

HW10: Sprint 4 Report, Project

Post-Mortem (75pts)

Replace all the **highlighted** placeholder text, below. Your team may change the style of this template, but please provide all the responses requested and keep the same sections / order.

Usability note: If you double-tap/click the placeholder text, that text should become selected so that you can easily type over it.

Team Information (0pts, but please complete)

What is the **name of your team**?

The Insiders

How many people are on your team?

5

Sprint 4 Increment & Review (48pts)

URL of your **GitHub release**:

<https://github.com/crescents720/Team03-COVIDCoach/releases/tag/v2.0>

Instructions for **how your customer/grader can run your team's project** (e.g., see README file):

See *README.md* for *Instructions on how to run the project*:

Project Title

CS 361 Project Team03-COVIDCoach

Setup:

Create a venv folder within the project folder:

```
$ python3 -m venv venv
```

To activate the environment:

```
$ . venv/bin/activate
```

Within the activated environment:

```
$ pip install Flask  
$ pip install newsapi-python  
$ pip install pandas  
$ pip install flask-bootstrap  
$ pip install flask-sqlalchemy  
$ pip install flask-wtf  
$ pip install email-validator  
$ pip install flask-bcrypt  
$ pip install flask-login  
$ pip install PyJWT  
$ pip install flask-mail
```

```
Start running the application:

$ export FLASK_APP=app.py
$ flask run
```

if meet urllib error

```
Run

/Applications/Python\ 3.7/Install\ Certificates.command
```

if app.db is existed before 4.28, delete it and open python console in pycharm

```
Run

from app.models import db
db.create_all()
```

Or visit this URL: <http://covidcoachcs361.herokuapp.com/>

How to avoid losing points:

- Provide detailed and correct instructions that are easy to follow (if you're not sure the instructions are clear, ask someone outside your team to try them)

Which user stories did your team complete during the last Sprint? Your team is required to have completed four user stories if you're a 5 or 6-person team and three user stories if you're a 4-person team.

User Story 1

User story **description**

As a user, I would like to view System & User Documentation for the covid coach project.

User story **status**

Done

Provide a **screenshot** that shows the current state of this user story.

System & User Documentation Google doc link:

<https://docs.google.com/document/d/1gnjQjvYWzunQ9sfbfRzCLAIvQMUaXWlv39xKz2ym77Y/edit#heading=h.v5whxswyf6pa>

Screenshot

Team Insider

System & User Documentation

For one of your final user stories, your team will need to create system and user | documentation. System documentation describes how the software is designed and how it works internally. User documentation describes how end users can install, run, and use the software. Please include the following:

1. Project

Our Project is COVIDCoach :a website which could show the latest number of confirmed cases worldwide, and the total number in the U.S. Our project can provide the latest news which relates to the COVID-19. The project also can give advice for how to avoid COVID-19 infection and transmission, check in with lonely friends in the discussion board, and could provide the map to find the nearby hospital.

The goal of our project is to help people to learn more and protect themself.

Main features:

- **Home page:** this page displays the total number of confirmed cases, dead cases, critical cases, and recovered cases worldwide and in America. Both of them are shown by chart and table.
- **Get News:** this feature will display the true Coronavirus news and reports for the user to help them understand the situation of the disease. Users can share the news to others or collect the news they like.
- **Get Help:** this feature has 2 sub-options.
 - COVID Instructions: it will introduce the Symptoms of Coronavirus and inform how to avoid the disease.
 - Find closest hospital: it will organize a map to the user and provide the link to the google map and find the closest hospital for the user and the information to contact it

- **Log in/sign up:** this feature will be required as the user wants to access the get talk feature, the user needs to log in to our website. If they don't

have the account our website provides to sign up an account by using the user's email address and the username.

- **Get talk:** this is the feature which the user can go into a discussion board and post or delete their question or ideas and it also can be replied or followed by other users.

2. How end users can install/build, run, and use the software

For our user, they could access our website by searching our title or search for the url. Then they could easily use almost all the features of our website, except the get talk feature without log in. if they want to use the whole website, they just need to sign in for getting an account.

For the web start see the readme.txt :

Setup:

Create a venv folder within the project folder:

```
$ python3 -m venv venv
```

To activate the environment:

```
$ . venv/bin/activate
```

Within the activated environment:

```
$ pip install Flask
$ pip install newsapi-python
$ pip install pandas
$ pip install flask-bootstrap
$ pip install flask-sqlalchemy
$ pip install flask-wtf
$ pip install email-validator
$ pip install flask-bcrypt
$ pip install flask-login
$ pip install PyJWT
$ pip install flask-mail
```

Start running the application:

```
$ export FLASK_APP=app.py
$ flask run
```

if meet urllib error

Run

```
/Applications/Python\ 3.7/Install\ Certificates.command
```

if meet urllib error

Run

```
/Applications/Python\ 3.7/Install\ Certificates.command
```

if app.db is existed before 4.28, delete it and open python console in pycharm

Run

```
from app.models import db  
db.create_all()
```

3. Your project's quality attributes

Quality attributes	Reason
<i>Flexibility</i>	<i>Because our product can bring convenience to them. Our web feature can give them different functions to use and methods to access. They don't need to do other research, only view our web page so they can know the latest news, confined number, hospital and discuss.</i>

Testability

Testing makes it easy to verify code changes. For a complex software system, nothing is more important than stable changes, and high-quality automated testing is one of the most important tools to achieve this goal.

Usability

Only this quality attribute works well, our team can get more users from the competition. It is still the important quality attributes we pay attention to.

Reliability

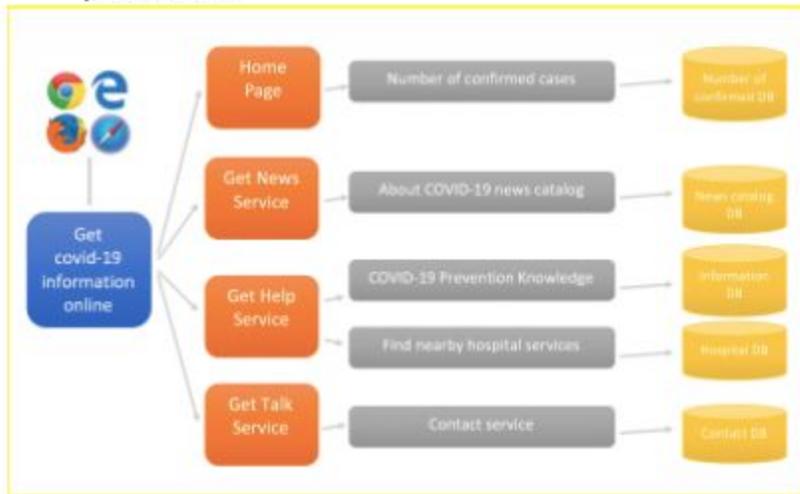
Our team focuses on its reliability, while we are working on designing it. show the good quality of our products.

Integrity

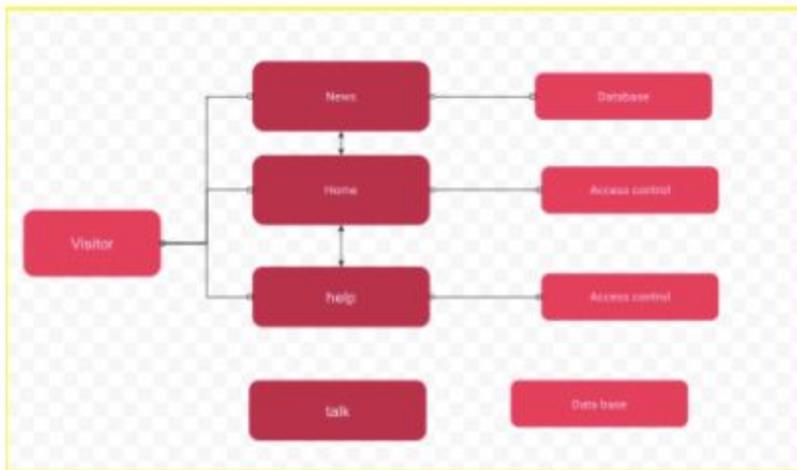
Integrity is the key to give users a good experience.

4. Overview of your software's architecture and design (about 1 or 2 pages)

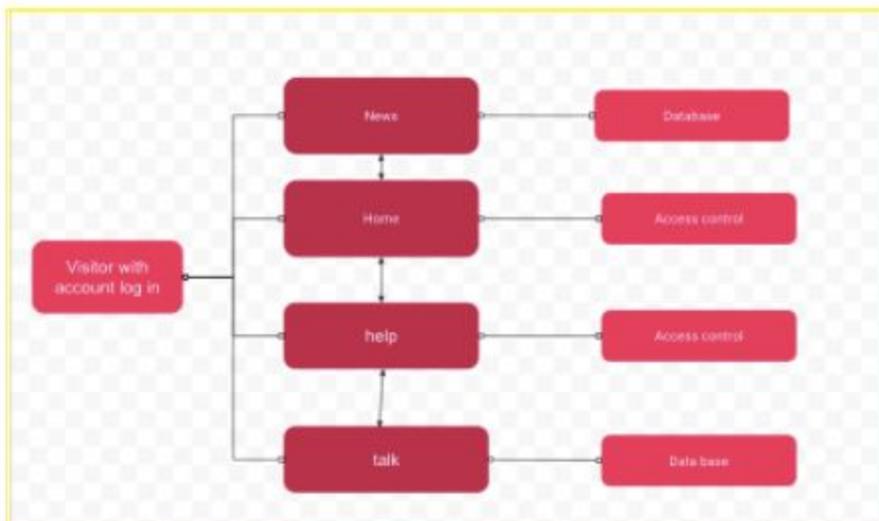
- Include at least **2 diagrams** communicating the architecture/design of your software



This is the beginning of our design. The skeleton of our website looks like these. However as we developing, we find that we only need the News database and Talk board database for the user and recording their behavior. So we change them into below:

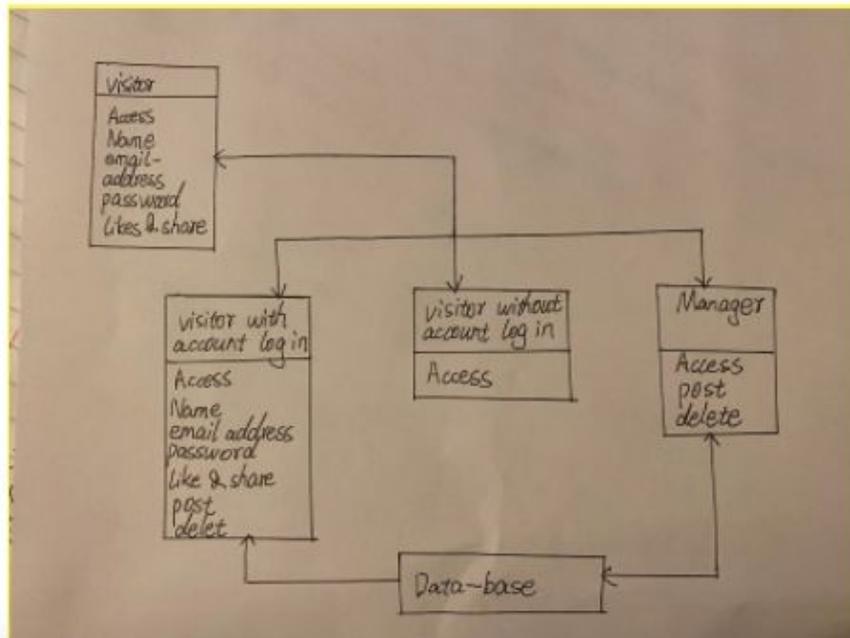


Visitors can access the Get News feature and the home page and the get help feature however it can't go to and use the gat talk feature.



However the Visitor with the account log in can access the whole feature of our web

However the Visitor with the account log in can access the whole feature of our web. And the talk feature and the News feature need the database to store their behavior and the original information. During the development the design Class Diagram looks like below:



Every function and the information variable they need is mentioned in it.

5. What design pattern(s) the software users, and in what modules/components.

Builder: Separate the construction of a complex object from its representation so that the same construction process can create different representations.

For this refactor, we created a package called "app" and instead of putting all import, routes and how to run our app, we split these stuff into different files. Now we have init.py, routes.py and run.py. We also split the models and other functions into different files. For instance, "models.py" and "helperFunctions.py". For those html files, we also use Pycharm move refactor feature to move those into our newly created app package.

6. The final state/condition of the software

- overview of **known bugs/issues:**
- what is left in the **Product Backlog:**

6. The final state/condition of the software

- overview of **known bugs/issues**:
- what is left in the **Product Backlog**:

7. A explanation of the general file structure of the project

Readme.md contains the method to run the web and also provide the objects that need to be installed first. And it also gives the method to dealing with the wrong situation while running the app.

In the App folder: it contains a static folder, template folder, and all the helper function python files used in our website.

Static folder: contain the css, js , image and bootstrap file for styling the web.

Template folder: contain all the html file which built the basic web skeleton of our web.

8. How members of your team and (optionally) new developers can contribute to the project in the future:

All group members can keep the project and if new people want to become contributors, it should require permission from other team members. They need to know the project is aimed at convenienting people on knowing and learning about the COVID-19 disease. It is also a good example for learning and studying the code for setting a website and creating the forum and dealing with the data.

What were the **acceptance tests** for this user story?

Check point 1: we will assign this to the google drive and every group member can see the process and can edit it.

Check point 2: we will set the due date as the deadline.

Check point 3: The documentation has project overview and main features at the top.

Check point 4: The documentation has instructions on how to run the project and quality attributes.

Check point 5: The documentation has software architecture and design

Check point 6: The documentation has flowchart design patterns and final state

Does your team have any **comments or notes** about this user story that you'd like to share with your customer/grader?

All group members can keep the project and if new people want to become contributors, it should require permission from other team members. They need to know the project is aimed at conveniating people on knowing and learning about the COVID-19 disease. It is also a good example for learning and studying the code for setting a website and creating the forum and dealing with the data.

User Story 2

User story **description**

As a user, I want to see the discussion board which includes other people's posts.

User story **status**

Done

Provide a **screenshot** that shows the current state of this user story.

Title	Post Time	Operation
test1	1591556741.793853	View
test 2	1591556793.918387	Delete View

test1

dfs sdf sdf

glnlulu 1591556741.793853

0 Replies

Reply

Post

Are you sure you want to delete this message?

Cancel OK

Title	Post Time	Operation
test1	1591556741.793853	View
test 2	1591556793.918387	Delete View

What were the **acceptance tests** for this user story?

Checkpoint 1: User goes to the homepage of the website, and homepage displays navbar that has the "Get Talk" button

Checkpoint 2: User clicks "Get Talk", renders the message board and a list of posts by other users is displayed.

Checkpoint 3: User clicks the title and directs user to the specific message page

Checkpoint 4: User will see a delete option if the message is posted by the user

Checkpoint 5: User will see a reply option when user clicks the message if that is posted by other user

Does your team have any **comments or notes** about this user story that you'd like to share with your customer/grader?

The discussion board can now be used to view both the logged in users post and other users posts. To make the functionality behave reasonably, our team made a delete button aside the view one. It appears when it comes to the users own post and disappears when it comes to other people.

User Story 3

User story **description**

Website Deployed: As a user I want to find the website on the internet with entering the URL

User story **status**

Done

Provide a **screenshot** that shows the current state of this user story.

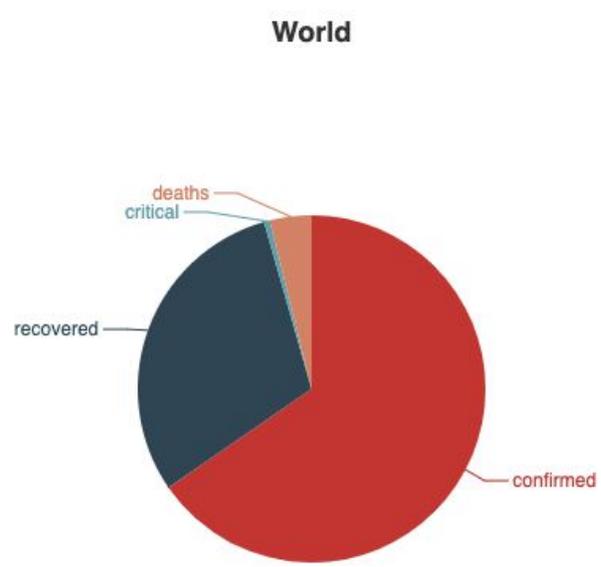
COVID Coach

Not Secure | covidcoachcs361.herokuapp.com

Home Get News Get Help Get Talk Sign Up Log In My Account

Home Page

World



A pie chart titled "World" showing the distribution of COVID-19 cases. The largest slice is red, representing "confirmed" cases. A dark blue slice represents "recovered" cases. A small teal slice represents "critical" cases, and a small orange slice represents "deaths".

Category	Count
confirmed	6481898
recovered	2992614
critical	54474
deaths	383485

confirmed: 6481898
recovered: 2992614
critical: 54474
deaths: 383485
lastChange:2020-06-03T00:52:12+02:00
lastUpdate: 2020-06-03T01:00:04+02:00

Latest Coronavirus News



COVID Coach Get Help

Not Secure | covidcoachcs361.herokuapp.com/help

Home Get News Get Help Get Talk Sign Up Log In My Account

COVID Instructions
Find closest hospital

Instructions

Symptoms & Testing

Call your doctor: If you think you have been exposed to COVID-19 and develop a fever and symptoms, such as cough or difficulty breathing, call your healthcare provider for medical advice.

Symptoms of Coronavirus

Watch for symptoms

People with COVID-19 have had a wide range of symptoms reported – ranging from mild symptoms to severe illness. These symptoms may appear 2-14 days after exposure to the virus:

- Fever
- Cough
- Shortness of breath or difficulty breathing
- Repeated shaking with chills
- Muscle pain
- Headache
- Sore throat
- New loss of taste or smell

When to Seek Medical Attention

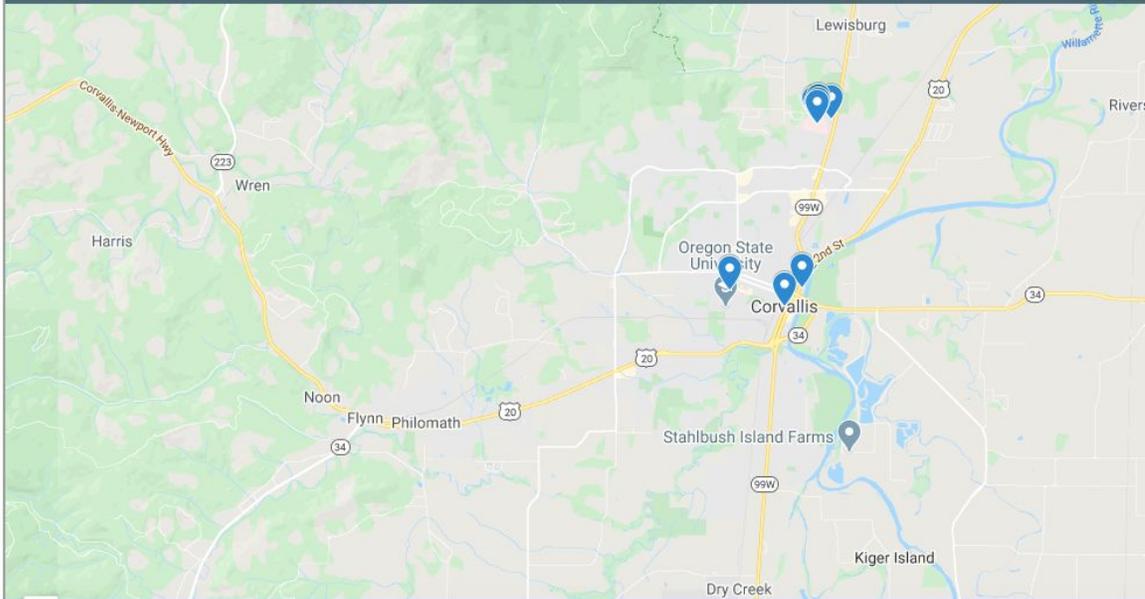
If you develop any of these emergency warning signs* for COVID-19 get medical attention immediately:

- Trouble breathing
- Persistent pain or pressure in the chest
- New confusion or inability to arouse
- Bluish lips or face

*This list is not all inclusive. Please consult your medical provider for any other symptoms that are severe or concerning to you. Call 911 if you have a

Find hospital in Oregon

Oregon Hospitals
This map was created by a user. Learn how to create your own.



COVID Coach Account Register x +

Not Secure | covidcoachcs361.herokuapp.com/register

Home Get News Get Help Get Talk Sign Up Log In My Account

Sign Up

Username

Email

password

Repeat Password

进行人机身份验证 
reCAPTCHA
隐私权 - 使用条款

Register

COVID Coach Account Log In x +

Not Secure | covidcoachcs361.herokuapp.com/login

Home Get News Get Help Get Talk Sign Up Log In My Account

Log In

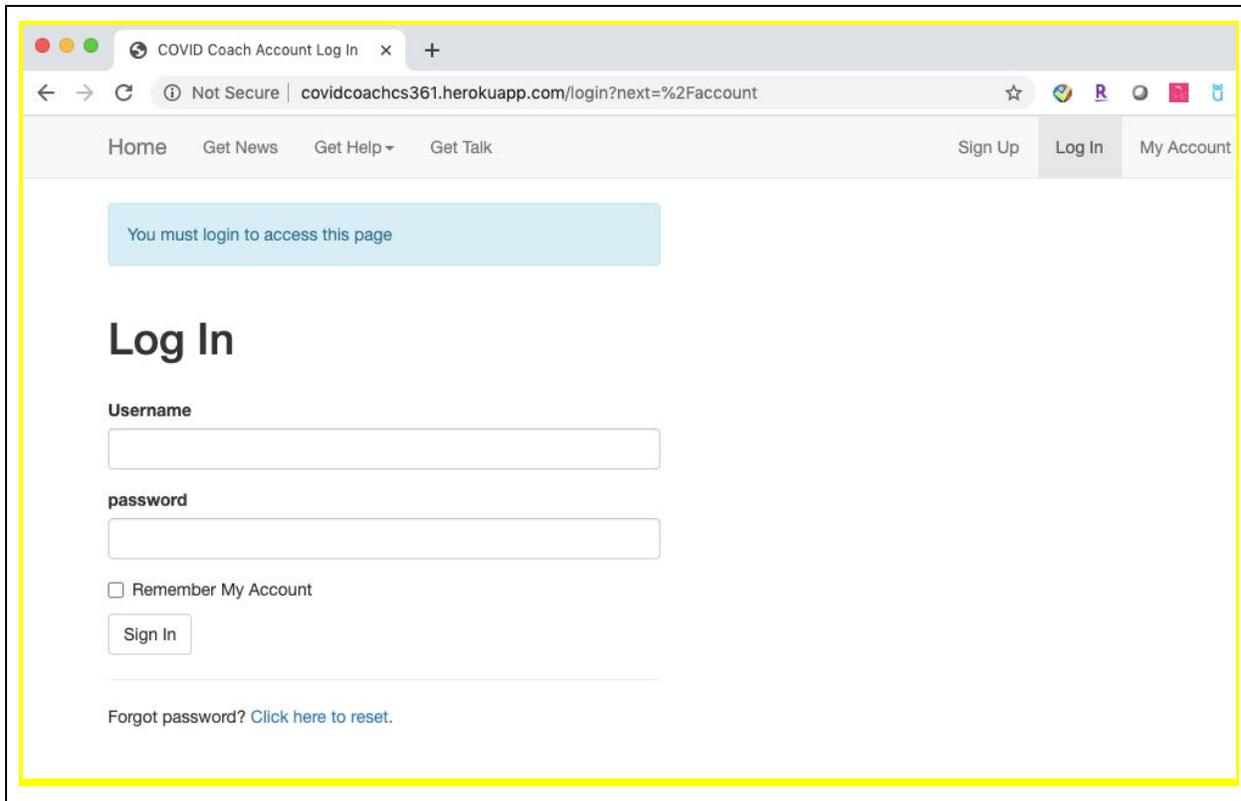
Username

password

Remember My Account

Sign In

Forgot password? [Click here to reset.](#)



What were the **acceptance tests** for this user story?

Checkpoint 1: User enters url of our website in browser and hit enter
Checkpoint 2: Browser directs to our website and shows the homepage of our website
Checkpoint 3: User clicks home will redirect to homepage
Checkpoint 4: User clicks Get News will redirect to news page
Checkpoint 5: User clicks Get Help and will be provided instructions and find hospital options
Checkpoint 6: User clicks Sign up will redirects to sign up page
Checkpoint 7: User clicks Login will redirects to log in page
Checkpoint 8: User clicks My Account when not login will redirects user to log in page
Checkpoint 9: User clicks My Account when login will redirects user to my account page

Does your team have any **comments or notes** about this user story that you'd like to share with your customer/grader?

We use herokuapp to deploy our website. It is harder than we expected but we finally completed deployment of our website. After the deployment, anyone with the website URL is able to visit our website.
URL: <http://covidcoachcs361.herokuapp.com/>
It takes some time to render the page since we use free deployment service.

User Story 4

User story **description**

Database Deployed:

As a user, I want to save my account info online that I don't need extra steps to access my account

User story **status**

Not Complete

Provide a **screenshot** that shows the current state of this user story.

Home Get News Get Help Get Talk Sign Up Log In My Account

Sign Up

Username

Email

password

Repeat Password

ERROR for site owner:
Invalid domain for site key

reCAPTCHA
Privacy - Terms

Register

Internal Server Error

The server encountered an internal error and was unable to complete your request. Either the server is overloaded or there is an error in the application.

What were the **acceptance tests** for this user story?

Checkpoint 1: User clicks register and shows info stored successfully online.
Checkpoint 2: User clicks forget password and enter the email correctly, a message saying reset password email is sent will be showed to the user
Checkpoint 3: User clicks link in the reset password email and enter new password, a message saying password update correctly will be showed to the user
Checkpoint 4: User clicks login and enter new password, the user will login in successfully within seconds

Does your team have any **comments or notes** about this user story that you'd like to share with your customer/grader?

The database is functional locally, however the database migration is incomplete, so on the deployed website users are unable to register and log in because user data is not stored. Heroku does not support sqlite which we used in development, although sqlite is easy to get started and can be run in memory backed by files on disk, and are easily created and moved around, it is not a production grade database. While this strategy works well for development for small projects such as our website covid coach, the Heroku's Cedar stack has an ephemeral filesystem, so the contents will be cleared periodically.
Sqlite does not run as a service, and since we have two databases, each dyno on Heroku would need a separate copy to run on.
After understanding why sqlite is a bad fit for deployment, I tried to synchronize our disk with the dynos, but ran into a lot of troubles. First it was required to make postgres the default database, then modify 'models.py', 'run.py' and add a 'manage.py' to use 'flask-migrate'. Then initialize the database and create the migrations folder. After adding the database addon in our app, we need to apply changes to Heroku database.

Eventually our team failed to configure our app to run on Postgres, but this also gives us a lot of insights regarding future website development and managing databases. So unfortunately, while our team members put in sophisticated work into the message board and functionalities of user accounts, it was not able to be displayed on the deployed website. They still work locally though.

Sprint 4 Retrospective (12pts)

How did the last Sprint go with regards to people, relationships, process, and tools? Write a sentence for each category.

With regards to...	Observations
People:	Our team's Yang Zhang and Yang Zhang have made great contributions to deployment of the website and Yang yang tried so hard to deploy the database even though we couldn't make it. Meng Ding and Zan Zhang have made great efforts in discussion board page integration and organizing documentation for our project. Team members get along well. All members made good contributions to finishing our project.
Relationships:	Our team members have a good relationship and communicate with each other in wechat to solve the problem. These two weeks, Yang Zhang and Zhang Yang are a pair of working groups. Meng Ding and Zan Zhang are working groups. Yiqiao Lu is a master of scrum.
Process:	The process of our work is to communicate before assigning tasks. After everyone has confirmed, they start to work in groups. In this process we will continue to learn new knowledge.
Tools:	Our main contact tools are wechat and asana. The tool to hand over the code is GitHub. The tools for learning are youtube and Google query.

How to avoid losing points:

- Don't be vague

Project Post-Mortem (15 points)

Congratulations, you've finished your project! Now, reflect as a team about what went well during the project, what could have gone better, and what you learned. Write what your team came up with below.

What went well, and why? (at least 3 things)

What	Why
------	-----

<p>Team Communication</p>	<p>Our team uses Wechat mainly for communication. Although our team members are in two different time zones, we are not affected by this situation. We post messages in our group chat no matter what time it is now and other members will reply as soon as possible when they can. We communicate every day even for the weekend.</p>
<p>Task Management</p>	<p>Our team uses Asana for team task tracking. We have different columns under the "board" category such as Sprint 1 Done, Sprint 2 Done, Sprint 3 Backlog and so on. Therefore, we could easily see which task is completed at which sprint and who did that. Every team member has access to manage so our task is always updated and correct.</p>
<p>Code Review</p>	<p>Our team works in pairs for assigned tasks and they will do code review in pairs first. After that, they will explain what the code does and what the interface looks like in the whole group so that everyone could benefit from this. Also, another pair and the scrum master will review the code and give suggestions. Therefore, we always make our code better and maintainable.</p>

What could have gone better, and why? (at least 3 things)

<p>What</p>	<p>Why</p>
<p>Communication Tool</p>	<p>Our team uses Wechat for communication since we all use this app for a couple of years. However, this app has a drawback of showing the dates. For example, the same day message will not show the date of today, only showing the time of the message. And everytime we would like to use the laptop version of the app, the laptop version will not have all the chat history as we have in our phone so we have to look up the history in our phone at the same time. Next time we will</p>

	<p>choose Slack instead since that app covers these problems.</p>
<p>Customer Meeting</p>	<p>Since our team is in two different time zones, we have to choose a time that everyone could fit in. Usually we will choose a time that late for some teammates and early morning for some teammates as well. Also, some members have to go to work even on weekends, so sometimes they cannot speak or listen carefully during the meeting. However, we all participate in the meeting no matter what. There, we had a good meeting with our customer, but this is what we could do better next time.</p>
<p>Resource Sharing</p>	<p>Although our team will share what their code does and the interface design, there is still some aspect like where they learn this or what resources they have that are not shared in the group such as how to set the login function or how to deploy the database. We will do better next time by sharing the resources right after we think it is useful in the whole group.</p>

What did your team learn, overall? (at least 3 things)

<p>What</p>
<p>We learned Flask from scratch of course! And we are so excited about it. Our team has five members and only Meng Ding used Flask for a couple of times. At our first meeting, he recommended Flask highly that it is suitable for our project. So we all researched that and found out it is very useful. Meng sent us a documentation of Flask so that everyone could learn from it. After all, we all get a clear concept of what Flask is and how to use it for a basic website build.</p>
<p>We practice in pair work along with team work. Since we are taking the CS361 class, we must have worked in pairs before. But this time, we have a whole group and also more than 1 pair in it. This pattern is more like real life working in some way so definitely it's better for us to practice in this way as early as possible. Our team members all have giveaways from doing this.</p>

We learned github skills. Some of our team members use github before but some don't. For those who have used github, they use different functions as well. Therefore, our team members could discuss the skills, how to use it and when to use it within the whole group so that everyone learns from the discussion. We also practice what we learn such as pushing code to master branch, creating a new branch, assigning reviews, merging pull requests to master branch and so on. We also learn about github release which is useful as well.