MiddRides App User Experience Doc

Views:

(1) **Sign in View:** A user can login through this view by entering their username and password. They can also choose to register.

(2) **Register View:** A user with valid credentials can register to become a user of the app. Note that only passengers can register. Dispatcher accounts will be fixed.

Requirements for registration:

- User must have a valid @middlebury.edu email address
- User must enter a secure password (1 letter and 1 word at least)
- User must confirm their email address by following a URL sent to their email.
- User must enter their first and last name
- User must agree to a small social contract which indicates to be nice and considerate when interacting with MiddRides staff.

(3) **Announcement View:** Any announcements which made by Pubsafe will be present here for the passenger to read.

(4) **Passenger (van request) view:** *If a van is available*, this view allows the passenger to request a van. To request a van, a passenger must choose a valid pickup location. Once a request is made, the number of passengers waiting on the DispatcherView (#7 below) for that stop must increase by 1.

If a van is not available, this view should inform the passenger that MiddRides is currently not operating.

Stops:
Adirondack Circle
R Lot
Robert A Jones' 59 House
Track Lot/KDR
T Lot
McCullough Student Center
E Lot
Q Lot
Frog Hollow
//Each user can make only one request.

(5) **Van arrived alert view:** Informs the passenger that they have requested a van at (stop name). Once the van is headed to that stop, it should inform the passenger that the van is approaching and that they should be at the stop.

(6) **Passenger settings view:** Allows a passenger to adjust the app's settings. (Settings TBD)
(7) **Dispatcher view:** This view will contain a map encompassing all the stops, with circles on each stop indicating how many passengers are waiting. This view allows the dispatcher to perform the following actions:
    (a) Turn the middrides service on or off. If the service is off, this should be reflected in the passenger view (#4 above).
    (b) Increase the number of passengers who are waiting at a stop.
    (c) Mark a stop as "currently being visited". This should alert all passengers subscribed to that stop that the van is arriving (this should happen on #5 above). Once a stop has been visited the number of passengers in that stop should be zeroed out. If the service has been turned off, zero out all passengers in all the stops.
(8) **Make Announcement View:** Dispatcher can use this view to make an announcement to all passengers. These announcements will be shown to the user in the Announcement view (#3 above).
(9) **Dispatcher settings view:** Dispatcher will be able to adjust the app settings from this view. (Settings TBD)

Problems/Questions to think about:

- How can we coordinate between Dispatchers turning off the service? For example, if there are two vans, then we don't want one of them to turn off the service for passengers before the other van has stopped operating. Solutions might include:
    - Requiring that the van service be turned off from the web API
    - Requiring that both vans "sign off" on service shutdown (service will not say it's turned off until both vans have said they are off duty)