Design Critique: UROP.io

by $\mathcal{D}_\rho$ (drachma@mit.edu)

Main Design Doc

- Good choice of problem and consideration of audience.
- Goals: sound like features. Should be more high-level: "Help more freshmen get UROPs," "Improve student satisfaction with UROPs."
- Concepts: concise, understandable to MIT audience.
- Features: good, but maybe name them all with nouns?
- Security: you address some, but not all, mitigations? Provide a threat model. How will you deal with spam and inaccurate reviews? How do you ensure that a student has only one account?

Design Challenges

- Review quality: will spam reviews be moderated?
- Review visibility: does allowing a supervisor to see a review taint the honesty of the review?
  - We don't think a student with a bad UROP experience actually talks it over with his supervisor.
- Site purpose hijacking: how do you ensure that the app is used as intended, i.e. not as ratemyprofessors.com?
- Convenience: do you intend for your app to replace other media: posters, emails, departmental websites?

Diagrams

- Context: differentiate between faculty supervisor and direct supervisor?
- Data model:
  - You have two "reader" relations, Supervisor → Review and Student → Posting, but you say earlier that all supervisors can see all reviews, and all students can see all postings.
  - Missing stuff: What's in a posting? What's in a review? Are these related to departments? Are reviews about supervisors or postings?
  - What happens to an accepted application? Do you record whether a student actually takes a UROP? If not, how do you know who's allowed to review a posting?
- State machine: Separate the application and review pages into "listing"/"single" pages. How do you create a posting?

For More Info...

- See our full critique doc, attached.