

Teamwork homework2

Assignment T2: Revised Project Proposal

Team name: Initial Dream

Member names: Yifan Li (yl4287), Shuwan Yao(sy2884), Zixian Zhu(zz2689), Yuqiu Gan(yg2684)

Part 0:

The date of your meeting with your IA mentor

Oct. 27th

Nov. 5th

Thank you Rennah and Minxuan! You did great help for our team!

Part 1:

Write a few paragraphs that provide an overview of the software engineering project that your team would like to do and answers all five of the numbered questions.

1. What will your project do?

Our project aims to develop a Healthy Code System for the Columbia University Community(Staff & Students).

We collect users' data including identity information and travel history and evaluate the risk level of being affected by Covid-19 of users. Then we will show a different Healthy Code to each user according to its risk level.

“Yellow Healthy Code”


For example, if a user traveled to a high risky state in the last 14 days without symptoms, we'll assign him/her “Yellow Healthy Code” to indicate he/she has the risk of being affected by Covid-19.

User can clear his/her “Yellow Healthy Code” and gain a “Green Healthy Code” in two ways:

The first way, quarantining himself/herself for 14 days after returning from a highly risky area without any symptoms. He/she may submit on the interface that he/she has quarantined for 14 days. The system will verify users' GPS signal to see if they stay quarantined. Then the system will check if it has been 14 days before his/her last yellow healthy code record. If it is, a green will show up for the user.

The second way, getting a Covid-19 test as soon as possible. He/she may upload his/her test result (pdf). Optical character recognition(OCR) api is used to valid personal

information and the test result of the user. If the result is negative, the yellow healthy code will turn to a Green Healthy Code. If the result is positive, the yellow healthy code will turn to a red one.

		CLINICAL RESEARCH SEQUENCING PLATFORM		CRSP 320 Charles St. CAMBRIDGE MA 02141 617-714-7000		COLUMBIA Columbia University 1116 Amsterdam Ave NEW YORK, NY 10027	
PATIENT NAME Yao, Shuwan		PATIENT ID sy2884	LAB MRN C11246	PHONE 9179697361	DOB 06/17/1997	AGE 23 Yrs	SEX F
REQUISITION # HCWSQMTWHPZJTYR	COLLECTED DATE 11/4/2020 3:46 PM	RECEIVED DATE 11/5/2020 3:10 PM	ORDERING MD Bernitz, Melanie				
Test Description	Results	Abnormal	Reference Range	Units	Lab		
Covid19_Diagnostic		Source: AN SWAB		(Status: F 11/05/2020 15:10)			
SARS-CoV2 Real-time Reverse Transcriptase (RT)-PCR Diagnostic Assay		NEGATIVE		Negative		CRSP	
<p>2019-novel Coronavirus (2019-nCoV) not detected by the qRT-PCR assay. Consider testing for other respiratory viruses or re-collecting for 2019-nCoV testing. Note: Optimum timing for peak viral levels during infections caused by 2019-nCoV have not been determined. Collection of multiple specimens from the same patient may be necessary to detect the virus.</p> <p>Methods and Limitations: This Laboratory Developed Test is a high-throughput version of the CDC 2019-nCoV Realtime RT-PCR test and has been validated in accordance with</p>							

If a lab result can not be recognized correctly, the user can submit a manual validation request to superuser.

“Red Healthy Code”

For users who are already affected by Covid-19(Covid-19 test positive or having symptoms), he/she will be assigned the “Red Healthy Code” directly.

If the user is cured from covid-19 and wants to get a green pass again, he/she may upload a negative report. And if the report passes OCR api’s validation, he/she can get a green pass.

“Green Healthy Code”

For users who never traveled to any high risky areas in the recent 14 days and don’t have any symptoms or users who got recent Covid-19 negative reports or users who had “Yellow Healthy Code” and satisfied conditions of clearing risk, they will be assigned “Green Healthy Code”.

Use of “Healthy Code”

“Healthy Code” is a certification of a user. Only users who hold the “Green Healthy Code” can access the Columbia University campus and buildings, labs.

“Personal Center”

In this website, each user can also use columbia email and password to login in the personal center, where they can check their health reporting record, get the status of Health Code(green, yellow, red), and fill in a new health record.

“Pre-training”

A pre-training is needed for each user at first time they log into the system. Our system will save the result in a table, when a user wants to get a code, always check if this user has finished the pre-training or not. This pre-training will include knowledge on how to stay safe under COVID-19 especially on campus. We will set several questions in each section to make sure the user has handled the knowledge. A user can be flagged as finished the training only if he or she passes the training questions.

“GPS aided tracking”

We plan to use GPS to track users in order to increase integrity. There are two purposes. The first one is to detect whether this user has been to a high risk area. The second one is to help us evaluate the quality of self quarantine. Once a user is required to self quarantine, we will limit the range of activities of this user according to his/her accommodation. When this user approaches the boundary, he/she will get notified to stay in the area, keep self quarantine. During the quarantine, if a user got out of the quarantine area, this quarantine would be treated as invalid.

“In-person Course ID”

We will record the in-person course ID and users' attendance. If a user has the same course and same class time with another user whose health code turns yellow or red in the future, this user will receive a warning.

2. Who or what will be its users? Your project must impose registration, authenticated login and timeout/explicit logout, so your answer should say something about this.

Users: authenticated users who have columbia emails (students and staff).

They should use their columbia emails to register new accounts. Once registered, they could use their columbia emails and passwords to login in, show the health code or check their health reporting record.

Once successfully logged in, people can enter any part of the campus within the six hours. After the time expires, they will automatically log out and must log in again.

3. Your project must be demoable, but does not need a GUI if there's a command line console or some other way to demonstrate the . (All demos must be entirely online,

there will be no in-person demos.) What do you think you'll be able to show in your demo?

In the demo, we can show the process of how a user interacts with our system and get a healthy code. Also, we can show how the manager views the information of students and staff who use our system today and the current status of users.

4. Your project must store and retrieve some application data persistently (e.g., using a database or key-value store, not just a file), such that when your application terminates and starts up again some arbitrary time later, the data is still there and used for later processing. What kind of data do you plan to store?

We will use a database to store our data persistently.

There will be only one database which contains several tables:

1. "Healthy Status" table will record all users' healthy status
2. "User Information" table will record all registered users' information
3. "Pre-training" table will record the training information of all registered users
4. "Course ID" table will record whether registered users have taken the same in-person courses.

These tables are connected with the key "uni/uuid".

5. Your project must leverage some publicly available API beyond those that "come with" the platform; it is acceptable to use an API for external data retrieval instead of to call a library or service. The API does not need to be a REST API. There are many public APIs linked at <https://github.com/public-apis/public-apis>

(Links to an external site.)

and <https://github.com/n0shake/Public-APIs>

(Links to an external site.)

. What API do you plan to use and what will you use it for?

Since Google Search API is not focused on COVID-19, we decided to change API to COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University. Which is currently the most recognized COVID-19 data source.

We plan to use OCR API to extract report text.

Part 2:

Write three to five user stories for your proposed application, constituting a Minimal Viable Product (MVP); registration/login/logout should not be included among these user stories, nor should 'help' or other generic functionality. That is, your application should do at least three application-specific things. Use the format

< label >: As a < type of user >, I want < some goal > so that < some reason >.

My conditions of satisfaction are < list of common cases and special cases that must work >.

The type of user (role) does not need to be human. You may optionally include a wishlist of additional user stories to add if time permits. Keep in mind that the type of user, the goal, the reason (if applicable within the system), all the common cases and all the special cases must be testable and demoable.

As a **student/staff**, I want to submit the form and update my daily health information so that I would get a green pass code.

As a **manager**, I want to review the information about people who get into campus so that I could log into the platform and check the information table.

Part 3:

Explain how you will conduct acceptance testing on your project. This means that every MVP user story must be associated with a plan for user-level testing. The test plan should address both common cases and special cases. Discuss sample inputs the user or client would enter and the results expected for the corresponding test to pass vs. fail. Note inputs might come from files, network, devices, etc., not necessarily from a GUI or command line, and results might involve changes in application state, files, outgoing network traffics, control of devices, etc., not necessarily outputs via a GUI or command line. You may optionally discuss testing plans for your wishlist additional user stories, if any.

If a user logs in as student/staff, reports no symptom and no explosion to potential risk, and gets a green-pass code, the test case passes.

If a user logs in as student/staff, reports no symptom and explosion to potential risk, and gets a yellow code, the test case passes.

If a user logs in as student/staff, reports symptoms but no explosion to potential risk, and gets a red code, the test case passes.

If a user logs in as student/staff, reports some symptom and explosion to potential risk, and gets a red code, the test case passes.

Manager will be assigned a superuser account. When a manager uses its superuser account to log in, he will have access to all the information about people who visited campus. The interface of superusers' account will be different from others, and will receive a message "Welcome SuperUser!". If a message pops out, then the test will pass, else it will fail.

Part 4:

Identify the specific facilities corresponding to JDK, Eclipse, Maven, CheckStyle, JUnit, Emma, Spotbugs, and SQLite that your team plans to use. That is, state what you plan to use for compiler/runtime (or equivalent), an IDE or code editor, a build tool (or package manager if 'build' not applicable), a style checker, a unit testing tool, a coverage tracking tool, a bug finder (that's not just a style checker), and a persistent data store appropriate for your chosen language(s) and platform(s). If different members of the team plan to use different tools, please explain. It is ok to change your choice of tools later after you start developing your application.

We plan to use eclipse, Maven, Checkstyle, Junit, Emma, Spotbugs, Postman, Mysql for our project.

As for the front-end, we plan to use JavaScript and React to build our interface.