

# Iteration 0 report

## 1. Team roles

Scrum master: Wenrui Zhang

Product owner: Xiaopei Xu

## 2. Customer meeting date/time/place

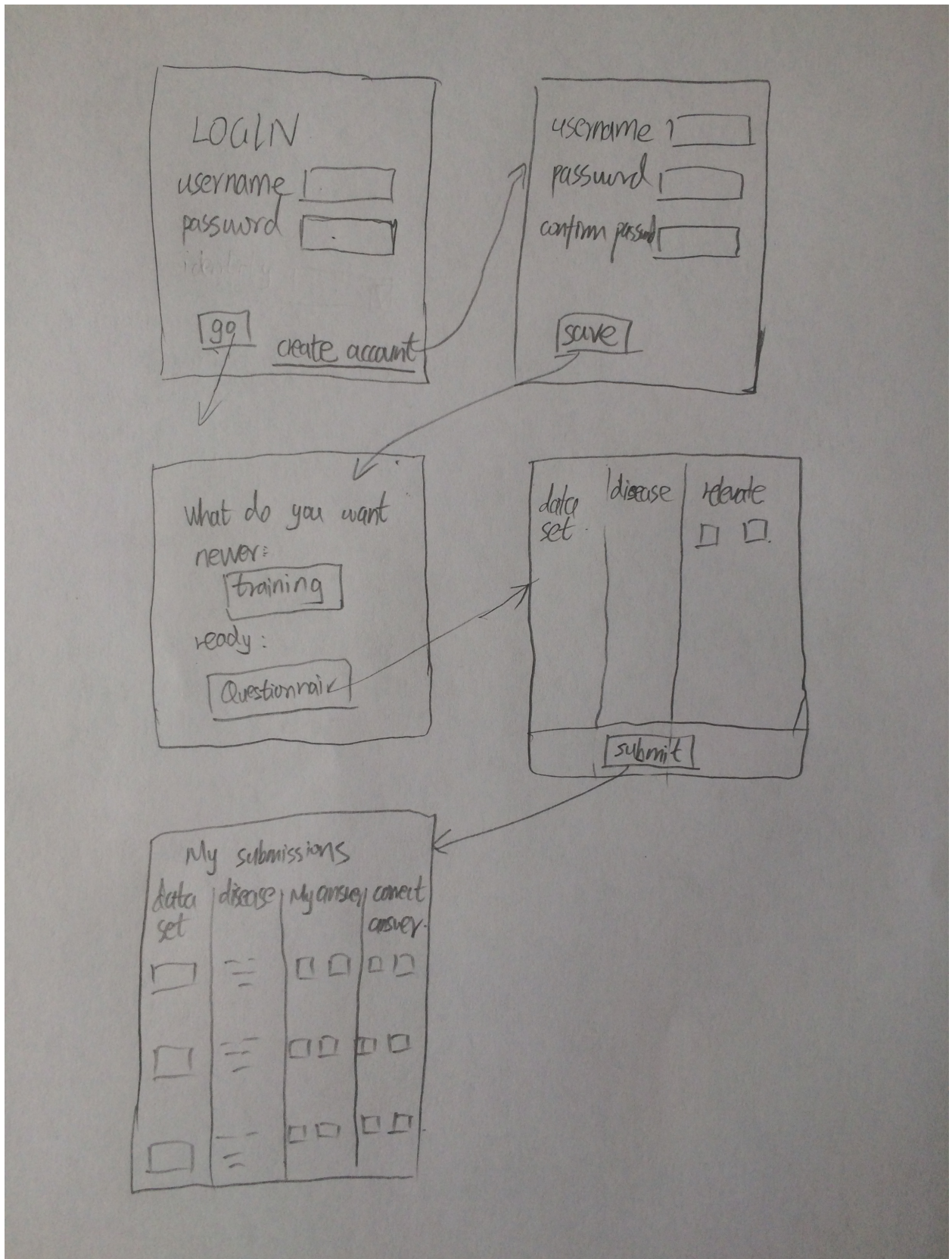
Meeting on Friday 4:30 pm at CBGSE (101 Gateway, Suite A, College Station).

## 3. Summary

The goal of this project is to design a website which serves as a basic framework for crowdsourcing of biodata curation. Specifically, this website should have 3 main features (more detailed requirements and feedbacks will be provided as the project evolves). First, an interested individual should be able to register on the website as a data curator. They can either set up a new account or login in with their social account like Facebook or Google+. Secondly, tutorial for this websites should be rendered to the curators in a user-friendly fashion (It should be a dynamic instruction. It's better to instruct user in a game way) so that even people without much biological background will be able to easily learn the necessary curation skills. Third and the most important, the website should record the curation results of these registered data curators which will be accessed both automatically (by comparing the results among curators using, for example, weighted majority vote) and manually (for the cases that are hard to be solved automatically). There should be a way to evaluate curators based on their past performance. Thus, the results from curators who have good curation history should be trusted more in deriving the correct results.

In order to emphasis the pain point of this project, in the follow, I'll show the main stream of this websites. After logging in, curators are given several datasets which are assigned by administrator with a few questions set by administrator. These datasets are extracted from Array Express or GEO automatically with the term for searching provided by administrator. Curator can answer the questions by simply choosing yes or no. The website judges the correction of these questions by either taking the most frequent choice of curators' answers or using the answers given by administrator. After getting correct answers, the website can automatically define the performance of each curators. For administrator, he can search term for related datasets. This terms are defined by the answers of questions. Also administrator can choose to only display the answers of curators who have high performance. Other features like improving database will be considered if time is possible.

#### 4. Lo-Fi UI



## 5. Legacy

Project Name: Genetic Dataset Curation Website

**Platform:** Ruby on Rails

**Tools:** Github Heroku RSpec&FactoryGirl

**Main Goal:** Provide an online website platform for users to submit their answers to the questionnaires about the relationships between disease name and gene NO. And support administration for administrators on the results.

**Function Description:**

The root-URI page is the log-in page. It offers options for creating new user, which will lead to a new user setting page. The top layer of the website offers some useful panels for quick access. They are Home, Help and Log in tags. The bottom layer of the website offers some development information, which are links to the information about the website developers.

After correctly login or submitting new user request, the website will be directed to user profile page. And there are two situations. For users, they will only see the results of the questions they've already submitted their answers, namely the "submissions". A button "Start" is offered to the users to direct to the disease page, where they can make their choices on relevance and reason and submit the answers. For administrators, they can not only do what the users could do, but look at the users' statistics, answers distribution as well. Besides, under every circumstance the DataSet value appears, there will be a hyperlink to an external website contains the information about this DataSet.

**Learning Codes:**

**Scheme of the tables stored in database:** 3 tables

Users: name email created\_at updated\_at password\_digest remember\_digest admin accuracy

Disease: disease accession created\_at updated\_at related unrelated closed

Submissions: disease\_id user\_id is\_related created\_at updated\_at reason

**Top View of the Website Design:** 3 models(User, Disease, Submission).

**Details:** Use sessions\_controller to control the root-URI (the login page) and check and store the user login information in session[], and finally direct to the User model.

The User model has two controllers-the admins\_controller and the users\_controller. These controllers take care of the Routes concerned with the users. There are ways in this model to connect the other two models.

The Disease model has only one diseases\_controller. This controller's work is mainly getting data from the database and support a user interface for users to submit answers.

The Submission model doesn't have code in the submissions\_controller. This model belongs to the disease and user. Inside this model, the work is mainly interacting the submission results of a specific disease and user with the database records.

**Previous stories improvement:**

Make the website much easier to maintain.

## 6. Pivotal Tracker

<https://www.pivotaltracker.com/n/projects/1887551>

## 7. GitHub

<https://github.com/xvxiaopei/CSCE606-project>

## 8. Video

<https://youtu.be/YVqrbsdxi5M>