": 0,

"mirror\_url": null,

"archived": false,

"disabled": false,

"open\_issues\_count": 0,

"license": null,

"allow\_forking": true,

"is\_template": false,

"web\_commit\_signoff\_required": false,

"topics": [],

"visibility": "public",

"forks": 0,

"open\_issues": 0,

"watchers": 0,

"default\_branch": "main"

},

{

"id": 777993033,

"node\_id": "R\_kgDOLl87SQ",

"name": "apigw-http-api-lambda-dynamodb-python-cdk",

"full\_name": "EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk",

"private": false,

"owner": {

"login": "EveWangUW",

"id": 142384748,

"node\_id": "U\_kgDOCHyebA",

"avatar\_url": "https://avatars.githubusercontent.com/u/142384748?v=4",

"gravatar\_id": "",

"url": "https://api.github.com/users/EveWangUW",

"html\_url": "https://github.com/EveWangUW",

"followers\_url": "https://api.github.com/users/EveWangUW/followers",

"following\_url": "https://api.github.com/users/EveWangUW/following{/other\_user}",

"gists\_url": "https://api.github.com/users/EveWangUW/gists{/gist\_id}",

"starred\_url": "https://api.github.com/users/EveWangUW/starred{/owner}{/repo}",

"subscriptions\_url": "https://api.github.com/users/EveWangUW/subscriptions",

"organizations\_url": "https://api.github.com/users/EveWangUW/orgs",

"repos\_url": "https://api.github.com/users/EveWangUW/repos",

"events\_url": "https://api.github.com/users/EveWangUW/events{/privacy}",

"received\_events\_url": "https://api.github.com/users/EveWangUW/received\_events",

"type": "User",

"site\_admin": false

},

"html\_url": "https://github.com/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk",

"description": null,

"fork": false,

"url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk",

"forks\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/forks",

"keys\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/keys{/key\_id}",

"collaborators\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/collaborators{/collaborator}",

"teams\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/teams",

"hooks\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/hooks",

"issue\_events\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/issues/events{/number}",

"events\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/events",

"assignees\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/assignees{/user}",

"branches\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/branches{/branch}",

"tags\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/tags",

"blobs\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/git/blobs{/sha}",

"git\_tags\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/git/tags{/sha}",

"git\_refs\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/git/refs{/sha}",

"trees\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/git/trees{/sha}",

"statuses\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/statuses/{sha}",

"languages\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/languages",

"stargazers\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/stargazers",

"contributors\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/contributors",

"subscribers\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/subscribers",

"subscription\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/subscription",

"commits\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/commits{/sha}",

"git\_commits\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/git/commits{/sha}",

"comments\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/comments{/number}",

"issue\_comment\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/issues/comments{/number}",

"contents\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/contents/{+path}",

"compare\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/compare/{base}...{head}",

"merges\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/merges",

"archive\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/{archive\_format}{/ref}",

"downloads\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/downloads",

"issues\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/issues{/number}",

"pulls\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/pulls{/number}",

"milestones\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/milestones{/number}",

"notifications\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/notifications{?since,all,participating}",

"labels\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/labels{/name}",

"releases\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/releases{/id}",

"deployments\_url": "https://api.github.com/repos/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk/deployments",

"created\_at": "2024-03-26T22:04:56Z",

"updated\_at": "2024-03-27T17:43:36Z",

"pushed\_at": "2024-03-27T17:42:46Z",

"git\_url": "git://github.com/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk.git",

"ssh\_url": "git@github.com:EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk.git",

"clone\_url": "https://github.com/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk.git",

"svn\_url": "https://github.com/EveWangUW/apigw-http-api-lambda-dynamodb-python-cdk",

"homepage": null,

"size": 35,

"stargazers\_count": 0,

"watchers\_count": 0,

"language": "Python",

"has\_issues": true,

"has\_projects": true,

"has\_downloads": true,

"has\_wiki": true,

"has\_pages": false,

"has\_discussions": false,

"forks\_count": 0,

"mirror\_url": null,

"archived": false,

"disabled": false,

"open\_issues\_count": 0,

"license": null,

"allow\_forking": true,

"is\_template": false,

"web\_commit\_signoff\_required": false,

"topics": [],

"visibility": "public",

"forks": 0,

"open\_issues": 0,

"watchers": 0,

"default\_branch": "main"

}

]

Assess existing information structure:

What is the FAIR principle

The FAIR principle is a set of guidelines to improve the findability, acccessibility, interoprability, and resuability of digital assets such as data in research and beyond.

The existing information strutcure在一些方面做得很好，比如，In my existing information structure，both the user data and the user repo dta have globally unique identifiers such as login, and id (F1) and the metadata clearly and explicitly include the identifier of the data they describe (F3), for example the two data both include urls to various resources such as ‘followers\_url’ and ‘commits\_url’.There is also rich metadata on the user and repository details, such as update timestamps, profile URLs, owner details (F2). The data are retrievable via standardized, oepn, and free APIs provided by GitHub (A1,A1.1). These data uses the JSON language format, which is a formal, accessubke, shared, and broadly applicable language for knowledge representation (I1). Moreover, the various urls included in the user and the repo data direct the users to related resources, for example the repo data includes the repo owner’s information (I3).

然而,if the user account or the repo is deleted, the metadata might not be accessible anymore,which may not conform to the A2 principle. In terms of resuability, although rich attributes are provided, 有一些attributes并不relevant to our recruiters, 比如avater\_url，is\_template等，没有实现R1。在user和user repo中，大部分的data都缺少了license information, 不conform to the R1.1. 最后，虽然time related attributes such as ‘updated\_at’,‘created\_at’提供了some provenance information，但是更多的details could be included to achieve the R1.2 principle.

To achieve data portability, it is also important to make sure the information structure conform to the three axes of data portability.

Clean meaning

首先，the information structure should have clean meaning,this means that attribute and the value of the data should be clear and the audience should be able to easily interpret the data. However，当前的information并没有为information users提供很clear的meaning. 当前的user info and user repo info只提供了raw metadata without interpretation, and the clean meaning is not expressed directly due to the lack of summary.所以recruiters只能自己manually提取信息to analyze来gain insights.

Clean meaning

需要给每个user和所有的repository提供一个summary, hilight key information such as most frequently used programming language, account active level等，帮助recruiter了解每一个data背后隐藏的inisghts

Accessibility

It should be accessible, meaning that we should have an easy way to access the information. 然而当前data is only available through manual GitHub API calls，或者直接访问这个github user的github page，并且recruiter需要重复地进行大量的主观interpretation来筛选需要的信息并generate insights，所以information并没有非常accessible.

Accessibility

可以develop一个accessible frontend web page where recruiters可以通过点击button直接search the github user and easily view the comprehensive analysis of their account，而不需要manually make api calls. 并且provide downloadable data so that recruiters可以存档并且做进一步的分析if needed

Transparency

The information should also be transparent, meaning that the process of how the data is collected, processed, and presented should be provided to the audience, ensuring that they understand the source and the transformation of the data. 然而当前的information缺乏transparency，因为它只是information in the json format, 而并没有解释这些github user的数据是从哪些地方收集到的，是怎么被处理和呈现出来的，the recruiters可能需要自己去understand这些信息，可能会导致一些理解误差和使用信息的困难。

Transparency

我们可以提供detailed documentation on what each metadata represent, how they are used during data analysis, and produced certain data metrics that should be understood in what way.

Interoprablity

It should also be interoperable, meaning that 这个信息应该可以很方便在不同的system中被使用，比如提供other data format such as XML and CSV.然而当前的information只有简单的JSON data format，并且是将user的json和repo的json分开的，并没有考虑到信息的整合，同时也没有考虑到对其他system的兼容性，recruiter可能需要额外的步骤将the JSON format data转换成他们想要的格式，加大了数据使用的难度。

Interoprablity

我们使用了typescript作为主要的programming language,对JSON data中的每一个attribute都有详细的data type and data schema documentation, 可以帮助the data to be used in other systems. 除此之外，our application used the React framework, which helps the webpage to be operable both on websites and on mobile devices. 未来我们也会考虑将data转换成CSV, XML的格式，来帮助提升interoprability.

Usability

It should also be usable， meaning that the information presented to the user should be immediately useful for the audience. 然而当前的数据只是raw metadata of the github user，缺少对于user的programming skill的deeper insights. 因此recruiter将不得不花大量的时间去分析这些数据，来做到对candidates的技能评估，降低了这个information的usability.

Usability

我们used the raw metadata and conducted analysis to generate deeper insights such as the most commmonly used programming language and created a summary of the user and their repos, 帮助recruiter quickly and systematically retrieve useful insights from the raw data

Structure

最后, the information should use an appropraite structure. In our case, the format and sturtucre of our data is in JSON, with a nested structure for user and repository information.

然而，当前的structure是两个分散的json file，并没有将user info和user repository info很好的combine在一起，以更加集中，更加summarizing的structure反应这个user的programming skills，因此在structure上也应该有所提高。

Structure

因此我们可以combine the user and repository information into a single information structure that includes two nested about user info and repo info analysis, 从而帮助我们更加structured access the analysis insights

How you decided to improve the structure

因此，根据我的dataset根据the fair principle和the three axes of data portability 所identify出来的deficiencies, 我decide to improve the existing information structure accordingly so that the information can better conform to FAIR and enhance its portability to achieve the ideal use case for the recruiters.

In terms of the FAIR principles,

therefore，to improve accessibility (A),

1. implement a backup system to retain metadata even if the user account or repo is deleted, which can be accomplished by allowing the user to download the data from the web application

To improve reusability (R),

1. Remove unnecessary attributes like avator\_url and focus on displaying information related to recruiters.

2. Add relevant license for this application

3. Add more analytics details and provenance information to provide recruiters with useful insights into candidate profiles.

In terms of data portability,

What your new structure is

这些信息将通过一个web application呈现，从app的operation journey来看，the recruiter can enter the github user’s name in the frontend page powered by react and typescript. The product will make an API call to the GitHub endpoits to retrieve the user’s account and repository information.

一方面，这些information会被displayed on the frontend in the original json format. 并且有一个download button for the recuiter to download the json file.

除此之外，after the json data is retrieved from the api, the product会对信息进行筛选，并进行合理的manipulation, 包括 The time of the user on github, the user value based on programming skills, and the active level of the user on github.

最终将这些information传递到frontend，为recuiter提供only和recruitment相关的user programming skill information, 帮助recruiters evaluate the skills of the candidate and make the recruitment decision.

首先，in terms of the information itself, 相比the existing information, 进行了信息的删减，保留了对于recruiter analyze candidate’s programming skill level有用的信息。除此之外，添加了new fields such as the time of the user on GitHub, which analyzed and summarized information from the user account and all the repositories’ information.

For the structure and format, the existing information structure is using raw JSON data directly retrievd from the GitHub API and includes overally details fields that may not be relevant to the recruiters, such as node Ids or avatar urls. As an improvement, the new structure uses filtered and summarized data that includes only relevant fields such as user name, repository name, and additional computed fields such as total time the user has been on GitHub and their programming skill levels, providing more relevant information to the recruiters.

In terms of access methodology, 过去the recruiter只能通过直接访问the target candidates的github page来了解这个user的information。然而，现在通过the access technology of a REST API call,the recruiter能够快速地get comprehensive information on the GitHub user and their repos. 这个user also adopts the GraphQL API access methodology to access更加tailored information, 比如某个github user的account name，它的前五个repo的creation time等，through在前端填入所需的信息。

* How would quality be identified and addressed

为了保证the project is durable and robust, 我们会在api call的代码中加入unit testing，保证99%+的test coverage, 保证从user interact with frontend to send request, receive response到manipulate data 到最后的frontend信息呈现,都能durable和robust的实现. 我们可以apply the testing framework of typescript 来realize这个

除此之外，为了保证过去的信息不流失，我们的app有download 当前analysis original data的功能。Recruiter可以直接click on the download button on the frontend to make sure the data不会lost

另外，为了保证api call和整个product operate的安全性，这个project is also deployed to the amazon s3 bucket. The recruiters不需要有任何的domain knowledge on software development 就可以easily open the application via the web url provided by s3 and 进行user profile的analysis.

同时我们使用了the MIT license for our code to ensure open-source accessibility.