ECE 568 Ride Sharing Service Danger Log

Yangfan Ye, Toshiko Li

1 User

1.1 Sign Up

1. Danger: client could sign up by scripts without limitations

   Solution: sign up with email as authentication

2. Danger: client could sign up by other's email

   Solution: use verification code. we send a verification code to the email and verify it in sign up progress

3. Danger: client could type their passwords wrong by accidents when signing up

   Solution: ask client to type password twice when signing up

1.2 Register as Driver

1. Danger: driver may type their vehicle information wrong or need to update it in the future

   Solution: we designed `revise` function for driver user

2. Danger: the vehicle types that customers want may mismatch what drivers typed, even though they are the same. For instance, driver has `sedan` but customer requests `saloon`

   Solution: We designed several specific vehicle types, driver can select from them when register, and ride owner can select what they want.

1.3 User Information

1. Danger: client could request other users' information by magically typing the right URL

   Solution: we use decorator `@login_required` offered by Django to protect user informations, and we keep the sessions so the user don't need to log in for every requests

2 ride

2.1 Register New Ride

1. User login statues:

   - Danger: Users who are not logged in will also be able to visit the page to register new rides.

     Solution: Add a `@decorator` at the view function `new_ride_view` to verify whether the user of the `GET` request has been logged in, if not then jump to the login site.

2. Form validation:

   (a) Danger: Departure & Destination: An unsuspecting user may request a ride from A to A. This does not cause material damage, but since no driver will confirm the order, such a ride will contaminate the search results.

   Solution: In order to prevent such illegal rides from being created, it is necessary to perform separate clean of the Destination and Departure place during the form validation.

   (b) Danger: Unexpected Arrival Time: If you ask users to enter their own ETA, it's not humane and can lead to a series of weird issues, such as the ETA is 9:30 a.m. yesterday...

   Solution: A front-end API is provided to allow you to select a timestamp only 7 days after the current time node

   (c) Danger: Car type limitation: Allowing users to fill in their own fields when choosing the type of vehicle they expect will make it difficult to match the target driver.

   Solution: With 7 different types of vehicle types and the option of Any to choose from, users can easily add new vehicle types.

2.2 Revise Ride Info

1. User login status:

   - Danger: If the user using the GET request is not a logged-in user, but still has access to, even revise the ride's information, it is not safe for the user's privacy.

     Solution: Add a `@decorator` at the view function `revise_ride_info` to verify whether the user of the `GET` and `POST` request has been logged in, if not then jump to the login site. At the same time, we need to verify whether the user who sends the `request` is the actual owner of the current requested ride, if not, the actual owner will be redirected to another location.

- Form validation:
  - Danger: Since it cannot be assumed that only one person can access the status of the same ride, whether or not the current ride's modified commit form can be validated successfully depends on the status of the current ride. Solution: The status of the ride needs to be verified separately during the form validation, and if the current ride is confirmed by the driver just before the modification is submitted, the form validation will fail.

2.3 Search Ride Requests as Driver

1. Danger: Drivers may find the orders of themselves, but we define accepting orders of their own as illegal

   Solution: We filter the ride orders so that driver can not see their own orders when searching ride requests

2. Danger: Drivers may want to filter orders by specific time or vehicle type

   Solution: We implement `search` function for this need

3. Danger: The current total passenger of a ride order may exceed capacity of driver's vehicle

   Solution: We filter them so the drivers can not see it

2.4 Search Ride Requests as Sharer

1. Danger: Sharers may want to filter orders by specific time or vehicle type

   Solution: We implement `search` function for this need

2. Danger: Users could join their own ride orders, we define it as illegal

   Solution: We filter them so that users can not see their own orders in sharing rides page

2.5 Join Ride as Sharer

1. Danger: Ride order information like destination can be changed by ride owner, but it may not suits with sharers

   Solution: Sharers can only join pending ride orders. After being confirmed by driver, the order information can not be changed anymore

2.6 View Ride Info

1. Danger: Different user types should have different content presentations for the same ride.

Solution: For the driver, the ride information will be displayed along with a `POST` button to receive orders. For users/share users, displaying ride information will provide different information densities depending on the status of ride.

2.7 Cancel Share Ride & Cancel Driver Ride

1. Danger: For these two forms, if another user incorrectly requests to cancel the ride, it will cause a mismatch between the number of passengers in the database and the actual number of passengers, and even cause a serious mismatch between the information on the driver's side and the passenger's side.

   Solution: Direct the user to the correct address by verifying that the user is the shared user or the driver of the requested ride.

2.8 Conplete Ride

1. Danger: The change in the status of the ride after confirmation should not be influenced by anyone other than the driver. No one else, including ride owners and sharers, should have the right to change the status of a ride.

   Solution: It has the most stringent identity checks, which needs to be met at the same time that the requested user is logged in, is the driver of the ride, and the status of the ride is `confirmed`.