- Design
    - Any feedback for the data model?
        - **Data model follows textual constraints: users can't follow themselves. If you have an arrow of something going to itself, there should usually be a corresponding textual constraint**
        - **Have a label for every arrow: eg. User to username**
    - Also do the goals for final product have a sufficient level of complexity?
        - **Feeds by tags are pretty key to content discovery**
    - Should we have Activity set?
        - Could be Post
        - Could also be upvoting a project, or creating a new project
        - Activity would have a timestamp and is associated with a user
        - Use Activity to generate Activity Feed
            - **Could aggregate all and send them to the user but one Activity set for all is cleaner, less redundant**
- Framework
    - Should we have a separate url for each page?
        - Annoying to refresh with current system
            - **Not a problem**
    - Session persistence
        - Should session survive server restarts?
            - **Minor issue**
- Front-end
    - How to implement error handling?
        - e.g. we could add a pop-up error dialog whenever an error response is received
            - **Since we're using handlebars we could have a div that we populate every time there's an error**
    - User profile page - how to show for other users?
        - e.g. we could change "My Projects" to just "Projects" on the current user page, and use that for the general user profile page at /users/{username}
        - or we can have one page to view your own profile, and another that is displayed if you're viewing profile of some other user
            - **Have a profile page /username. Better since you can send a friend a link to your profile page**
- Other
    - Videos
        - Finalize how to store video media
        - e.g. as a Youtube video ID
            - User enters either Youtube video ID or URL; if URL, then we try to extract the ID, and store as ID in database

- **Either way works. ID somewhat safer since a user could put a weird URL**

- Rating
  - How to rate projects on trending projects view?
  - e.g. could just segment by day or week, so all projects today are shown first, then yesterday projects, etc.; then rate each time segment by upvotes
  - or could do something that takes into account both age and upvotes
    - **Let's stick with sorting by date, then by upvotes**
    - **If someone's makes an error on a project, they should be able to edit it. And a new version of the project should be a new project altogether**
- Workflow
  - Pull requests
    - We said in team contract that we wanted to have code reviews
    - But we haven't been doing this for starter code
    - Should we start doing this? Or just review after code is committed?
  - Unit testing
    - Are we satisfied with current unit testing strategy?
      - **Up to us to decide**

FEEDBACK ON DEMO

- Looks good
- Show that user has already favorited a project
- Show that user has already upvoted a project

AFTER MEETING

- Next deadline? Tomorrow, Reflections due!
- What's next?
  - Implementation left (from our revised teamwork plan)
    - Security
    - Authenticating that users are MIT by emailing activation link
    - Rich UI styling
    - Searching for projects and filtering by project type won't be implemented.
    - Terms of service (to protect intellectual property rights)
    - Featured projects - so old / low vote projects get more visibility
    - Trending projects
    - Following users, activity feed
- To do before 11pm today:
  - Other users' profile pages. Should be easy to do since we decided to just copy over template from current user's profile page minus Favorites.

HAPPY THANKSGIVING!