Design Journal

The goals of this project were to create a site with a strong content moderation system that allowed the River Legacy Living Science Center to build a meaningful place for their community members to connect with one another. The site places emphasis on moderation principals and makes is simple for admins to remove posts as needed.

Parts of the site are designed with flexbox and a coded grid system to simply the design of the site and make it look good. While the focus of this mock-up was the functionality of the forum the new home page was simplified and modernized to provide a pleasant user experience and the forum itself is created to be simple yet elegant. It is intended for majority desktop users and as such may not be as user friendly to those on mobile.

The forum is made up of three database tables with a few proposed tables that would support features that were not implemented.

Implemented tables:

The Users table contains all of the accounts that belong to people who use the site. Columns on this table include: UID, UserName, Password, FirstName, LastName, EmailAdress, StreetAdress, City, State, Zip, UserImageURL, Admin, TrustedUser, ParentID, ForumAcess, and RejectedPosts. Users who are either admins or trusted users may post without needing approval. However if a trusted user's post is removed their status will change to reflect that. Rejected posts measures the number of posts a user has that have been rejected or removed. Parent ID connects the account of children to their parents. User images and multiple forums were not implemented.

The threads table serves to classify posts. All posts on a thread should relate to the thread title. The columns include: TID, Title, PostDate, FPID, and FID. FPID denotes the first post on the thread. The FID would contain the forum it was posted on if the project had multiple forums.

The posts table contains the majority of the information posted on the site. The threads table includes the columns: PID, postDate, UID, TID, ApprovalStatus, Rejected, and Content. TID denotes the thread this post belongs to and UID shows the user who posted it. A thread us understood to have been made by the same user as the first post (contained in FPID). ApprovalStatus is automatically set to false if the user is not a trusted user or an admin. If ApprovalStatus is false the post will not show up on the forum. Rejected is automatically set to false and becomes true if the post is rejected (never goes up on the site) or removed (was up on the site and was removed).

Proposed Tables:

Forums: Contains all the different forums with a key identifier of FID that is linked to threads.

Images: contains images linked to individual posts. This way the user can add a non-specific number of images to their posts (although the number would be limited)

Removed posts: This would include copies of removed are rejected posts. When a post is removed or rejected the user has a chance to edit it for reapproval and ApprovalStatus and Rejected both get reset to false. This table would keep a record of pre-edited offensive posts for reference in case of a dispute.

The forum has two navigation menus based on the user's admin status or lack thereof. Although all of the pages that are available to non-admins are available to admins as well the information on or functionality of the page will be dependent on the user's status.

The only pages for which this is not true are the edit user pages. On those the admins are sent to a form with more inputs for things such as trusted user, admin and other controlled information. The admin version of this page also contains a list of all users for easy reference when looking for UID to put into the parent ID input.

Below is a truth table inspired depiction of potential permutations meant to show the various possible outcomes when a user views a thread. There is a similar process implemented on the thread list page. Not shown in the graphic is the choice to remove the delete and remove buttons from the first post on a thread. In order to delete this the user must delete the entire thread. Similarly, if an admin wants to remove the first post of a thread, they must remove the whole thing.

	TUID == UID	Rejected	Post Approved	Admin	Outcome:
1	T	T	T	T	Sees post, Removed notice w/ was approved, has option to edit, delete or approve
2	T	T	T	F	"" W/o option to approve
3	T	T	F	T	Post and removed notice, option to edit, delete or approve
4	T	T	F	F	rejected notice w/ option to edit or delete
5	T	F	T	T	Sees post, option to edit, delete or remove/ unapprove
6	T	F	T	F	sees post with option to edit/ delete
7	T	F	F	T	pending post by you w/ option to approve option edit/ delete
8	T	F	F	F	Pending post by you option to edit/ delete
9	F	T	T	T	See post/ removed msg/ option to approve
10	F	T	T	F	Post removed msg
11	F	T	F	T	Post rejected/ option to approve
12	F	T	F	F	Nothing
13	F	F	T	T	post w option to remove
14	F	F	T	F	Post

TUID==UID means that this user is the one who made the post. The other columns are explained in the implemented tables section. One key takeaway from this table is that if a post is approved and removed all users get the message that a post was removed from the thread. If a post has not been approved or rejected it can be approved or rejected. If it has been approved it can be removed and if it has been removed an admin can restore it.

The login works via sessions, and provides the user with messages to let them know if they put in the wrong password or have other similar issues. When and admin logs in they are redirected to the pending approval page where all unapproved posts are listed following their thread title. The admin can click to view the thread and approve from there if they prefer. When a non-admin user logs in they go to the approved thread list page. This acts as their home page.

Many of the pages that approve or add entries to the database are only used for a moment and then redirect to the users last page. This allows the user to see their newly made post on the site for validation and prevents duplicate entries into the database as the header redirect clears the post array.

The forum also blocks users from editing pages they don't have access to via query string.

Proposed features that were not implemented:
- Ensure that usernames are not already in use.
- Add a list for admins to view a person's rejected posts
- If a post is restored the rejected posts de-increments
- Add a link or button to view users admin page to make users trusted