Final Report

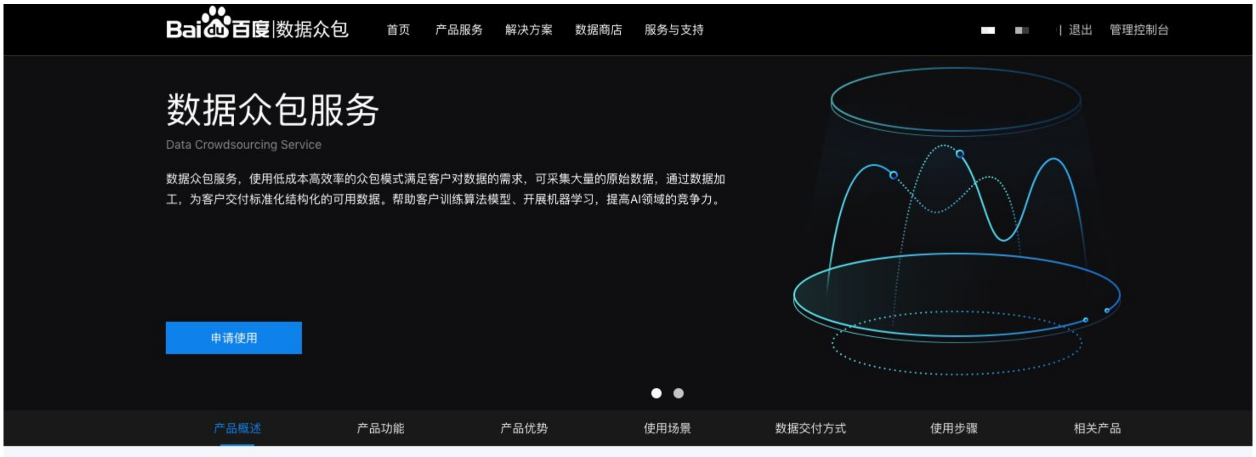
Data Labeling Platform by Yimian Corp.

Jiang Tiankai	11510109	Xu Chenhao	11510102
Hu Yuxing	11510225	Xie Danning	11510718
Yan Xiangyi	11510706		

15 June, 2018

Motivation

The artificial intelligence technology is more and more popular and useful today. Some brilliant company such as Yimian Data, Tencent, etc. have applied AI in the commercial field to estimate their services performances and optimate their product. Therefore, a big quantity of labeled data set which helps the machine learns well is essential. However, there is not a very powerful free data label platform that is oriented to every normal person on the internet.



数据众包服务

Data Crowdsourcing Service

数据众包服务，使用低成本高效率的众包模式满足客户对数据的需求，可采集大量的原始数据，通过数据加工，为客户交付标准化结构化的可用数据。帮助客户训练算法模型、开展机器学习，提高AI领域的竞争力。

Baidu Zhongbao. Charged and not oriented to everyone.

Therefore, our SkyCat, a new generation data label platform is designed for that everyone can use it to help technology company label the data and gain some credits which can be exchanged with some merchandise correspondingly. As a consequence, our platform must have the ability to let everyone use it. It must have the restricted function such as upload data set via .zip files, support at least 100 connect at a time, user and credits system, fault tolerance mechanism, etc. And now, we implement all the function above which indicates the platform is ready for being used.

Features And Requirements

We held meeting with stakeholders twice and agreed on the features and requirements of the platform at last.

Yimian Corporation is a data analyzing company for other business companies such as P&G, BMW. And they have a series of data analyzing tools, from data preprocessing to nature language processing, which means that our product should fit into the whole series. And most of the feature and requirements are established based on the function of other products.

There are 9 requirements in total.

1. **Flexible labeling UI that satisfies various labeling requirements.**

To some extent, Yimian corporation is a middle man between companies like P&G and users. Companies gather information about their products such as comments from Taobao or Tmall and send them to Yimian Corp. All comments will be preprocessed by computer and some of them will be sent to this platform to get a label. And further these labels will be sent back to companies and become data to analyse their products. Because of that, the labeling platform should be flexible. Because each company that sends data here has a unique format. The platform should be easily modified to satisfy different format from different companies.

2. **Compatibility for both PC and mobile platform.**

Unlike Amazon turk or Baidu Zhongbao, for which the labeling people label as a full time job, this platform is aimed for everyone. Everyone can sign up for a account and label some data and get some credits. Therefore, not only efficiency should be considered. Nowadays, most people use smartphone more often than PC. The platform should be compatible for both mobile and PC so that everyone can take part in it.

3. **100 people labeling simultaneously.**

Yimian corporation is not a big company, but still the efficiency is an important index. The platform should support at least 100 people to label at the same time. It is not about upgrading the configuration of the server, it is about properly using the SQL language and backend optimization.

4. **Fault tolerance algorithm to ensure the accuracy of the result.**

None of other similar products like Amazon turk and Baidu Zhongbao has fault tolerance philosophy behind the scene. Because their worker works as a full time job and the accuracy can be ensured somehow. However, our platform is aim for everyone, people with different educational backgrounds. A fault tolerance algorithm has to be set up to prevent malicious usage. The algorithm should be sensitive enough to detect wrong answers and efficient enough to let a question not appear too much times.

5. **Integration with third party platform, so that the labeling result can be used as Recaptcha or something else.**

Google uses pictures taken by autonomous vehicles as Recaptcha to train and verify their algorithm. Similarly, our platform should provide an interface for third-party platforms. They can use labeled sentences as distinguishment between human and computer, and the result also can be used for scientific researches.

6. **Administrator background for publishing tasks.**

This is a basic requirement. The platform is not only built for turkers, it is also for companies who wants the label information. Therefore, we must provide an approach to let them upload the questions. Considering that usually the question bank is extremely large, we should set a rule of files to be upload. The server will dump all files with wrong format. Apart from uploading, the uploader should be able to download his/her files anytime, even when the question set is not fully labeled.

7. **User management. There are three kinds of identities in this system. The ordinary users to label the tasks, the administrator to publish tasks and the super administrator to view all tasks and users.**

We set the rules for them. Ordinary users can only label. There is a web page summarizing their achievements and another for them to label on. Uploader can only publish tasks, there is a web page for them to download tasks they published previously and another page to publish new tasks. Super administrator can publish tasks also, but their major role is to manage all users, so they have a page to view all users' information, such as email and register time. We also should build another page for them to view and manage all tasks.

8. **Support tasks with different priorities, the one with higher priority will be labeled earlier.**

There are three priorities in our platform, namely, high, mid and low. We can arrange tasks by their priority so that tasks with higher priority can be labeled earlier.

9. **Credits management based on user accuracy.**

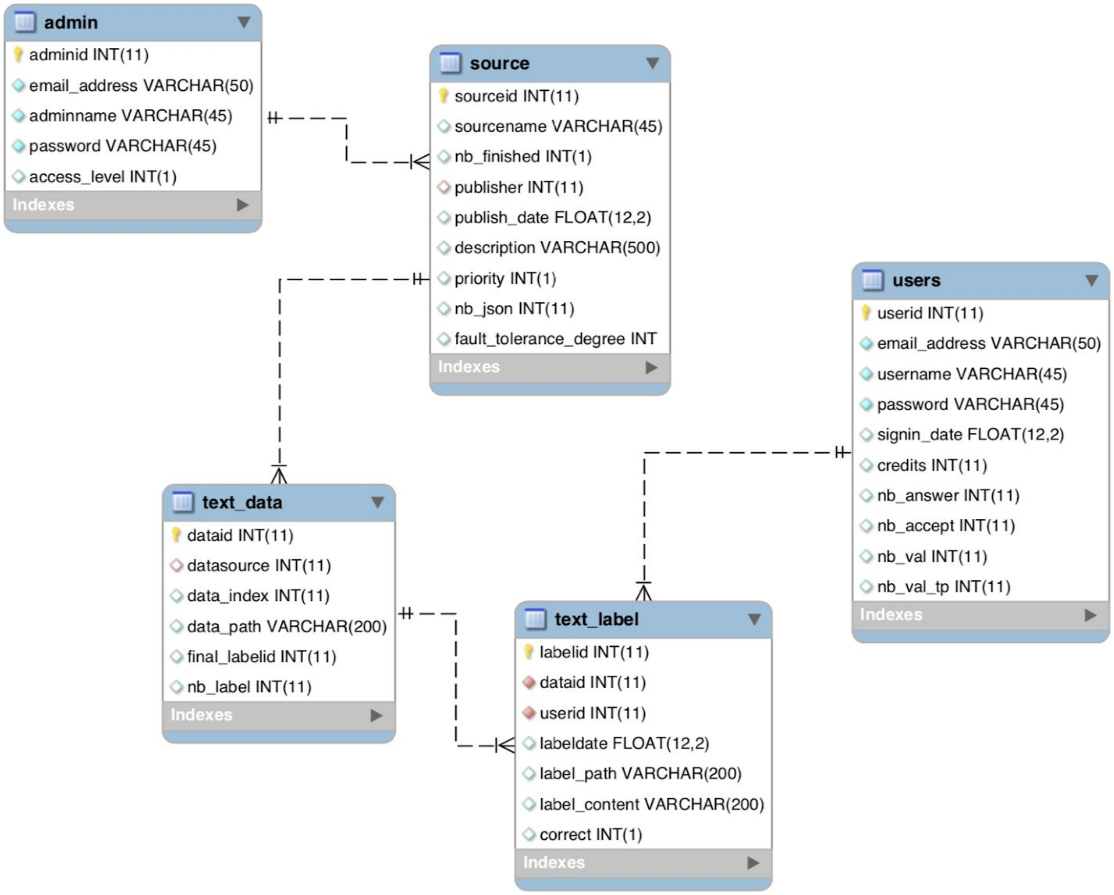
Because this platform is built for everybody, we have to provide awarding mechanism, or no one will work for it. We use credits as a parameter in fault tolerance algorithm, but here credits is a different term, user can use their credits, aka, points, to exchange prizes, or money. We left an API here for Yimian corp. to decide the usage in the future.

Above are all features and requirements brought up by stakeholder in Yimian Corp. At the end of this project, all of these requirements are supported by our platform.

Design And Implementation

Database Design:

The ER diagram of the database we designed is shown in the following figure. We designed table *users, admin, source, text_data, text_label.* Each of the table has some attributes, some of which we will explain further.



1. **users**

   nb_answer : Total number of questions answered by user

   nb_accept : Total number of user's accepted answer

2. admin

access_level : 1 normal admin; 2 super admin

3. source

The table is designed to store all the labeling tasks/projects.

publisher : foreign key, reference to adminid in table admin

priority : 1 low (default); 2 normal; 3 high

nb_json : Total number of .json file uploaded, i.e., total number of questions

nb_finished : Total number of questions that have accepted answers.

(The progress rate is calculated by nb_finished/nb_json)

fault_tolerance_degree : 0 turn off (default); 1 low degree; 2 high degree

4. text_data

The table is used to store all data (questions to be answered) from each source

datasource : Foreign key, reference to sourceid in table source

data_index : For each source, every data should have a unique index . *data_index* is the index in the source the data belongs to.

dataid is the primary key. The combination of *datasource* and *data_index* should also be unique

data_path : The absolute path of the .json data file.

final_labelid : Foreign key, reference to labelid in table *text_label*. If the question has an accepted answer, the *final_labelid* should point to one of the accepted answer's *labelid* in table *text_label.* Else *final_labelid* = NULL as it is initialized.

nb_label : abandoned.

5. **text_label**

dataid : Foreign key, reference to *dataid* in table text_data. The corresponding data of this label.

userid : Foreign key, reference to *userid* in table **users**. Data labeler.

label_path : The absolute path of the '.json' label file.

label_content : Part extracted from the '.json' file. We do fault-tolerance process on only this part.

correct : Whether the answer is accepted. 0 incorrect/not determined yet (default); 1 correct.

Task Publish API Design:

The only thing a task publisher needs to do is to write a script to fit their data in our format.

1. File structure

Here's an example file structure of **upload.zip**. After extract your **.zip** file, we should get and only store the following files.

```
upload                 # The folder
 ├ meta.json            # Project describer
 ├ 1.json               # Unlabeled data
 ├ 2.json
 ├ 3.json
 ├ 4.json
 ├ 5.json
 ├ 6.json
 ├ 7.json
 ├ 8.json
 ├ 9.json
 ├ 10.json
 └ 11.json
```

2. Project describer: meta.json

Firstly, you need your **meta.json** to describe your data set.

```
{
    "projectName": "test",
    "description": "This is a test for se2018.",
    "fault_level": 2
}
```

You can choose 3 values for fault tolerance **fault_level**, which are **0** for turn off, **1** for low level and **2** for high level.

3. Unlabeled Data: 1.json

```
{
  "projectName": "test",
  "index": 723154,
  "data": "物流速度快，性价比高",
  "task": [
    {
      "mode": "single",
      "front": "option",
      "aim": "请选择这句话的语言：",
      "label": "中文",
      "choices": ["中文", "英文"]
    },
    {
      "mode": "multiple",
      "front": "box",
      "aim": "请选择这句话的情感：",
      "label": ["开心", "满意"],
      "choices": ["开心", "满意", "失望", "生气"]
    },
    {
      "mode": "open",
      "front": "blank",
      "aim": "请选出句子中的形容词：",
      "label": ["快", "高"]
    }
  ]
}
```

In **1.json**, you need to announce your **projectName**, **index**, **data** and **task**.

- **projectName** is the name of your project.

- **index** is the index of your data. (We recommend to name the json file as index.)

- **data** is the unlabeled text data.

- **task** includes many subtasks.

In one **task**, you need to announce the **mode**, **front**, **aim**, **label** and **choices** of your subtask.

- **mode** includes **single** for single choice, **multiple** for multiple choice and **open** for whatever you want the users to type in.
- **front** includes **option** for single choice cycles, **box** for multiple choice boxes and **blank** for the boxes whatever you want the users to type in.
- **aim** is the question you want to ask.
- **label** is the answer of **aim**. It can be blank or something you already have and want users to modify if it's not correct.

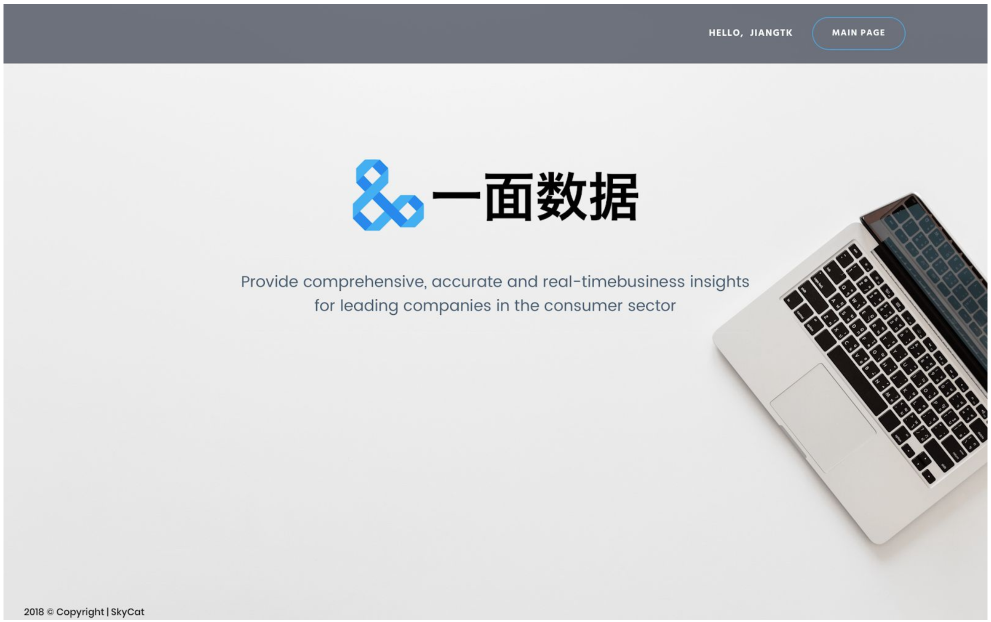
User Interface

There are several part of user interface. As the company required, we are suppose to design three different user interface for different user to use our program. So in this document, the order of this sector will from low level entrance to high level entrance.

Our website has already deployed on our Alibaba Cloud server, and has a public network IP to access it. The IP address is http://47.106.34.103:5000, and we recommend you to access it by yourself whether than reading our document, and all of our source code is open sourced on Github as well.

There will be three types of user status provided on our website, regular user, regular admin and super admin. Regular user can only check their labeling datas, their status and to label. The regular admin has the permission to upload tasks and to publish them. The super admin has the most powerful range of permission including publishing task, creating regular admin and checking the detail of every single user.

The first page of our website is welcome and info page. And this page display as below:

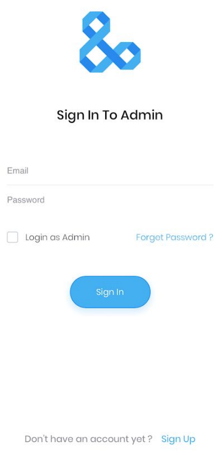


Provide comprehensive, accurate and real-timebusiness insights for leading companies in the consumer sector

2018 © Copyright | SkyCat

The welcome page not only including the login and register button, if scrolling below, you may also found some feature about our website and some other stuffies. Such as expert comments, page preview and the way to contact us.

So the next page we want to introduce is login and register page. Need to mention if the tourists stays here without login, he may not access any of our website. At login and register page, user can register as regular user or login as user, regular admin or super admin, just by clicking the button we “Login as Admin”.



Sign In To Admin

Email

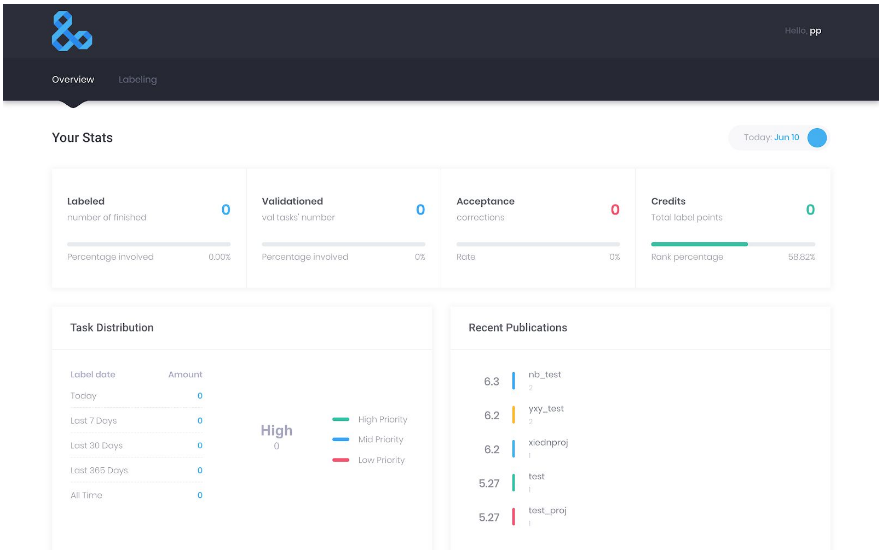
Password

□ Login as Admin   Forgot Password ?

Sign In

Don't have an account yet ? Sign Up

As regular user, next page he shall see is the personal center. There’s some data at the top with the percentage bar with it like credit and accuracy. In the left corner is the time analysis and priority distribution. And the right corner is recent published task.



Hello, pp

Overview   Labeling

Your Stats

Today: Jun 10

Labeled	Validationed	Acceptance	Credits
number of finished	val tasks' number	corrections	Total label points
0	0	0	0
Percentage involved 0.00%	Percentage involved 0%	Rate 0%	Rank percentage 58.82%

Task Distribution

Label date	Amount
Today	0
Last 7 Days	0
Last 30 Days	0
Last 365 Days	0
All Time	0

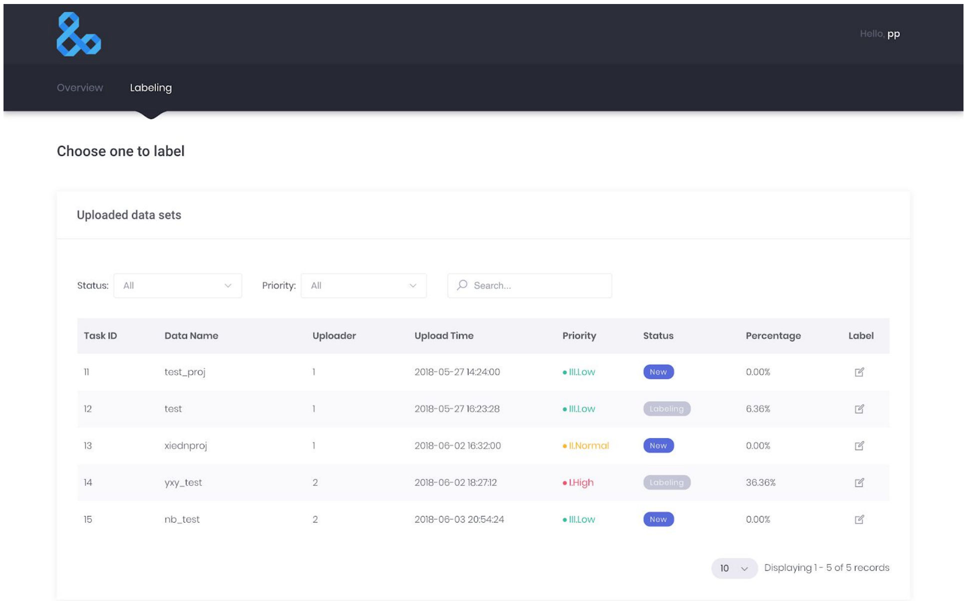
High
0

🟩 High Priority
🟦 Mid Priority
🟥 Low Priority

Recent Publications

6.3	nb_test	2
6.2	yxy_test	2
6.2	xiednproj	1
5.27	test	1
5.27	test_proj	1

The top navigation bar provided another option to click which is the labeling page. In the labeling page, user can choose one task to label, each task has lots of information displayed on this page including uploader, status and priority. And all these data is changing dynamically. Click the label icon, will lead to labeling page.



Choose one to label

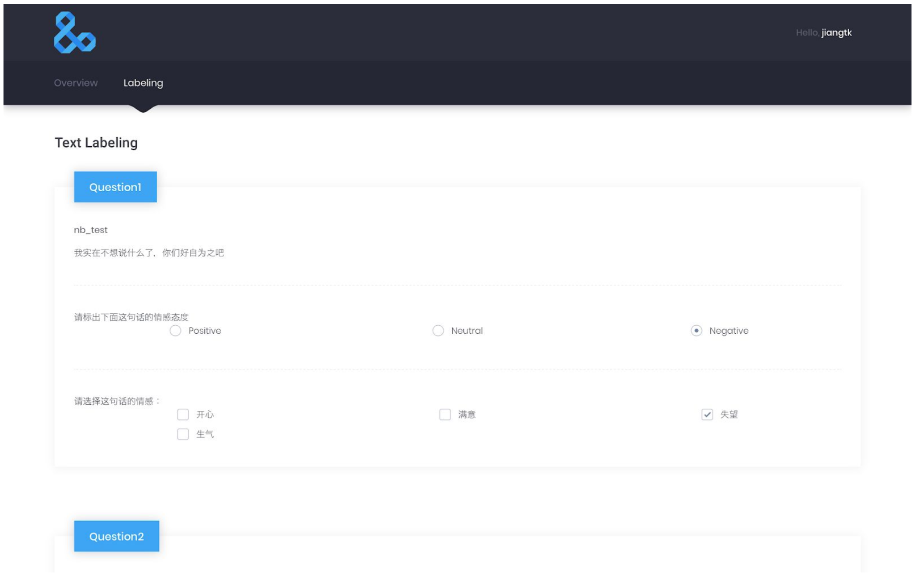
Uploaded data sets

Status: All
Priority: All
Search...

Task ID	Data Name	Uploader	Upload Time	Priority	Status	Percentage	Label
11	test_proj	1	2018-05-27 14:24:00	● III.Low	New	0.00%	✎
12	test	1	2018-05-27 16:23:28	● III.Low	Labeling	6.36%	✎
13	xiednproj	1	2018-06-02 16:32:00	● II.Normal	New	0.00%	✎
14	yxy_test	2	2018-06-02 18:27:12	● I.High	Labeling	36.36%	✎
15	nb_test	2	2018-06-03 20:54:24	● III.Low	New	0.00%	✎

10 Displaying 1 - 5 of 5 records

So here is the labeling page. In this page, we can see that there are several question stated here. Each question is directed import from uploader's json file and all the choices can be smart introduced into this page. On the button of this page, we provide two options: you can click submit to upload your answer and quit this task or click five more to both save your answer and also get five more answer to continue your labeling journey.



Text Labeling

Question1

nb_test

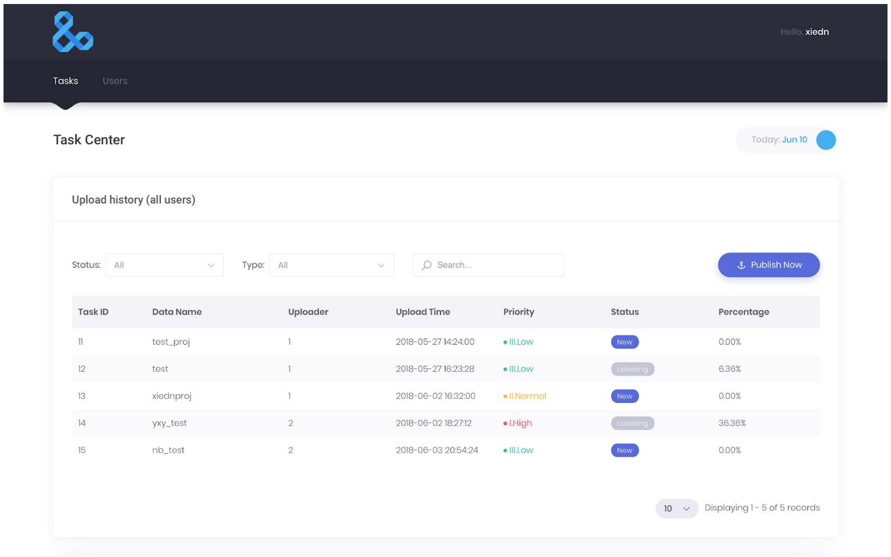
我实在不想说什么了，你们好自为之吧

请标出下面这句话的情感态度
○ Positive ○ Neutral ● Negative

请选择这句话的情感：
□ 开心 □ 满意 ☑ 失望
□ 生气

Question2

After the user's view, it's time to check how regular admin will receive the interface. Regular admin can see all the information of tasks that he uploaded and can download those data whenever he want. Or click the publish button to publish a new task. This webpage is highly similar with the view that super admin will see, except super admin will see all the task no matter who upload it. Just like the screenshot below.



Task Center

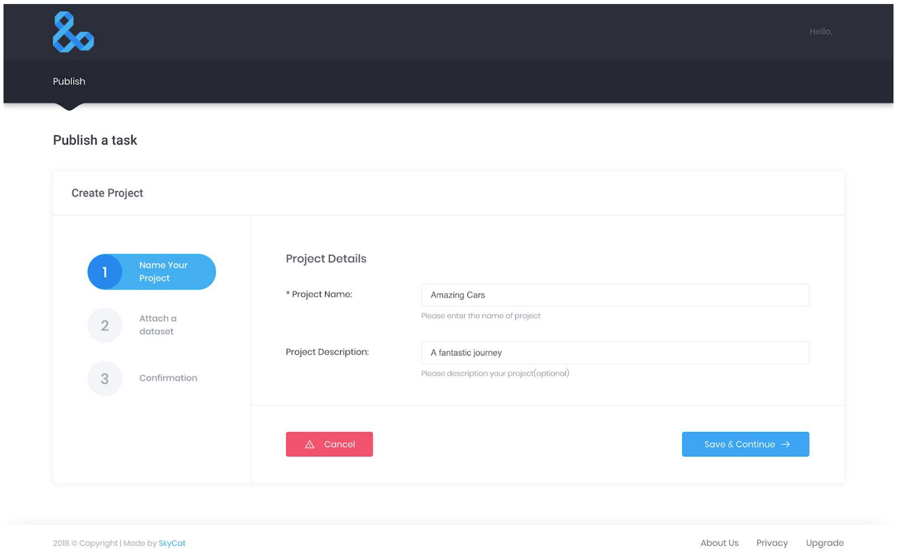
Upload history (all users)

Status: All Type: All

Task ID	Data Name	Uploader	Upload Time	Priority	Status	Percentage
11	test_proj	1	2018-05-27 14:24:00	● III.Low	New	0.00%
12	test	1	2018-05-27 16:23:28	● III.Low	Labeling	6.36%
13	xiednproj	1	2018-06-02 16:32:00	● II.Normal	New	0.00%
14	yxy_test	2	2018-06-02 18:27:12	● I.High	Labeling	36.36%
15	nb_test	2	2018-06-03 20:54:24	● III.Low	New	0.00%

10 Displaying 1 - 5 of 5 records

Click publish, and we are in the publish page. This page require admin to type in some basic info about the task and also the database of this task. The format of database should be a zip package, and we recommend our upload to follow the rule we provide to set the upload file correctly.



Publish a task

Create Project

Project Details

* Project Name: Amazing Cars

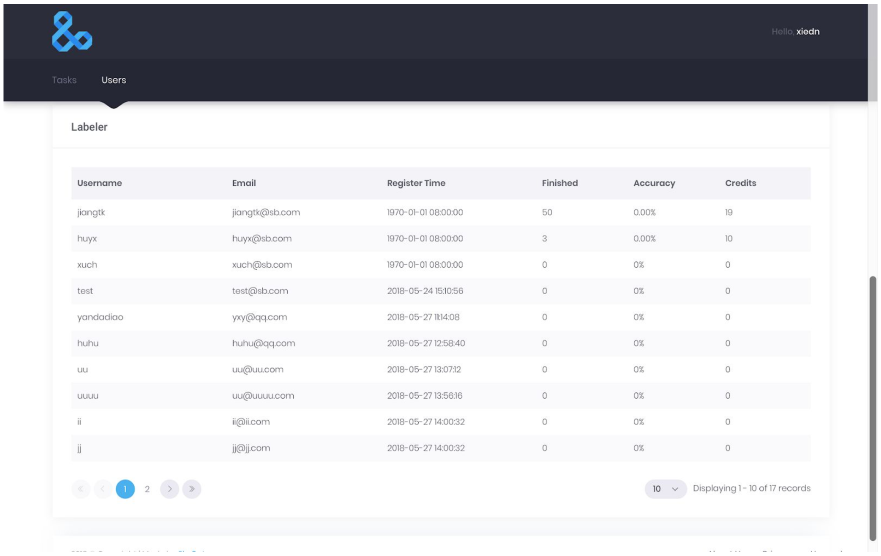
Please enter the name of project

Project Description: A fantastic journey

Please description your project(optional)

2018 © Copyright | Made by SkyCat

And for super admin, he is able to check the status of all the users and regular admins on the user page. He can also create some new admin so the process of upload task could be more efficient.

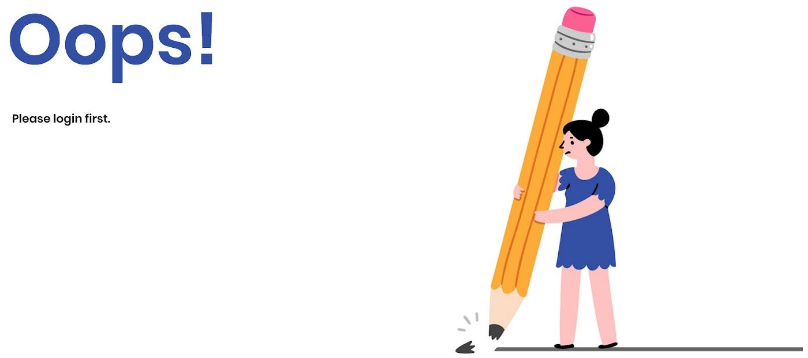


Labeler

Username	Email	Register Time	Finished	Accuracy	Credits
jiangtk	jiangtk@sb.com	1970-01-01 08:00:00	50	0.00%	19
huyx	huyx@sb.com	1970-01-01 08:00:00	3	0.00%	10
xuch	xuch@sb.com	1970-01-01 08:00:00	0	0%	0
test	test@sb.com	2018-05-24 15:10:56	0	0%	0
yandadiao	yxy@qq.com	2018-05-27 11:14:08	0	0%	0
huhu	huhu@qq.com	2018-05-27 12:58:40	0	0%	0
uu	uu@uu.com	2018-05-27 13:07:12	0	0%	0
uuuu	uu@uuuu.com	2018-05-27 13:56:16	0	0%	0
ii	ii@ii.com	2018-05-27 14:00:32	0	0%	0
jj	jj@jj.com	2018-05-27 14:00:32	0	0%	0

Displaying 1 - 10 of 17 records

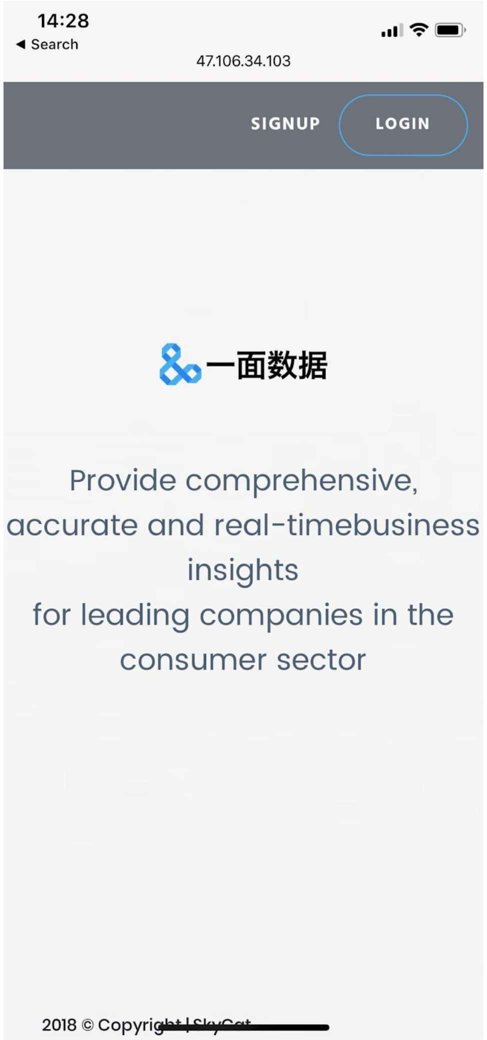
If anything goes wrong, we create some fancy 404 page to display for our user and the words on it give the direct command of what to do.



Oops!

Please login first.

Last but not least, we noticed that for things like labeling could fully use the fragile time of a person just on his phone. So we create the version for phone as well for our user to label anywhere or anytime he like.



Provide comprehensive,
accurate and real-timebusiness
insights
for leading companies in the
consumer sector

2018 © Copyright | SkyCat

Conclusion

Above is the introduction to the labeling platform, the feature, the usage and the API it provides.

We have tested this platform under multiple situations, including stress testing, and the platform handles all well, which means that it is a industrial level product, a product that can actually be public to the whole world.

We hope the technologies and techniques, or simply code, of this platform can inspire more people. And everyone is welcomed to bring up push requests or issues on the github of this project (https://github.com/lifesaver0129/Sky-Cat-Labeling).