The Loop: Design Documentation

Calvin Li

Richard Lu

Christian Mata

Mikael Mengistu

Motivation

The Loop is a web application that allows students to view and publicize MIT events in a centralized location. With The Loop, users will be able to filter to find events of interest, RSVP to events, and follow groups. Group admins will be able to create events, edit event information, add and remove group admins and count RSVPs on behalf of their group. This app provides an alternate medium to share events from previous forms of bulletin boards in the Infinite Corridor, spammy email blasts and pesty Facebook event invites. This allows students to effectively seek out events rather than having events presented to them when not sought after.

Purposes:

- Provide centralized portal for student run events happening at MIT
- Give focused, alternate medium for advertising events
- Allow students to express/gauge interest in the activities

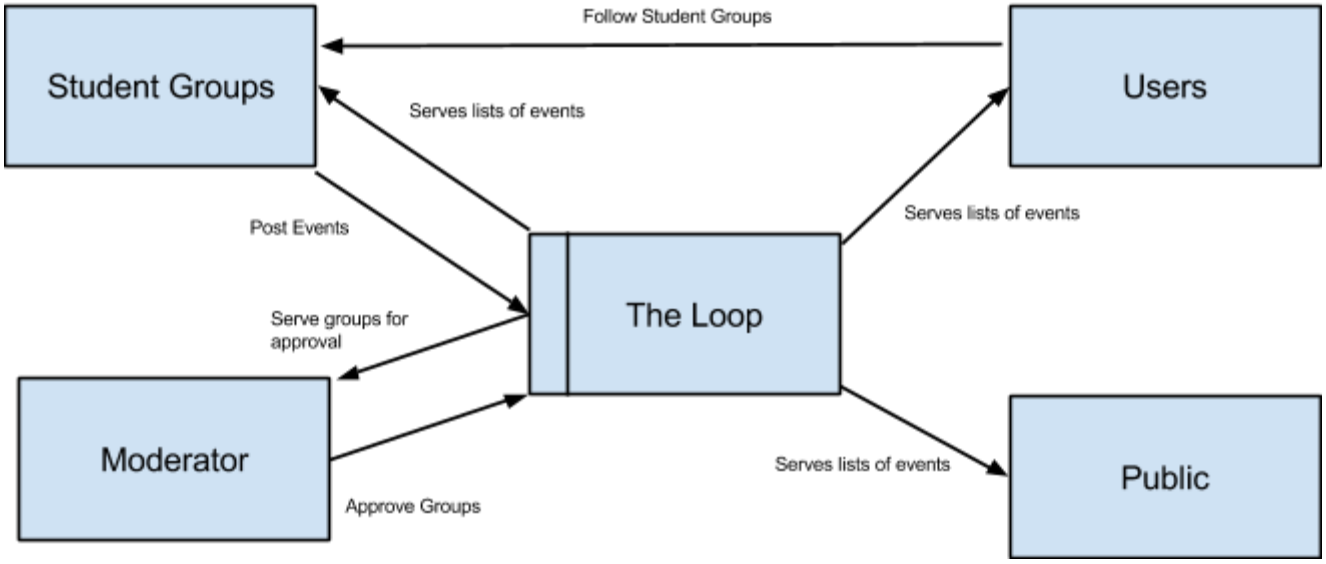
For the first purpose “provide centralized portal for student run events happening at MIT”, The Loop is the first MIT-oriented web app to post and view events in general. Currently if a student wants to find events that are occurring on campus, he/she is left looking through emails or staring at bulletin boards and The Loop aims to improve the effectiveness of this process.

For “give focused, alternate medium for advertising events”, as it is now, student groups rely on various ineffective mediums for publicizing their events through emails, event invites and flyers. Using The Loop allows the groups to have one more alternative for sharing their event that is completely dedicated to displaying events and makes it easier for those who would be interested in their events to find it.

For the last reason, the Loop includes a feature that allows students to RSVP to events and express their interest in attending. This allows the student to show that they are willing to attend and allows groups to gauge interest in the event their hosting and get an estimate of the number of guests.

Currently, there does not exist a solution that provides a single event directory for MIT student group events. The closest solution that is currently available is events.mit.edu, which has some events, but isn’t comprehensive. The other solutions to publicizing events mentioned before are bulletin boards located around campus, spammy email blast sent to mailing lists and Facebook event invitations.

Context



Concepts

Main Feed - Feed of upcoming events, filterable by location, tag and other categories.

Group Feed - Feed of upcoming events hosted by a group.

Events - A group's entry for a specific activity that will occur at some point in time.

Tags (of events) - Predefined identifiers that can be used to help distinguish events and make events filterable.

Site administrators: - Can edit/delete any event, revoke privileges for any group.

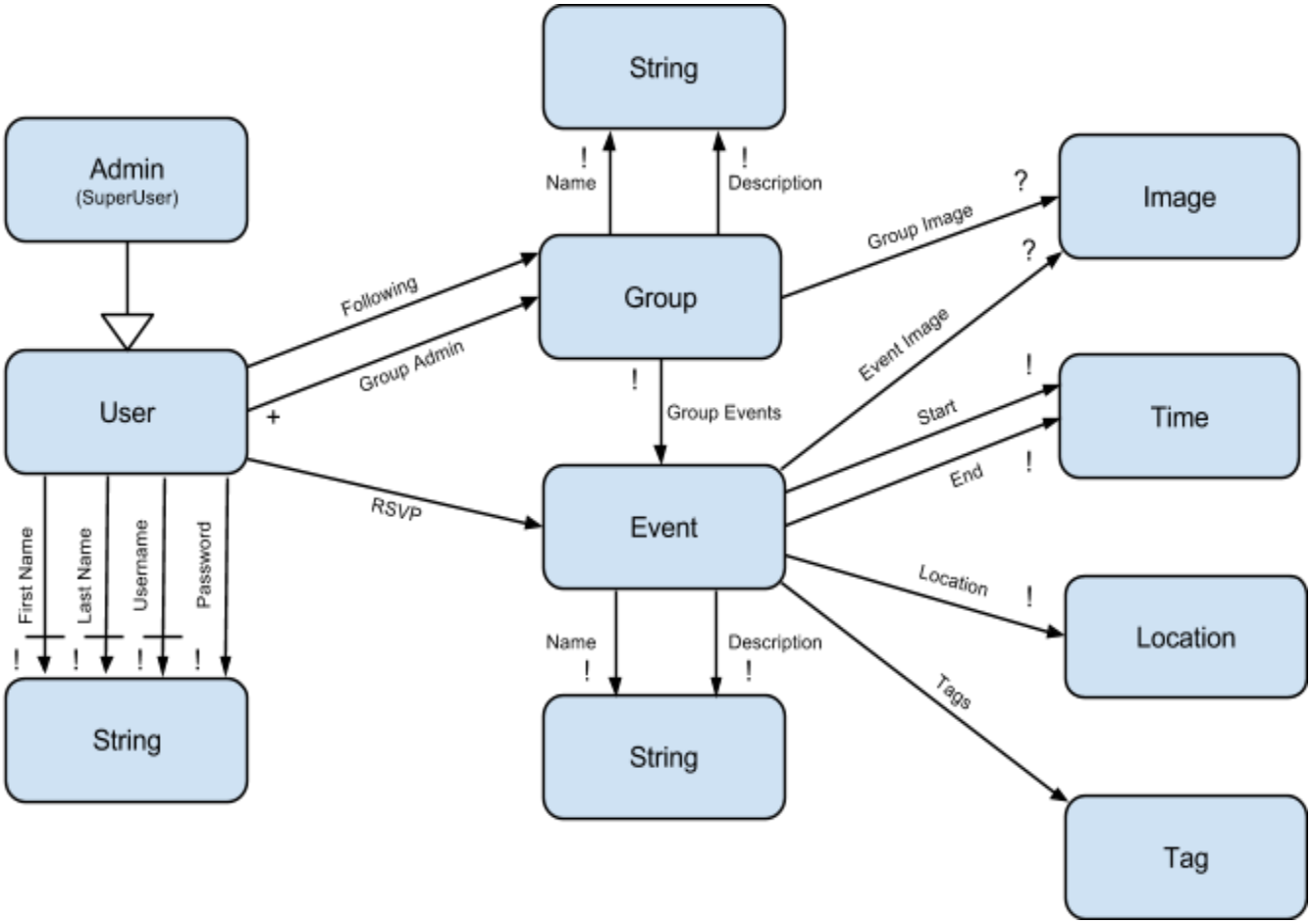
Groups - A student club or group with the ability to contribute events.

Group administrators - Users appointed by current group admins, can administer group's events.

RSVP - A response that indicates a user's interest in attending a specific event.

Following groups - A user's ability to choose groups whose events interest them and save the subset of groups. Allows user to later filter by this subset quickly.

Data model



Security Concerns

Some of security concerns were addressed in the **Risks and Mitigations** section, but we will mention them again.

1. **Account Hacking**

   Some of the users are admins for the groups, so anything the user posts will be representative of the group. As a result, if a user has his/her account compromised, it will provide a negative image on the group if any malicious information is added onto the website, where the public can see it. In addition, admins of the entire site have all access to creating, editing, and deleting anything, so having their login information compromised is a security breach for our entire application.

Mitigation: We will use a strong password encryption such as bcrypt so that no one can access the information if they somehow have access to our database. In addition, we will also require a strong password (greater than 8 chars, with letter and number) for all users so hackers cannot easily decrypt them.

2. **Injection**

   Because our website primarily displays user-created information, we have to worry about malicious users injecting HTML or javascript into our database that may be displayed on our website.

   Mitigation: We will sanitize our text inputs (names, descriptions, username, etc) by escaping dangerous special characters like ampersand, less than, greater than, etc into &, <, >, etc. before inserting it into the database. The only exception would be password, which will never be displayed on the website.

3. **User Authorization**

   We do not want people outside of MIT using the application as they may promote activities or events that are irrelevant to MIT students. We will need a method of confirming if they are a member of the MIT community.

   Mitigation: We will use only @mit.edu emails for users. We will use email confirmation to make sure that the email exists and is active.

4. **Malicious Groups**

   Users may make groups that are not approved by the MIT community, so we need a way to filter groups from being created. In addition, groups that are already created may maliciously post content onto the website, so we need to regulate those groups from doing so again.

   Mitigation: Site admins must approve of a group in order for it to be created. In addition, site admins have full control of the application, so if they choose so, they can ban a group by deleting it.

5. **Malicious Posts**

   Similarly, groups may post malicious posts or display information in a negative manner, so these posts must be moderated.

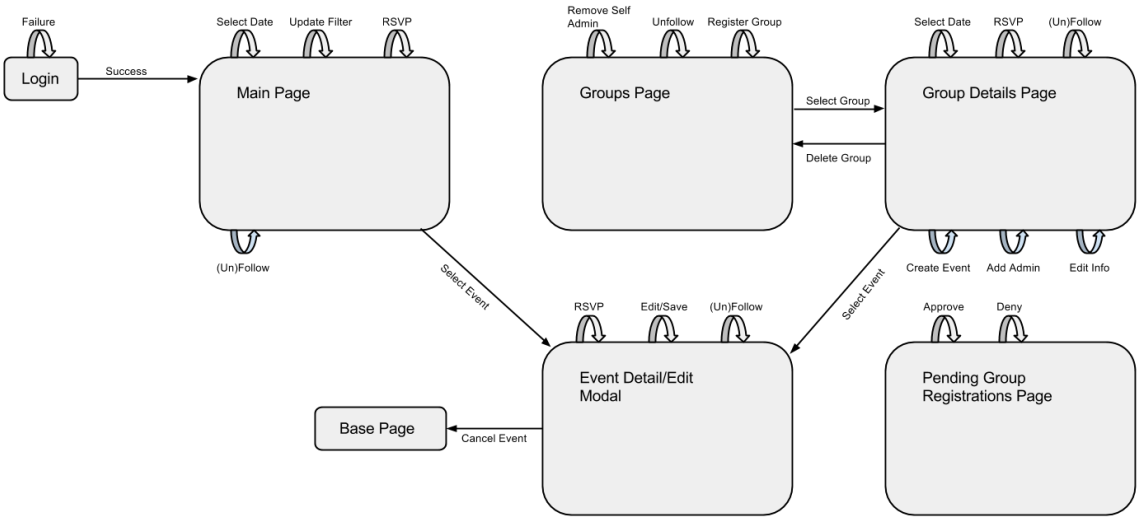
Mitigation: Site admins have full control, so they can edit the events. In addition, as stated before, if the situation gets out of hand, they have the right to ban and delete the group.

User interface

Wireframes

(See UI_Wireframes.pdf)

Interactions Diagram



Design challenges

Do we want to moderate post pre-publication or post-publication?

Post-publication, to facilitate smaller submission-to-publication latency; and it's easier and just as effective to warn and/or ban group accounts after questionable posts are publicized as it is to filter posts before they become public.

Do we want to implement a separate my-followed-groups feed page in addition to the main feed page, or should "My Followed Groups" just be a filter option in the main feed?

We could not find enough differences in the functionality and layout of a personal feed vs. a filtered global feed, so we will make "My Events" a filter option right on the main feed page, rather than having it be a separate page. This also simplifies our UI in the two cases of a logged-in vs. a non-logged-in user; a logged in user

would just have more filter options on the main page, instead of a whole new page available to them.

Should RSVP identities be anonymous or public?

The utility of a public identity is that your friends may be more likely to go if they see you're going; the utility of an anonymous identity is that users who may not want people to know that they're attending an event will be more likely to sign up. Thus, we will have a selector from which a user can choose to RSVP anonymously or publicly, so both cases are covered.

Is there a difference between filters and following? (Can following a group be implemented as saving a set of filters in which that group is selected?)

This depends on whether we think there is a use case for filtering by groups on the main feed at all - and we think there is not, or at least not enough. It does not feel like a natural action to filter by multiple groups in the same way that one might filter by multiple event types or locations, and filtering by one group can be achieved by going to that group's page. So we will keep separate the notions of filtering and following (following is not simply a saved set of filters).

Where should a user be able to RSVP? (Should each event summary box have a dynamic "RSVP" option, or should that only exist on the event details page?)

The headline, type, host and time of an event is usually enough for many people to decide whether they want to attend - so we should make RSVP available anywhere that information is present, including event summary boxes.

How should we organize the main feed?

1. We only want to display current or future events - perhaps allow groups to go through their past events for inspiration, but that is not important to the general public.
2. It should be chronological by event date (the more distant in the future it is, the further down it is) - so not like a future-fied version of Facebook/Twitter/email, and posting date does not matter.
3. The feed should be sectioned by day, with a calendar selector on the side for quick navigation.
4. We want each group to have an easily-accessible page with a description and a list of their upcoming events. This page should be accessible from a global groups page which lists all registered groups, so groups can be known and reached even if they have not yet hosted an event.

5. We do not want any free-input search. The only searchable information we have are in fully-defined, closed lists (tags, groups, locations), so filters and selectors make more sense than open-ended search boxes. This also erases a potential injection attack area.

Do we want to separate event text and image in the display? Or do we want to superimpose the text on the image, with fancy blending and/or fading effects?

We want to keep them separate, since we won't know which part of the image is less important and thus more sensible to fade/blend text onto. It is also much easier for us to implement.

What about events which are hosted by multiple groups?

While this is a legitimate concern (Nightmarket, for example, is hosted by >5 Asian student groups), we feel that the low frequency with which this concern arises is not sufficient to justify the design and implementation costs. Most events are hosted by one group, and it is not unreasonable to ask multi-host events to designate a representative group.

Do we want users to have self-selected usernames? Or Kerberos ID or email address?

Since our app is only designed to be used by the MIT student community, Kerberos ID or MIT email makes sense; and it gives us an existing guarantee of uniqueness.

Between Kerberos ID and email, we decided to go with MIT email. We can easily extract the kerberos portion from it, and entering email feels much more natural than entering Kerberos ID (from experience) - people often are confused by statements such as "Enter your Kerberos name", especially if it not accompanied by the statement "without @mit.edu"; whereas "Enter your MIT email address" is basically foolproof.

To enable personalized greetings and such (so users aren't referred to by their impersonal IDs), we will also ask for users' first and last names on registration.