Rally

STL Neighborhood Networking Application

Programmers:
Adam Sands
Alex Jove
James Griesedieck
Jami Defenbaugh
Nathan Coursey
Sam Adkins

What is Rally?

❖ Website tailored to neighborhoods with the STL metropolitan area

❖ Designed to fulfill the need for increased community connection

❖ Provides information, resources, services, opportunities, and connections with residents

❖ Includes:
  ○ community forum for messaging
  ○ events page to offer project ideas and to solicit volunteers;
  ○ restaurant listings in the community with reviews
  ○ services page listing offers or requests for freelance work or needed items
  ○ resources page that includes a table of important local information

Our website is deployed!   Check it out at https://home-5013300400.app-ionos.space/

Technologies Used

❖ Front End
  ○ Javascript / Typescript
  ○ Angular, live and deployed at https://home-5013300400.app-ionos.space/
  ○ Bootstrap

❖ Back End
  ○ Java
  ○ Spring Boot with Spring Security, acting as a RESTFUL API between the front-end and backend, handling all requests.
  ○ Hibernate
  ○ MySQL, live connected to AWS as the backend

Home

- ❖ Create Impression that captures what Rally represents
  - ○ Logo
    - ○ Simple, understandable, expresses a CTA
  - ○ Colors - visual impression
    - ○ Green: freshness, vitality
    - ○ Orange: bright, positive
    - ○ Blue: peace, trust
  - ○ Language - sense of community
    - ○ We, us, our - part of something

My Profile

❖ Create an online persona on the Rally application to engage with the Rally community.

❖ Login to an existing account or Register as a new user.

❖ Registration process:
- Create a user name
- Enter a valid email address
- Create a password
- Enter user information to display on profile
- Validate the account via confirmation email
- Login!

❖ Profile Picture/ Search/ Direct Message
- Add Profile Picture
- Make some friends
- Search users by Name or by characters
- Send a Direct Message to someone
- Update your Information

❖ Post History
- We need to make some posts to see this section in action. Let's see this in action with an established account.
- View Services, Events, and Forum Posts.
- Hide Posts from your history, or show them again when you're ready.
- Let's explore the rest of what Rally has to offer!

❖ User Security
- Users are issued an access token using JWTs
- Passwords are encrypted using Bcrypt
- Using a real email helps to mitigate fake accounts being created.

Events

- ❖ Event List
  - ○ filter by category/date
- ❖ Event Page
  - ○ updates # attending, comments
  - ○ Google map
- ❖ Create Event
  - ○ stock photo
  - ○ accessible from user profile
  - ○ Creator - view UPDATE button on Event page
    - ○ populates form
    - ○ edit or delete

- ❖ Join Event
  - ○ comments, # attending updated
  - ○ accessible from user profile
  - ○ Joined - view UPDATE RSVP on Event page
    - ○ populates form
    - ○ edit or delete
- ❖ Future
  - ○ photos, map, responsive design

Services

- ❖ Home page to Add Service or Search Services
- ❖ Add Service Form
  - ○ Services Repository
  - ○ Tagging for Categories and Service Type (by offering or requesting)
  - ○ Redirecting to a new form after submission
- ❖ Search Services Table
  - ○ Displaying all services
  - ○ Two-way sorting, filtering by Service Name and Service Type
  - ○ Links on each item to the full description
- ❖ Service Item View
  - ○ Full details of the item to include description and contact information
  - ○ Option to message the poster to get more information or book the service
  - ○ If the original poster, button intended to remove the item if desired – but not part of MVP

Resources

❖ User Actions:
  ○ **Create**: new resource
  ○ **Read:** existing resources
  ○ **Update:** edit resource details
  ○ **Delete:** remove a resource
❖ Resource Details:
  ○ Name, Address, Category, City, State, Zip, Website, Email Address, Telephone Number
❖ Resource Categories:
  ○ Athletics, Arts, Business, Civic, Education, Entertainment, Fitness, Hospitality, Medical, Park, Religious, and Retail
❖ Challenges
  ○ Spent a lot of time troubleshooting 400 errors related to different nomenclature in post and get requests from front to back-end

Restaurants

- Home page -view restaurants by restaurant type, perform search, view all restaurants or add a restaurant to the website
  - search- displays given restaurant/s according to users search
  - view all- redirects user to view a list of all restaurants
  - add restaurant-form submission that add a restaurant to the database and redirects user to view all restaurants in the same category upon submission

- ❖ Clickable "review" link on all restaurant redirects users to a review component, within which they can leave a review for the given restaurant

- ❖ Challenges- linking front and back end via get/post requests in typescript files.

Community Forum

- ❖ Forum home page
  - ○ Five Different Topics to Choose From, Each Linked to a that Forum Topics specific route.
- ❖ Forum Topic Pages
  - ○ List of posts, sorted by newest to last displayed
  - ○ Like buttons, with the ability for a User to "like" a post he enjoys or agrees with. A user cannot like the same post twice, clicking again will undo like.
  - ○ Ability to click on the name of the users that posted, linking to their user profiles.
  - ○ Ability to click on each Post title and take it directly to their individual view pages
  - ○ Authentication - Ensures that only logged in users can create or like posts. Logged out users can only view
- ❖ Forum search page
  - ○ All pages come equipped with a search bar on the right-hand side, the search will catch any username, post title, or post description.
  - ○ Displays list of posts that come in the search, with the ability to click on the name of the users that posted each one, linking to their profiles.
- ❖ View Individual Post Page
  - ○ CRUD - Ability to create, update, or delete both your individual posts or replies.
  - ○ Authentication - Ensures that only the person who posted both the post or reply have the ability to edit or delete them.
- ❖ Miscellaneous
  - ○ Light-dark theme, ability for a user to switch between a light and dark theme, their preference being saved in local storage.

Questions?