# Intentions:

**CodeTerra Website:**

A Project by: Avi Varma & Sam Peter

This project aims to build an online platform that will facilitate developers to create and engage in online discussions about coding and other technology, similar to forums such as Reddit and Hacker News but with a focus on this specific niche of the internet.

The CodeTerra website will allow registered users to engage in online discussions and unregistered users to view discussions similar to forums such as Reddit and Hacker News. The goal is to produce a similar type of service that allows users to register, post stories and make comments on items. Additionally, unregistered users must be able to view the content but will not be able to edit or comment on posts.

# Baseline Objectives:

## Website user’s objectives:

Browse discussions without registering

Search for posts by keyword without registering

Search for items/posts by keyword without registering

Register at the site by providing their name, e-mail and image

Allow user login by providing user id and password

Create and comment on discussions when logged into the site

Users are required to be able to view/edit their profile

User password recovery (via email)

**Website administrator’s objectives:**

Search for user by name, email or post

Enable/disable users

Edit/remove items and comments

## Minimum Functional Requirements:

Hand-styled layout with contextual menus (i.e. when user has logged on to site, menus reflect change). Layout frameworks are not permitted other than Bootstrap (see above). 2 or 3 column layout using appropriate design principles (i.e. highlighting nav links when hovered over, etc) responsive design

Form validation with JavaScript

Server-side scripting with PHP

Data storage in MySQL

Appropriate security for data

Site must maintain state (user state being logged on, etc)

Responsive design philosophy (minimum requirements for different non-mobile display sizes)

AJAX (or similar) utilization for asynchronous updates (meaning that if a discussion thread is updated, another user who is viewing the same thread will not have to refresh the page to see the update)

User images (thumbnail) and profile stored in database

Simple discussion (topics) grouping and display

Navigation breadcrumb strategy (i.e. user can determine where they are in threads)

Error handling (bad navigation)

Additional Requirements:

Search and analysis for items

Hot threads/hot item tracking

Visual display of updates, etc (site usage charts, etc)

Activity by date

Tracking (including utilizing tracking API or your own with visualization tools)

Collapsible items/treads without page reloading

Alerts on page changes

Admin view reports on usage (with filtering)

Styling flourishes

Responsive layout for mobile

Tracking comment history from a user’s perspective

Accessibility

Access a shared coding environment with other users

Code Friendly Comment Threads

# Pre-Planning Design Considerations:

**Colour Scheme:**

In order to facilitate our design process, we decided to generate a colour scheme for use on our pages. Although this may not represent our final product, it allowed us to have some compatible colours for use within our design process.

Chart, treemap chart

Description automatically generated

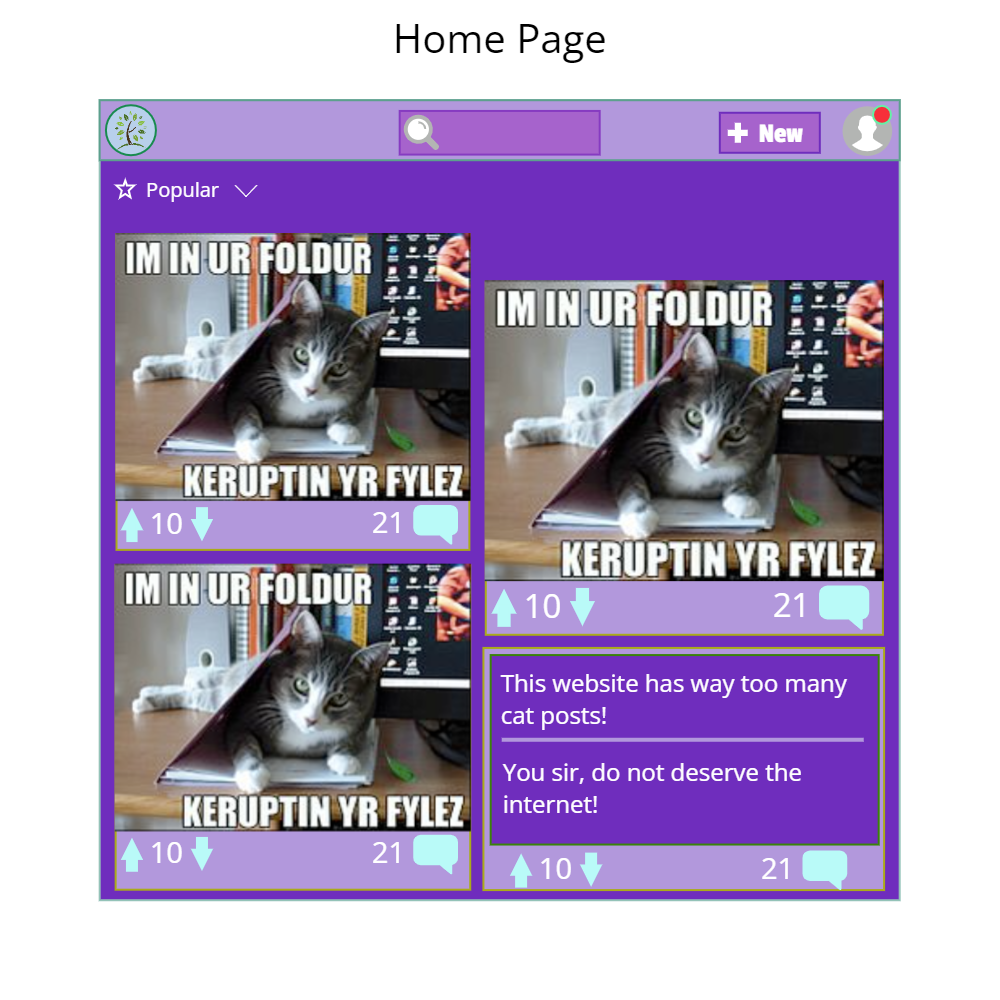
The first three colours feature prominently on our home pages and core content, while we mostly utilize the last two to highlight important content and buttons.

# Main Page

This is the main page of the website, and the first thing users see. This page can be navigated to from any page on the website by clicking the logo at the top left of every page. This page displays the various threads that have been posted to the site, sorted either by most recent, most popular, or trending (hot). Any user that is logged in may vote on any of these threads from this page.

From here users may navigate to any of the following pages:

* Any **thread** shown on screen, by clicking on the given thread.
* The **login** page if they are not logged in, by clicking the blank profile picture in the top right, or by clicking the **new post** button.
* Their **profile** page if they are logged in, by clicking their profile picture in the top right, and can be clicked to access more account details.
* The **search** bar at the top can be used to filter content on the site, and search through existing posts.
* The **new post** page if they are logged in, by clicking the new post button at the top of the screen.



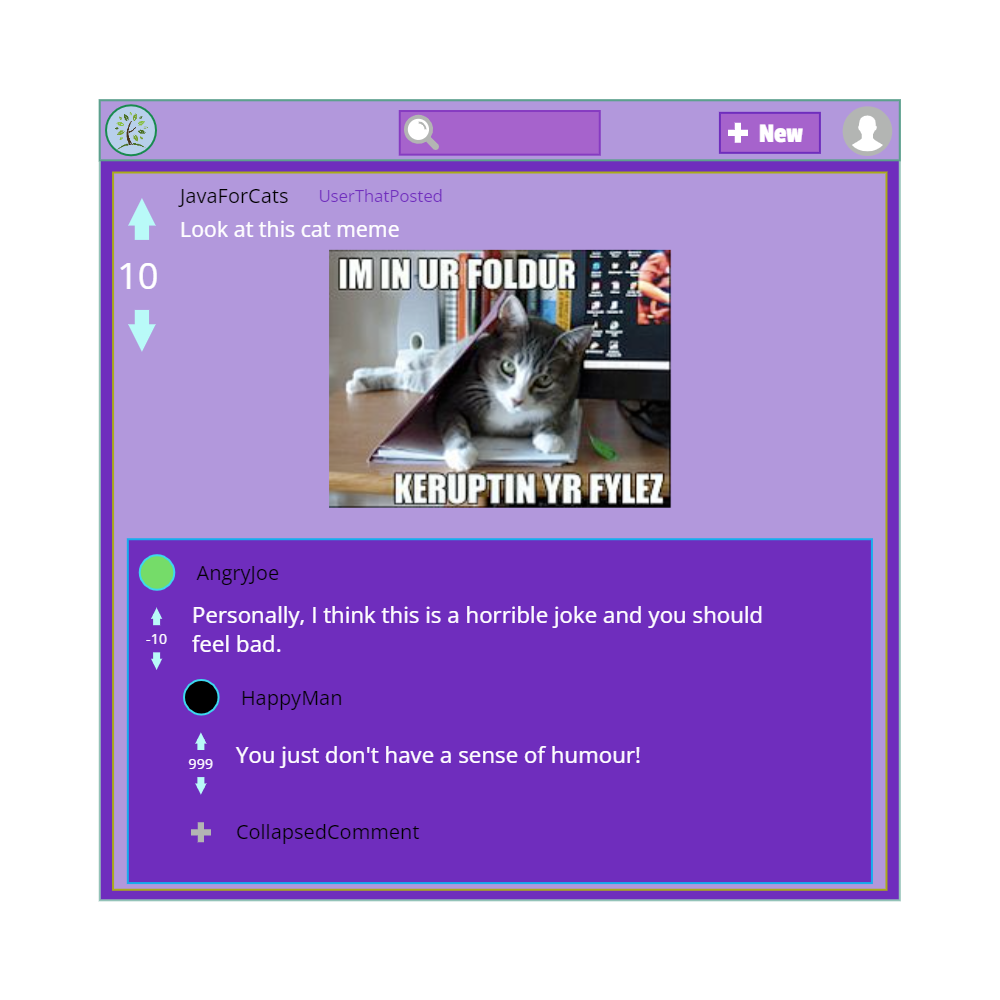
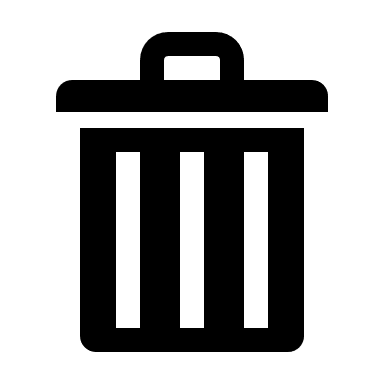
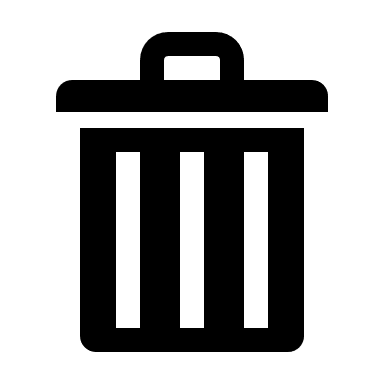
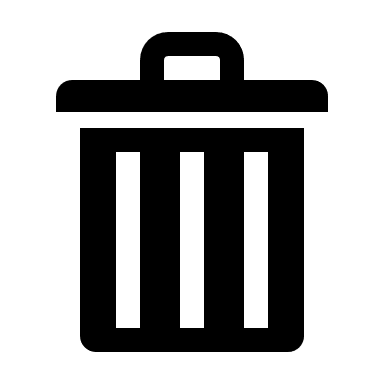
# Threads

Each thread on the website has its own page, where any user can see any comments that have been made on it, **sorted by popularity, or date**. Users may also collapse any of the comments, along with all their sub comments. Any user who is logged in may also vote on any of the comments that have been made, or the post itself, and make their own comments directly under the post or in response to anyone else's comments.

Admins should be able to moderate the posted content and comments associated with it via this page. They should be able to delete comments, and users should be able to report them as well. The user should also be able to delete their own post.

From here users may navigate to any of the following pages:

* The **main** page, by clicking the logo at the top left
* The **login** page if they are not logged in, by clicking the blank profile picture in the top right, or by clicking the **new post** button
* Their **profile** page if they are logged in, by clicking their profile picture in the top right
* The **profile** page of the user that posted the thread, or any user that has posted a comment on screen, by clicking on their username
* The **search** page, by typing anything into the search bar in the top right and clicking the magnifying glass or hitting enter
* The **new post** page if they are logged in, by clicking the new post button at the top of the screen



# Login

The login page allows any user to log into an account they have already made for the site, or to register a new account. After either of these is done, the user returns to the **main** page, now logged in. From here users may navigate to any of the following pages:

* The **main** page, by clicking the logo at the top left, or the “x” button in the top right corner of the prompt box.

Graphical user interface, application, Teams

Description automatically generated

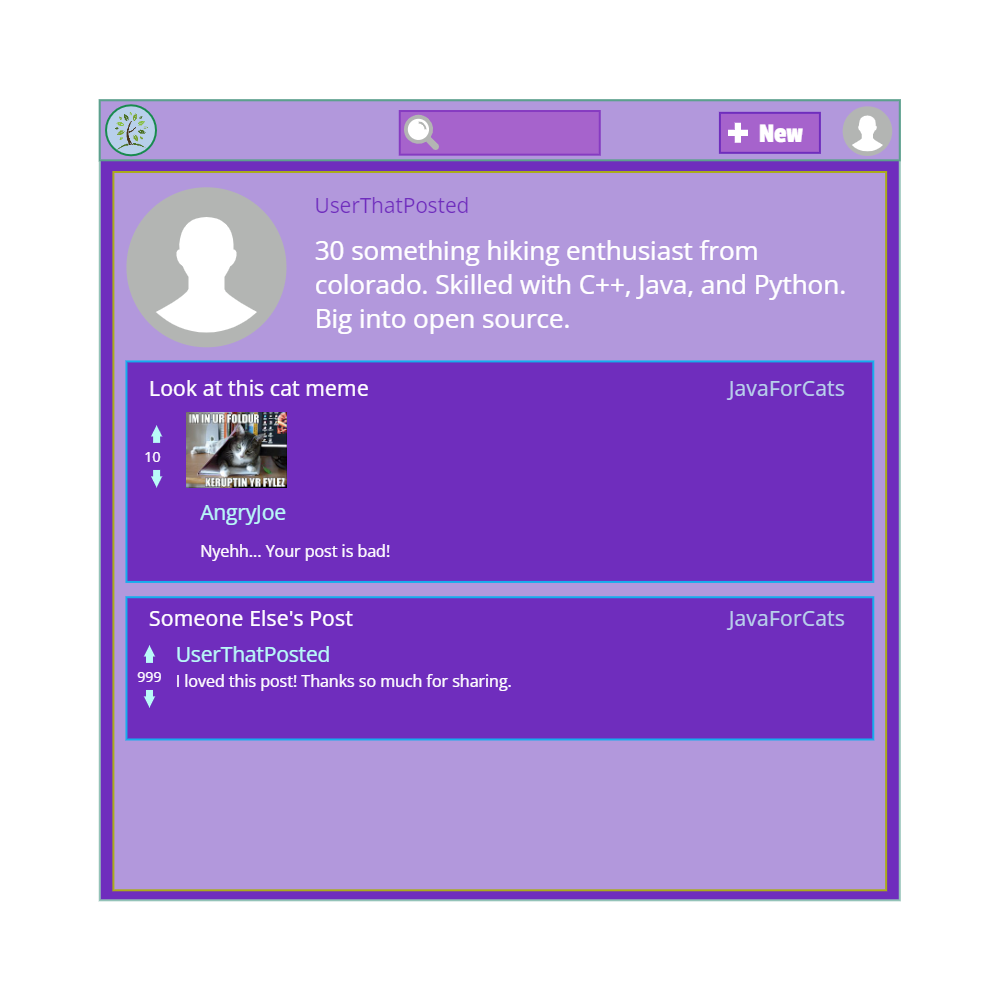
## Client Side Validation/Security Considerations:

* The users should not submit login information in an unaltered format. Ideally we should have some sort of hashing algorithm between password entry in the form, and submission to the server for processing/storage.
* Users should not be able to log in with missing information in the field. (i.e. they should not be able to submit the form with a blank password or username.)

# Profile

The profile page of a user displays that user's username, profile picture, and bio, along with any posts or comments they may have made, sorted by most recent. If it is the profile page of the user that is currently logged in, that user may edit the profile picture and bio from here. From here users may navigate to any of the following pages:

* The **main** page, by clicking the logo at the top left
* Any **thread** the user has posted or participated in, by clicking on the thread.
* The **login** page if they are not logged in, by clicking the blank profile picture in the top right, or by clicking the **new post** button
* The **search** page, by typing anything into the search bar in the top right and clicking the magnifying glass or hitting enter
* The **new post** page if they are logged in, by clicking the new post button at the top of the screen



## Profile Box:

Users should also be able to access some information about their profile from the profile overlay, which should be accessible from any page. You should be able to access some subpages from here, and your main profile page.

A screenshot of a computer

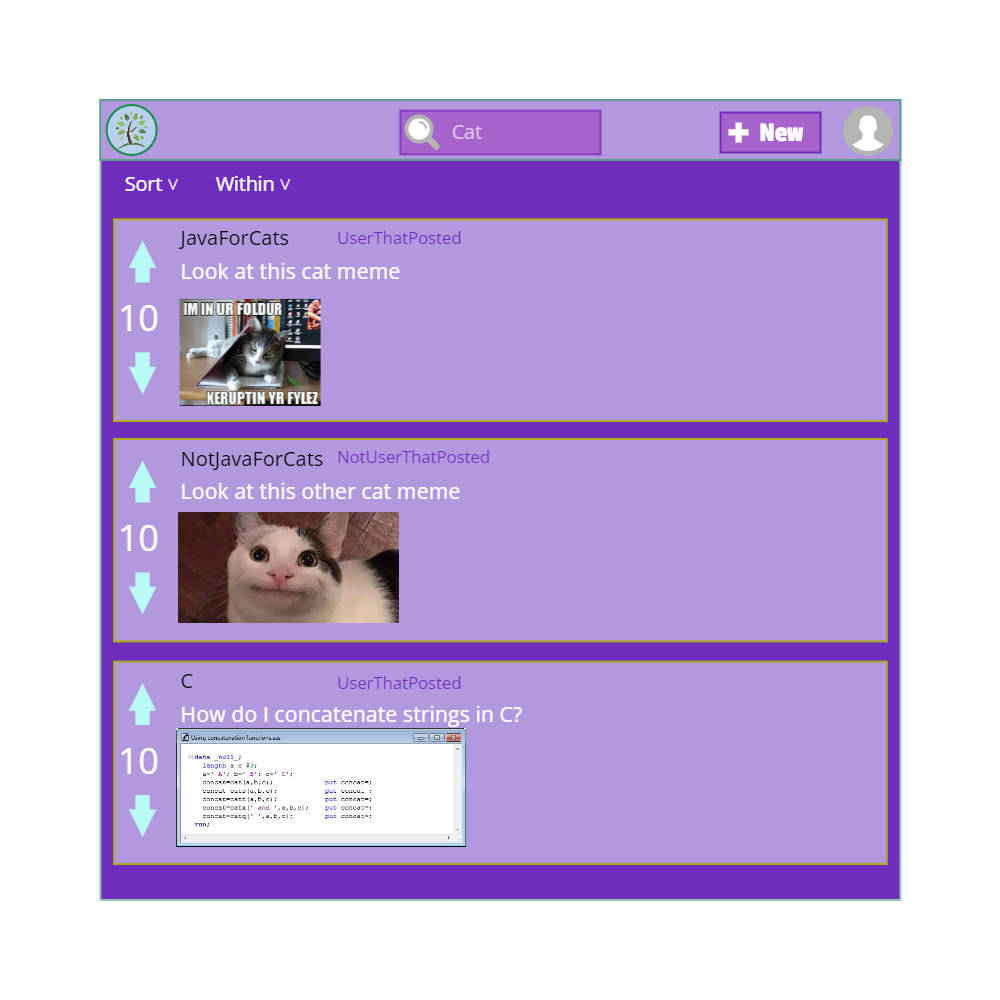
Description automatically generated with medium confidence

# Search

The search page displays threads sorted by relevance to the search terms. Users should also be able to select their sort method, by either most popular, most recent, or most commented on. Users should also be able to pick the time span results are posted within.

From here users may navigate to any of the following pages:

* Any **thread** shown on screen, by clicking on the given thread
* The **login** page if they are not logged in, by clicking the blank profile picture in the top right, or by clicking the **new post** button
* Their **profile** page if they are logged in, by clicking their profile picture in the top right
* The **search** page, by typing anything into the search bar in the top right and clicking the magnifying glass or hitting enter
* The **new post** page if they are logged in, by clicking the new post button at the top of the screen



# New Post

The new post page allows users that are logged in to create a new thread, by posting either a text post or an image. If a user would navigate to this page but is not logged in, it instead directs them to the **login** page. From here users may navigate to any of the following pages:

* The **main** page, by clicking the logo at the top left
* Their **profile** page, by clicking their profile picture in the top right
* The **search** page, by typing anything into the search bar in the top right and clicking the magnifying glass or hitting enter

## Form Validation:

Users should not be able to submit the form without correctly selected information, such as no selected group or no title.



# Admin/Moderator Hub

A page that we do not have a finalized design for, but plan to implement is the moderation hub. This page should feature:

Visual display of updates, etc. (site usage charts, etc.)

Activity by date

Tracking (including utilizing tracking API or your own with visualization tools)

Collapsible items/treads without page reloading

Alerts on page changes