

Resource Wall

Pinterest for learners.

Allow learners to save learning resources like tutorials, blogs and videos in a central place that is publicly available to any user.

Requirements:

- users should be able to save an external URL along with a title and description
- users should be able to search for already-saved resources created by any user
- users should be able to categorize any resource under a topic
- users should be able to comment on any resource
- users should be able to rate any resource
- users should be able to like any resource
- users should be able to view all their own and all liked resources on one page ("My resources")
- users should be able to update their profile

User Stories

1. User Views posts
 - A User can view posts created
 - Post will display a title, a description, and a url
 - A User can click on a post to view the post
 - A User can like posts
 - A User can rate posts
 - A User can comment on posts
 - (stretch) Videos are embedded using youtube API iframe
 - (stretch) Article urls have a Preview
 - (stretch) main page displays posts based on users interests
2. User Creates and manages their Posts
 - a. A User can create a post with a title, description, url, and topic, *KIND*.
 - b. A User can edit their post
 - c. A User can delete their posts
 - d. A User can view a list of topics

3. User can view their posts and liked posts
 - a. A User can view their created posts
 - b. A User can view liked posts
4. User can search posts
 - a. A user can search posts using title, author (usernames), topic, (stretch) posts creation date
5. User can edit their profiles
 - a. A user can edit their profile information
 - b. A user can change their password

App Stack:

Javascript: Node, Express, EJS, cookie-session

Database: PostgreSQL

Possible packages for stretch ideas: moment(For Post creation dates), Youtube API (for embed youtube videos), Link-Preview-JS

ERD:

Users: id, user / email / password / profile image(url)

Post_likes: id, user_id, post_id

Posts(Learning_resources): id,USER_ID, title / url / description / created_at / updated_at / topic_id

Topics: id, topic_name

Post_Comments: id,post_id, commenter_id, comment_text, comment_date

Post_ratings: id,user_id,ratings,post_id

Each User has many liked posts

Each User has many posts

Posts can be liked many times

Each topic has many posts

Each post has many comments,

Each user has many comments for a single post

Each user can rate

Each post has many ratings(avg should be displayed as a total)

Routes

NOTE: Highlighted is not required, is stretch

Login Routes

GET

/login/:id - auto login

/login - display login form

/register - display create account form

/logout - clears session cookie and redirects to /login

POST

/login - checks login credentials, create session cookie, redirects to home page /

/register - creates user account entry in db, create session cookie, redirects to home page /

Posts/Resources Routes

GET

/ - Home page, display login button if not logged in, if logged in display posts

/users/:user_id - displays user profile page & displays user posts and liked posts(username button)

/users/new - displays create post form(create button)

/posts/:post_id - displays specific post

/search - Searches db and displays queried posts(search button)

/topics - topics button

POST

/user/:user_id - edits user information

/users/new - inserts new post entry in db, redirects to /posts/:post_id – (submit button in the create new post form)

/user/:post_id/update - updates post information

/user/:post_id/delete - deletes post

Comments, Likes, and Ratings

GET

/:post_id/comments - queries comments from DB to display on posts

/:post_id/ratings - queries and averages ratings and ratings for post to display on post

/:posts_id/likes - queries and sums likes for post to display on post

POST

/:post_id/comments - creates comment entry DB

/:post_id/ratings - creates a rating entry in DB

/:post_id/likes - creates a like entry in DB