# Agenda:

- MVP Feedback (see below)
- Progress since last meeting:
  - Achieved milestones:
    - Implemented reviews
    - Ride details page
    - Passwords hashed
    - Google Maps support for creating new rides
    - Prevent users from creating rides in the past or more than 2 weeks in the future
    - Prevent users from creating rides with <2 capacity
  - Missed milestones:
    - confirming @mit.edu email
  - Things we decided to change:
    - Creators can now kick people out instead of a voting system
    - Decided not to use passphrases
- Questions:
  - how to do security?
  - feedback on near final app?
- Next steps:
  - confirm @mit.edu email
  - location based search on rides page
  - clean up code
  - finish writing up tests
  - errors can't show stack trace

## TA's Comments about MVP

- Login/Authentication Feature
  - Passwords are stored as plain strings (insecure). These should be hashed. You can look into PassportJS, a npm package, and use it for authentication. (-2)
    - We didn't say we were going to implement this for the MVP. Other groups didn't get points off for this.
  - There is no prorogation of error. If a user tries to login with an incorrect password it crashes without any propagation of errors
  - Authentication feature incomplete (-4)
    - What do you mean by this?
- Creating Rides
  - API allows you to create a ride in the past. This should be result in an error as it shouldn't be allowed (-2)
    - We handle the error by not allowing other users to join the ride, but still have the ride show up in case someone wants to log a past ride.
- RESTful Routes
  - Good RESTful routes design, but not making use them properly. (-2)

        - For example, in POST /reviews/:review, you're also passing in the reviewID in the body. This review ID should be the parameter that is passed in the URL :review.
        - Reviews was not in our MVP.
- Shared rides feature (main feature) incomplete / broken (-15)
    - User A created a ride for 4 people. User B logs in and can't see any open rides.
    - We tried it in our app and it works fine.
    - Is the ride created in the past? If so, it is a design choice, not a bug.

## TA’s Comments about Initial Design

- Data Model
    - User
        - User rates User is an action and not a relation. The data model should describe the system state but not actions. A user could have a rating that is an aggression of all reviews. (-2)
    - Immutability markings are missing in some cases and wrong in others. (-2)
        - Why can’t the ride a location or time is associated with change? It seems you’re missing an object that encompasses both the location and time.
        - Design choice for users to encourage commitment and to prevent people from getting screwed if they don’t check the ride details later.
    - Ride is a scheduled set of users, time and location. What about the actual transportation mean (personal car, taxi, uber, etc.)? (-2)
    - No thoughtful insights. Your current insights are depicted by the submitted data model. A good insight would encompass something that is not obvious and isn’t obvious at first sight. (-6)
        - **Our insights were thoughtful.**
        - **Insights:**
        - **1. A user can leave a ride he has created.**
            - This implies that other users in the ride may remain in the ride, and that we didn’t have the concept of an “owner” of the ride, so nobody “becomes” the new owner when the creator leaves.
        - **2. A user can sign up for multiple rides at the same departure time.**
            - This was a design decision, because we didn’t have a way of determining whether a user was in overlapping rides. We decided to allow the consequences of doing this be bad reviews from other users.
        - **TA’s insight:**
            - Ride deletes itself when everyone leaves.
                - We put this as a textual constraint, which makes more sense.
- Security (-7)
    - Look over the security recitation and discuss how you’ll prevent the discussed security issues.
- Wireframes/UI
    - Nice wireframes!
- Design Challenges
    - Good design challenges. However, please format them in a more organized manner that states the challenge, the considered