Parser Formatting

- Will we be expected to create a parser for user-submitted code, or can we use an already existing parser generator (yacc/bison/lemon)?
    - *Ureel will provide a parser.*
    - If the former, what languages will we be supporting?
        - *Java. Other languages will ideally be possible to add.*

Storing User Submissions

- Should user submissions be stored in a database after feedback is returned?
    - If so, how should those snippets be stored?
        - *Currently in the same database. Ureel is okay with client-side storage.*
- Why are we storing the user's code?
    - How do we expect to use this stored code?
        - *A score allows us to numerically evaluate data. The interpretation of the "Score" system is up to us. Potential data, such as time interval between submissions, could be factored into the score.*
        - *We can return feedback on the data patterns (such as time between a critique being given and it being addressed) to the client as well.*
- Should we store the user submitted code and critiques in a separate text file, or just link critiques to line numbers on the page? Should we have the option to output a txt or PDF file with the code and critiques listed out?
- How should we give the user's critiqued code back to them?
    - Give a unique link for each submission and have the user store the link themselves? Or have it linked to a user's account, session, cookies, or ip address?
        - *Display as HTML, optionally provide file download.*
- When (not necessarily if) should we empty the saved code critiques?
    - Should we wipe a submission X days after it's been critiqued?
    - How can/should we prevent spam from filling the db?

Gauging a Student's Growth Over Time

- Should we be creating a user login authentication system in order to keep track of user's growth over time?
    - How would we create/measure statistical growth within the user's growth? (graph?, past history list?)

Security

- How can we prevent db injection, or spam? We were thinking making a unique link (potentially with time constraints) per user instead of full user authentication system with usernames and passwords.

    - *This constraint does not exist if we store data locally.*
    - *If we store data on our server, users need to be authenticated (Spring?).*
- How can we prevent a user from accessing other user’s submissions, critiques, etc.? This implementation changes with how we keep track of a user’s submission (by ip, cookies, …)
    - *Ureel says “at least for the duration of the course.” Realistically, a span of 5-6 months (roughly one semester) is ideal.*

If we feel we have a significant deliverable before the end of the semester, we can submit to a conference or The Infinite Loop.