

Meeting Minutes

Participants: Rachel, Angus, Usman, Suniyya, Noor

Action Items

- Have agenda in repo in advance **(for subsequent meetings)**
- Have a layout file, within views. Repeated features in the view won't need to be typed up in each of the pages then.
- In design, mention that we will be using some kind of hashing for links (e.g. by using occasion id's)
- Do away with the privacy concept. It shouldn't be called a concept

Rachel's Recommendations

- Use Facebook for authentication and sharing
- Look into Passport for user authentication if that is useful
- Look into bcrypt to encrypt passwords
- Support upload of images in the final version
- Don't use both email and facebook for authentication: choose one
- If we use email as identifiers for users in our mongoose schema, we should *specify* that
- Implement soft deletes

Detailed Discussion:

*Rachel's comments are in blue

Went over the progress made thus far, discussing the following:

1. Concepts, specifically the Privacy settings: Replacing that with the concepts of Participant and Recipient which are decided by the Creator of the occasion. Privacy settings are then functions of these concepts. Comment by Rachel: [Good design decision but this isn't a concept e.g the concept of a tweeter \(not a concept\). Permissions should be embedded in other concepts.](#)
2. Design challenges regarding who sees what can be reflected in the data model and the implementation. This is a bit of a routine notion and so it shouldn't be called a concept. Occasions and thoughts will be sufficient. Now changed. Changes discussed. Rachel's comments were: [how can celebrate one occasion in one place and make it easy to involve other people in one place. One place.](#) So for the description of the concept we really just need a higher level idea.
3. Explained the change to the data model (in response to comments on the confusion caused by Role). Role was an ordered pair of (User, Occasion) which is how we imagined how entries would be in a SQL table. This makes more sense, Rachel said. [If this how we are designing](#)

around, then Role makes sense. Initially the criticism was based on a different understanding of our implementation.

4. Facebook issue: It depends how we are using Facebook. Do we require everybody to use Facebook?
5. Security: Biggest security concern: How will we hash the links? We don't just want links to say .../sam'sbirthday. We want some kind of id. We need to mention this. Even with the link, we decided user needs to authenticate themselves. (Rachel agrees [this is good](#)) Seems like Facebook authentication is the way to go but we will have a discussion about that as a team. [As long as you are preventing against the main, standard attacks, you don't need to worry too much about this. Don't worry much about man-in-the-middle attacks. May not be worth our time to set up SSL. On heroku, they already use secure https. Focus more on design.](#)
6. Encryption: [Definitely encrypt passwords. Use bcrypt unless you use facebook which will then be obsolete.](#)
7. Mongoose and Delete from DB: Changing a boolean tag (a soft delete) versus a permanent delete. Most systems do this, Rachel said. [It shouldn't be calling you problems but soft deletes are actually a good design decision.](#) If problem, email Rachel and send her the file.
8. Mongoose identifier for Users: Using user id or email. If we use emails, when we add people to an occasion it'll be easier from a coding perspective. Under the assumption that the user types up the emails of people who can see, querying by email would be good. Rachel says [if email, specify that email is unique. But we should be able to identify using either](#)
9. Problem with going from one page to another: `res.render('filename')` in the routes/register file. Turn button into href (`/register`). The change is on Usman's computer. ajax expects something back from the server. that's asynchronous. If we need to do computation, to submit we say inside the HTML where the form is defined, `type= POST...`
10. Pictures: can store in bitmap. Space is the main concern. Ask for image urls. So that we don't host it. Concern: cross site scripting attacks? For the purposes of this class, it's okay... We could require all image urls end in .bmp or .jpeg. Nothing further. [Should definitely support images in some way.](#)
11. Friending. If we want to use Facebook friends to help invite people to events, we might as well use Facebook more. How we use Facebook depends on how we will decide on event sharing. [Good opportunity to use facebook.](#) Rachel recommended this idea. But we [should choose: either invite by email or by Facebook, but do not do both.](#)
12. Our MVP Concerns:

- a. User authentication: Do we do facebook or email and do that for the final version or also for the MVP?
- b. For sharing we were just thinking about copy pasting the link.

Rachel: You should think about and decide on how to invite people to occasions first. Don't write a bunch of code that will have to be discarded later.

If no Facebook, then allow friending within your own system. [don't like this idea]

Look into authentication using Passport.

Don't discard Facebook just because you think it'll be difficult