

ECE 568 Homework #1 Danger Log

Team Member:

Chenglong Tang

Yingxu Wang

January 12, 2022

-When entering email information, a user enters a random combination of characters and numbers. But it has to be used to send emails.

We use regular expression to validate the correctness of the input to be sure that user enter correct username, emails and password.

January 12, 2022

-There is no link that we can go back or go to next page of our ride app.

We add button called "go back to home page". It can help user go back to previous page if they do not want to do anything in current page.

January 14, 2022

-What if a user requests a ride whose time is before current day or time?

We set the minimum time for the ride. It will be later than current time. When choosing the arrival time, the minimum time that a user can set is current time.

January 15, 2022

-What is the status of a ride order if there is no driver to confirm that ride and current time exceeds its required time?

We do not know if there exists a function which can update the information inside the database every single time. If it exists, it can check the data in the database to see if current time exceeds its required arrival time or not. If so, it can directly set the status of it to CLOSED. However, what we did is in a more efficient way. Every time only when a user checks its information, we check the data in the database to see if it expires or not. If so, we set it to CLOSED and display to user.

January 16, 2022

-A user cannot join the ride he/she create.

After we implement the join operation for users, we tested it and found that a user can even search the ride he/she created and join it. But it does not make sense. Therefore, we adjust the return value from database. In the return result, we will filter and not return those rides order whose owner is itself.

January 20, 2022

-In the history orders of user, some of them can be edited while others cannot. How to display the information to users correctly?

We divide the rides history into 2 parts. The first part is “Rides History” and the second part is “Future Rides”. Users can check those ride histories for finished orders, outdated orders. They cannot edit those orders. However, for future rides, they can edit it or cancel it. Meanwhile, we will display the role user plays in this ride order. For example, we will display if this user is an owner or sharer in this order. If this user is owner of this order, when clicking edit, he/she can edit information for this ride. When clicking cancel, he/she will cancel this order and this order will not show in the history again, either for owner or sharers. If this user is sharer of this order. When clicking edit, he/she can adjust the number of people he/she will bring to this ride. When trying to change the people, we will always check if total number of passengers will be greater than 8 (The maximum number of passengers we set). If so, we will display a prompt message to tell the user to enter appropriate number. And if a sharer clicks cancel button, he/she will quit this order and does not share it anymore.

January 21, 2022

-A user cannot join a ride several times.

When we were testing the join function, we found that after joining a ride, a sharer can still join that ride again. That is, the result will still appear in the list. However, it is not correct. If a user joins a ride, that ride information should not appear in the share list again.

January 23, 2022

-A driver cannot be the driver of its own ride order.

When testing the search ride order and join function for driver, we found that a driver can search its own ride order. However, it is not correct actually. A user requires a ride, but at the same time he/she confirm its own ride. That is not true. Therefore, we use filter to exclude this portion of data.

January 25, 2022

-What happens when a user wants to join a ride? But at the moment he clicks join, a driver already clicked confirm button for this ride order?

For this situation, we can imagine that a person opens a web page but does not do anything to it until next day morning. The information displayed in the web page has already outdated. Therefore, what we should do is to check every time when a driver clicks confirm or when a sharer wants to join a ride.

January 27, 2022

-We find that the created time in the database is always 5 hours faster than current time even if we set `USE_TIMEZONE = FALSE`. And it makes OPEN ride orders to be closed within 5 hours.

At first we think it's the problem of `datetime.now` function in the Django. However, after testing and printing all the results, we find that there's no problem with it. Then we test several times, and find the time is always 5 hours faster than current time. So, we think it might be the problem of timezone in Django. Even we set `USE_TIMEZONE`

= false, it will automatically use the timezone set in the setting.py. Then we change it from UTC to EST. Everything works correctly now.

January 29, 2022

-We find that even we provide date time for user to choose, users can still choose a date by inputting date time. We use `<input type="date">` to get date from users. Although the date inside our chosen list is all valid, users can still input a time like February 31, which does not exist. The system inside will automatically regard it as January 31. Both front-end and back-end will think the time chosen is January 31, which is not correct.

To solve this problem, we disable the input by keyboard. In this way, users can only choose a valid date from us. And doing so will prevent us from getting an invalid input.