

## Demo of the progress & talk of bug fixes

- DML mentions problem with our MVP
  - Corrects our grade (because he had created the ride in the past)
  - Make sure to check on the database side that the ride isn't created in the past, not just with UI
- Explain that we won't use passphrases because people would have to keep checking when people leave/join rides
- He says it might be nice to send out a reminder that a ride you joined is coming up soon with rides details
  - Emails- Gmail wrappers, Mandrill, twill,
- Talk about errors
  - Return a response in the 400s range (200s are success, 300s are redirect codes, 400s are error codes, 500s are server side errors), error message if you try to create a ride in the past.
  - Put the post request back in?
  - At minimum put a response in the server (?)
  - He will test through the terminal, bypassing the UI so we must be able to handle that
  - He might try to leave rides as another user, or join a ride as someone else. (Cross site request forgery)
    - Make sure to have specific User IDs for each request
    - CSRF ex: click a link and perform an action specific to a user
    - Look at notes to prevent attacks with CSRF
    - To fix: Update all renders to include csrf token
- Testing: model tests are sufficient
  - If we want to/ have time, test routes
  - Test helper functions

## Comments about MVP and initial design

- Authentication: other teams should have had that also, we should all get points off for this if we are missing this from the MVP, the site crashed completely with an incorrect password
- We shouldn't be able to create rides in the past; this isn't how our app is supposed to work

- Email him about any other points we want back to get a more detailed reason.
- For data models – there was no distinction between a creator and a rider, so our first insight isn't insightful -> if it could have been included in the data model, they are not insights

Next steps:

- Sufficient to check that the users register with an email @mit.edu
- Location based search: you can store longitude and latitude, mongo can support us asking for any location that is within 30 miles, look for geopoint? Geolocation? You can ask for square or circular fence.  
->the query can support searches on both ends for proximity



