

Group members:

Kenneth Ou (oukennet@usc.edu)
Luke Reshwan (reshwan@usc.edu)
Rbhu Gandhi (rbhugand@usc.edu)
Kevin Sangmuah (sangmuah@usc.edu)
Zain Merchant (zsmërcha@usc.edu)
Henry Zhu (liao zhu@usc.edu)
Tammy Truong (thtruong@usc.edu)

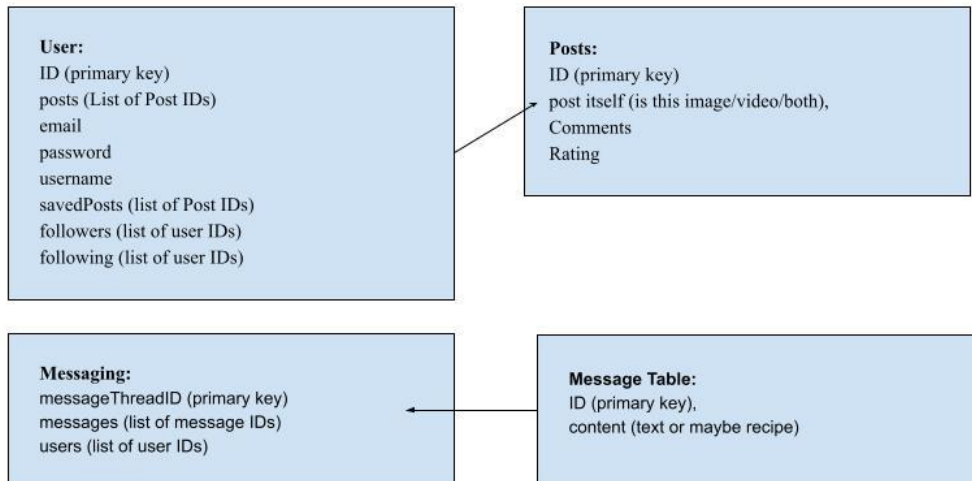
Detailed Design

Hardware/Software Requirements

- Frontend: React
- Backend: Java
- Database: SQL

- Internet connection: required
- Up to date web browser (Chrome recommended)
 - Chrome System Requirements
 - Windows
 - Windows 7 or later
 - An Intel Pentium 4 processor or later that's SSE3 capable
 - Mac – macOS High Sierra 10.13 or later
- Memory: 4 GB (recommended)

Database Schema

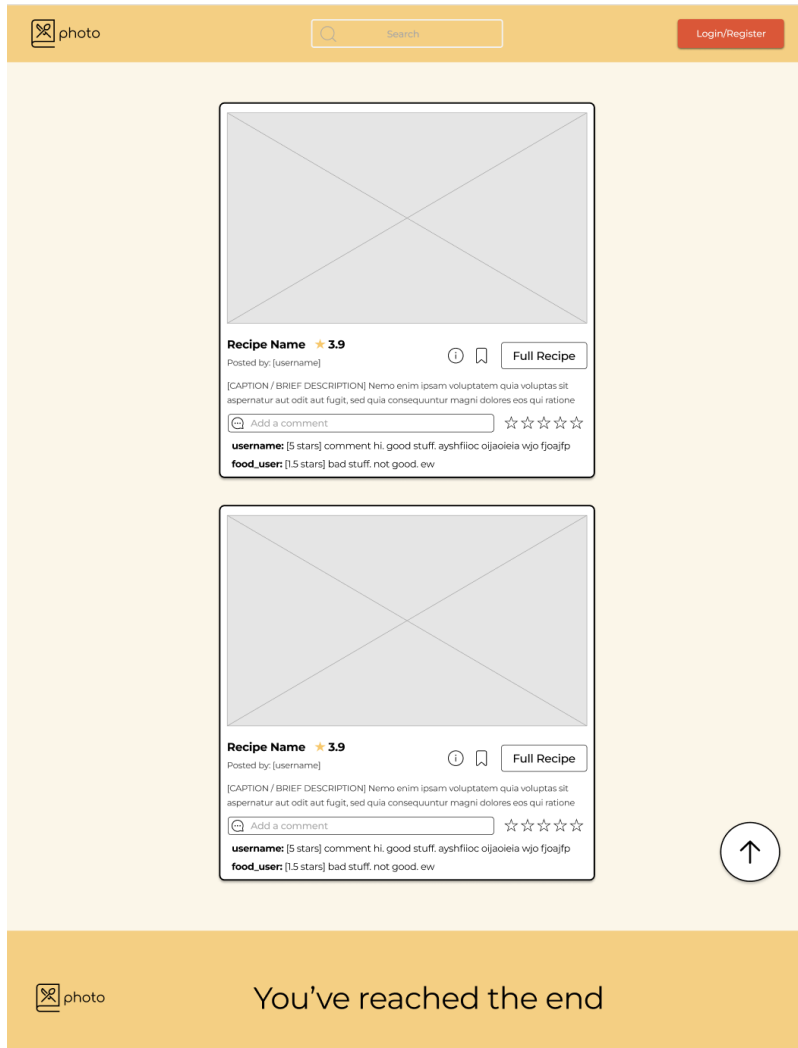


Design and Website Functionality

[figma designs](#) – color hex codes are include in the figma page

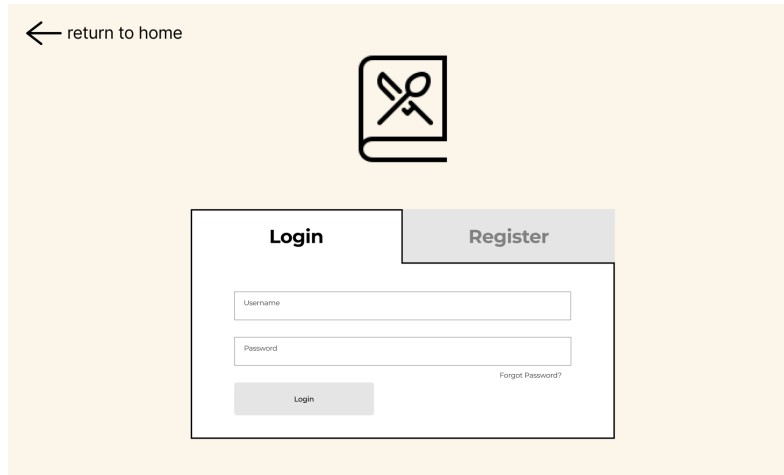
user opens the website

- users are shown the home/explore page with the **unregistered user navigation bar**




user can login

- user clicks the “login/register” button on the home page which takes them to the login page
- user must enter a valid username and password
 - if the username or password is not valid, reject it and show an error message
- user hits the “login” button
- user is redirected to website home page and is now signed in
 - home page will now have the registered user nav bar which has their account profile, messages, and create new post functionality



← return to home



Login **Register**

Username

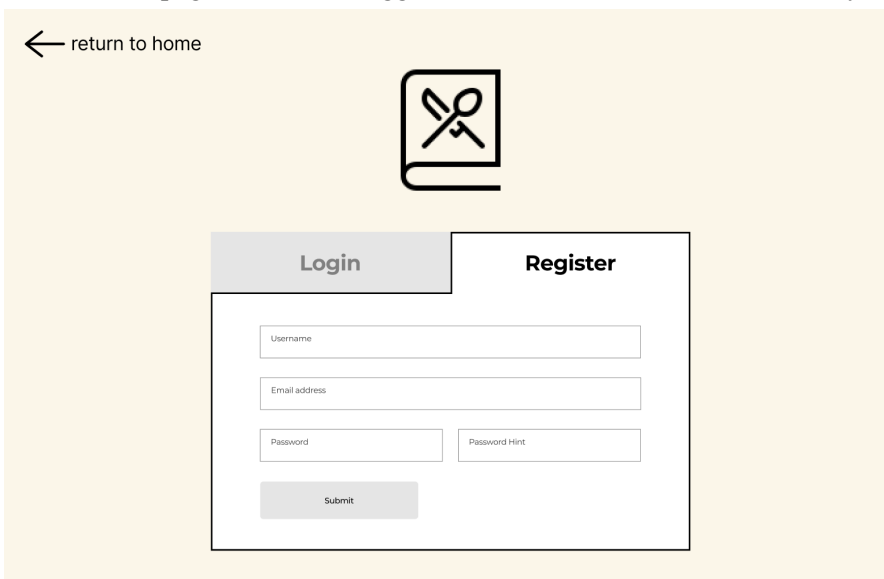
Password

Forgot Password?


Login

user can sign up

- user clicks the “login/register” button on the home page and clicks the “register” tab
- user must enter a username, email, password, and password hint
 - if username is already used, show error message and redirect user to login screen if they want
 - if email is not in a valid format, reject it and show error message
 - if email is already used, redirect user to the login screen
 - if password is less than criteria, reject it and show error message
- user hits the submit button
- user is directed to home page and is now logged in, with the account stored in the system



← return to home



Login **Register**

Username

Email address

Password

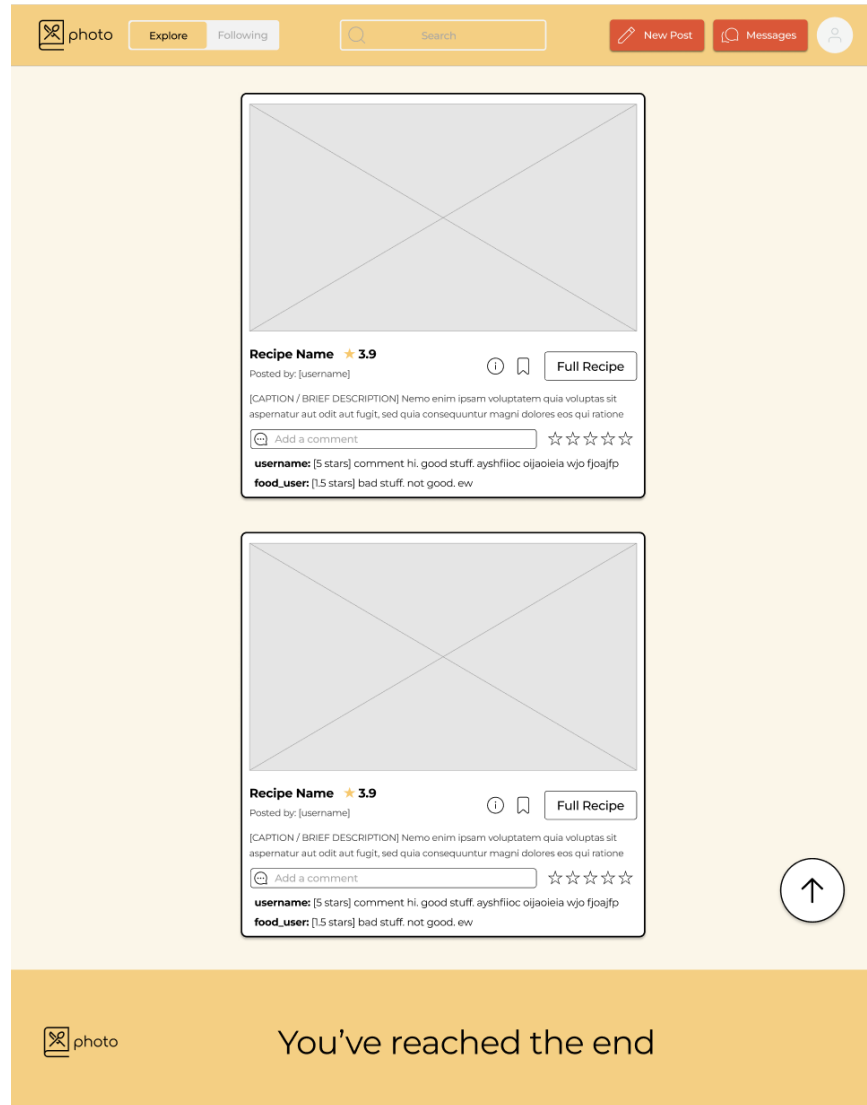
Password Hint

Submit

user can view explore (home) page

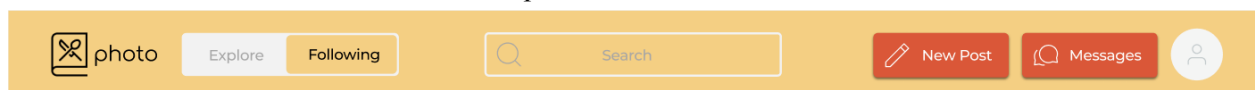
- users can scroll through a feed with the most popular posts at the top
- user can view and comment on posts
 - if they are not logged in, show message and redirect them to the login page

- user can rate any post
 - if they are not logged in, show message and redirect them to the login page
- user can view the full recipe if there is a pdf or separate website linked
- if user reaches the bottom of the feed, they will see footer that tells them they reached the end and a button that redirects them to the top



users can view their following feed

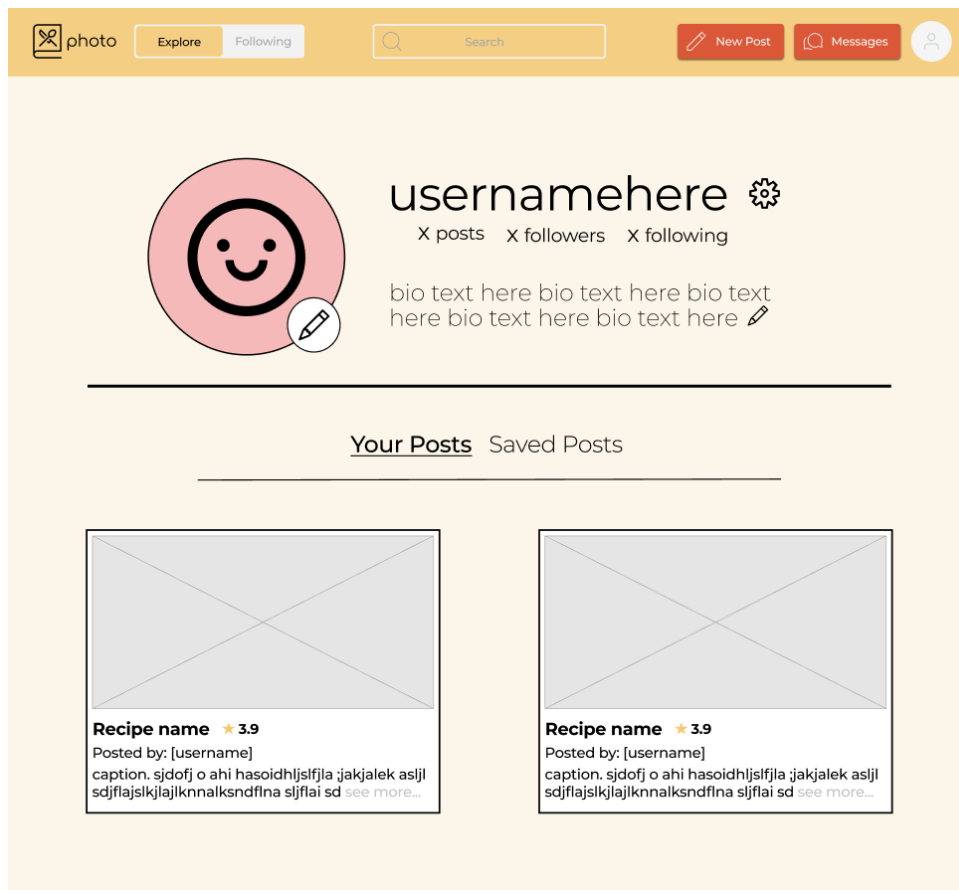
- this feed is only available for users that are logged in
- can get here by clicking the explore / following button on their navigation bar
- user will see posts only from people they follow
- user can comment, rate, and view full recipes
- if user reaches the bottom of the feed, they will see footer that tells them they reached the end and a button that redirects them to the top



NOTE: frontend design is the same as the home/explore page just with a different navigation bar and feed.

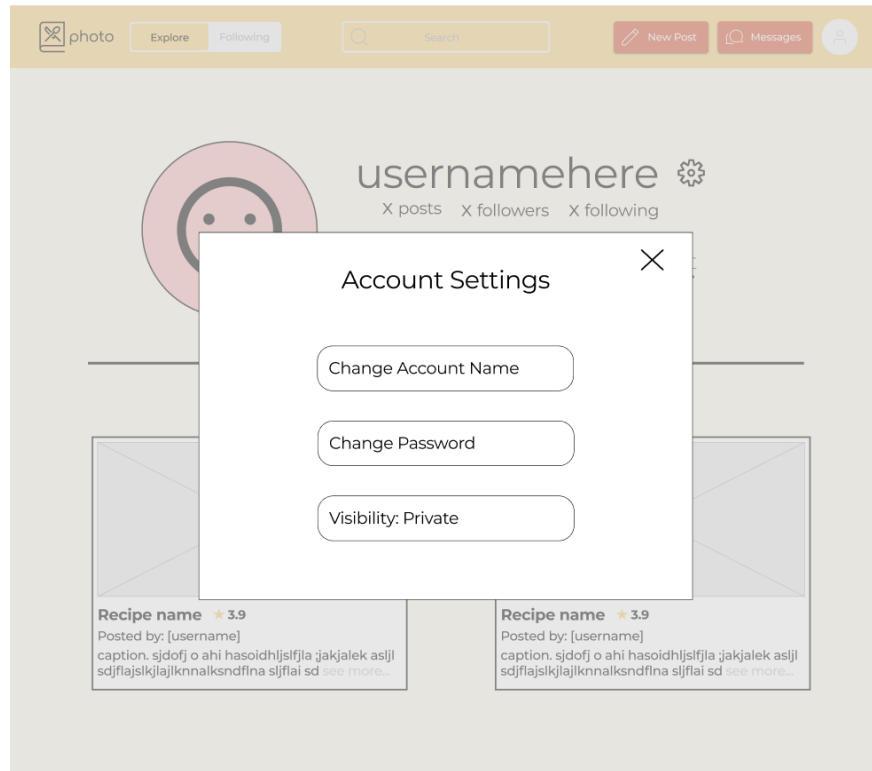
user can view their account profile

- user must be logged in and click their account button in the nav bar
- page will display the user's account photo, username, bio, number of people they follow and number of people who follow them.
- user will see all their posts
- user can change their account settings by clicking the “settings” button
- user can see their saved posts by clicking the “saved” button



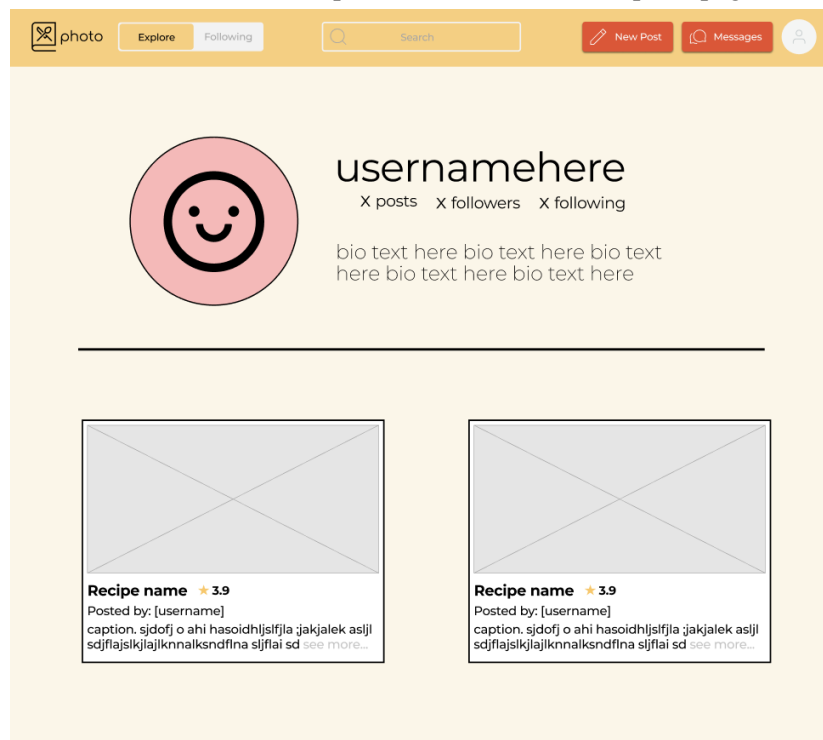
user can update their account settings

- user can navigate to their account settings by clicking their user profile and the settings button
- settings page will popup
- user can change their account name
- user can change their password
- user can change their visibility to public or private



user can see other user's profiles

- user can click on another username and see their profile
- design will look the same as the user profile without the saved posts page and the setting button

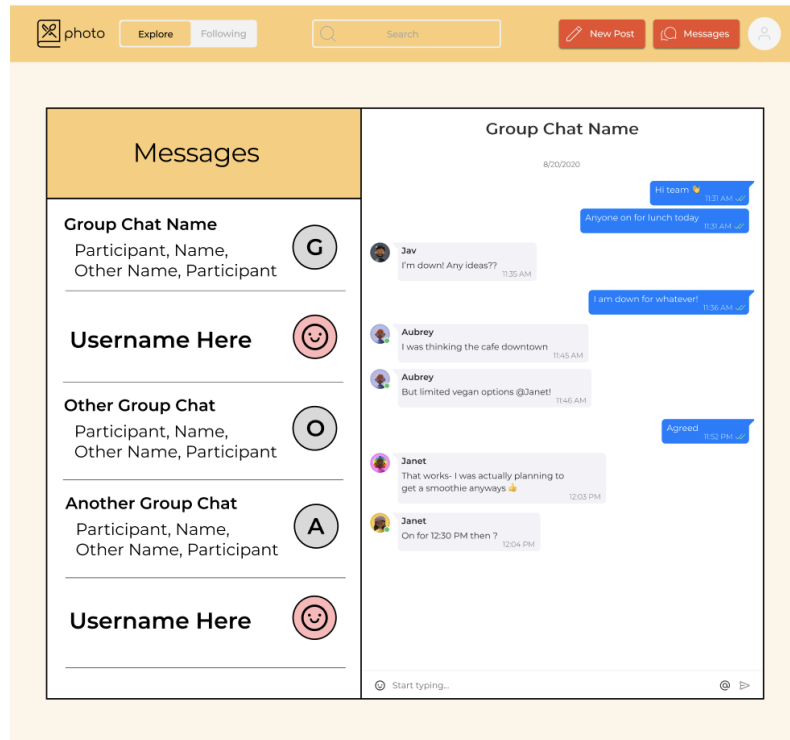


users can view all their messages

- user clicks the messages button in the navigation bar
 - only applicable for users that are signed in
- user can view all their group messages and direct messages

users can chat with another user or with group

- user can click on an individual chat room and message with the participants

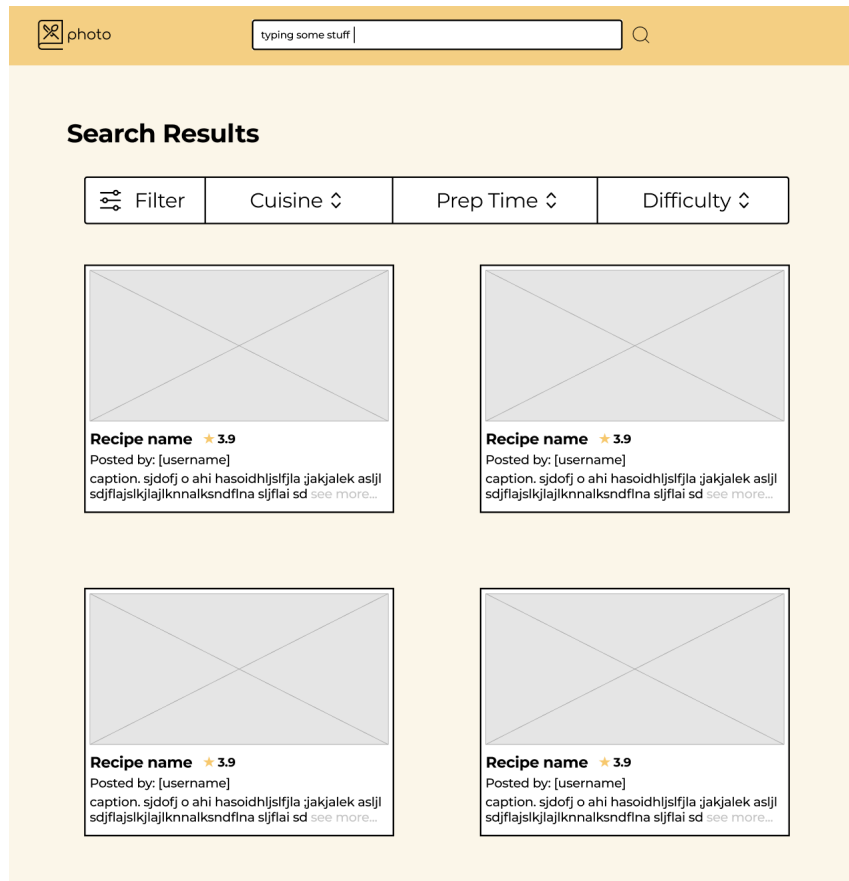


users can search for a post

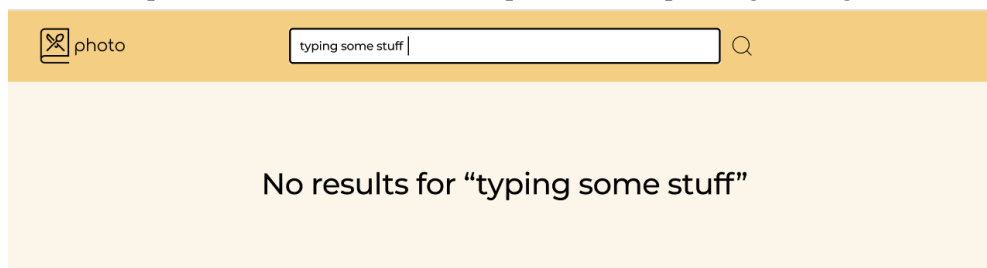
- user clicks the search bar and is redirected to the search page
- user can search by categories like cuisine, prep time, difficulty level, etc.



- user can type any keyword(s) into the search bar and press enter
 - if there are relevant posts, display them in a list and allow users to filter their search
 - if no relevant posts, display message



- if there are no posts with the search term, output the corresponding message

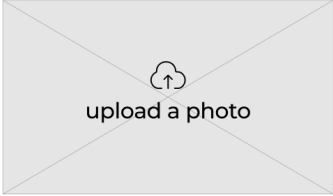


users can create a new post

- user can create new post by clicking the new post button
- user can upload a photo, set a name, caption, and additional information.
 - required fields: recipe name, caption, estimated prep time, difficulty, cuisine, some form of recipe (pdf or link)
- user can paste a link to a webpage with the recipe or upload a recipe in a pdf format
- user clicks on post to post their recipe

← return to home

New Post



Recipe Name


Caption

More Information


Estimated Prep Time

Difficulty

Cuisine

upload recipe (pdf) 

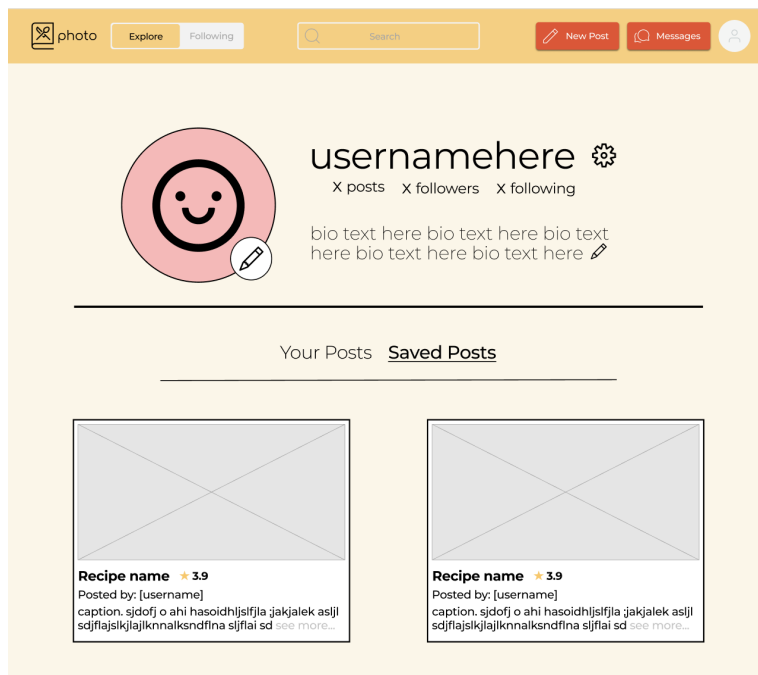
Recipe Link

 Post

or

users can view their saved posts

- user must click their account profile and navigate to the saved posts button
- users can see all the saved posts and is able to click on them for more information and recipes



user can view any individual post

- user can click on any post's image or name and be directed to the individual post page
- user can view all the comments, full description, link to recipe, and more



Algorithms

```
quicksort (array){  
  if (array.length > 1){  
    choose a pivot;  
    while (there are items left in array){  
      if (item < pivot)  
        put item into subarray1;  
      else  
        put item into subarray2;  
    }  
    quicksort(subarray1);  
    quicksort(subarray2);  
  }  
}
```

Linear Search

Algorithm pseudocode:

```
set found to false;  
set position to -1;  
set index to 0  
while index < number of elements and found is false  
  if list[index] == search value  
    found = true  
    position = index  
  end if  
  add 1 to index  
end while  
return position
```

Taken from:

https://www.researchgate.net/figure/Pseudocode-of-quicksort-adapted-from-1_fig1_332434596
<https://slideplayer.com/slide/7698898/>

class UserFunctions:
String username

class Search:
String searchString
String getSearchString()
void setSearchString()
Recipe[] findSimilarRecipes()
Account[] findSimilarUsers()

class HomeExplore:
Recipe[] followedRecipes
Recipe[] exploreRecipes
void displayHomeRecipes()
void displayExploreRecipes()

class recipe:
int viewCount
int ratingAverage
int amountOfRatings
Image recipePicture
String comments[]
int getViewCount()
int getRatingAverage()
int getAmountOfRatings()
Image getRecipePicture()
void setViewCount()
void setRatingAverage()
void setAmountOfRatings()
void setRecipePicture()

class LoginSignup:
String newUsername
String newPassword
String newPasswordHint
bool checkCredentials()
String getUsername()
String getPassword()
String getPasswordHint()
void setUsername()
void setPassword()
void setPasswordHint()

Class Account :
String username
String password
String passwordHint
Recipe[] userRecipes
String userBio
int amtFollowers
int amtFollowing
int amtPosts
Recipe[] savedRecipes
bool publicOrPrivate
void displaySaved()
void displayUserProfile()
String getUsername()
String getPassword()
String getPasswordHint()
Recipe[] getUserRecipes()
String getUserBio()
int getAmtFollowers()
int getAmtFollowing()
int getAmtPosts()
Recipe[] getSavedRecipes()
bool getPublicOrPrivate()
void setUsername()
void setPassword()
void setPasswordHint()
void setUserRecipes()
void setUserBio()
void setAmtFollowers()
void setAmtFollowing()
void setAmtPosts()
void setSavedRecipes()
void setPublicOrPrivate()