Technical info

* The website is made with html/CSS/JavaScript and PHP.
* It is a simple textboard.
* It is hosted on a server at school PC-64 running Debian 11.
* I work completely remotely by directly SSHing into the server.

**Completed**

* Front-end is pretty much done.
* Hosted on a dedicated server computer.

**TO-DO**

* Database and saving of information
  + First there needs to be boilerplate html for the thread.
    - Somehow be able to collect the boilerplate, populate it with user info then save it.
      * Echo this back to show on the site
  + Try and do something like this for now and figure a better structure later.

**Functionality it should have.**

* When the user creates a thread, it should dynamically create a new html document with the thread and relevant elements.
  + The thread should appear on the page it was made on. E.g. (Forum.com/tech/” thread”)
    - Routing, not sure how I’m going to do this can be done via front-end or back-end. (React, Ruby)
  + The reply field is connected to the thread so it only updates the page it’s on.
    - Asynchronously update the page without reloading. Planning on doing this via Ajax. jQuery incorporates Ajax so I’m thinking of doing it with jQuery.
  + Threads will expire over time.
    - Add “bump” functionality to threads.
      * Threads on slower boards expire slower and vice versa.
* Admin threads that never expire.
* A counter that increases every time that a post has been made on a board to determine the post number for a post.
* A link to the thread should appear on the board it was created on.
* Information about a thread will be shown on the bottom (number of posts, maybe number of unique posters)
* Posts have a unique identifier.
* Threads are sorted by bump order.
* “Sticky” functionality
  + An admin can “sticky” a thread. This means that the sticked thread shows first on the list of threads and has a unique signifier (maybe an icon)
* Moderation
  + Create a mod user that can delete posts and threads.
    - Eventually be able to ban them via IP and maybe even ban via browser user agent.
    - Mods will only be able to remove posts and not ban anyone.
    - Removal of posts and threads will be done via jQuery.
  + Admin user that can ban people.
* Backend
  + Currently have absolutely no clue on how I’m going to handle the back end.
  + For now, every board is its own table, the entire html page is saved and is going to be echoed back via PHP.
  + In the future every thread will have metadata saved (number of posts, unique posters etc).
    - Every post will have metadata saved (the IP behind the post).
      * This will help with moderation.
    - The number of unique posters will also be saved and maybe even shown on the bottom of thread.
  + Will switch from PHP to something for back-end. PHP is old and modern and even better solutions exist. Heavily considering Ruby but NodeJS is also an option.

**Note**

I use post and comment interchangeably.

I use poster and user interchangeably.

A unique poster is a new person, this can be determined by IP. Everytime a new IP makes a new post