COMPONENTS: Application Break Down

| BACK END | FRONT END |

View Partial: Navbar

- Bootstrap Navbar
- One for logged in
- One for logged out
- Buttons for:
-> Home
-> Options (turning platforms on and off)
-> Sign in/out (or Sign up)
-> Account (settings)

Likes System

- Implement acts_as_voteable gem
- Edit views as necessary
- Modify necessary controllers
- Follow documentation's steb-by-step

Facebook API

- Basic Info
- Login/Sign up function
- Friends list
- Likes list
- Build proof-of-concept for collecting data

Google API

- Basic Info
- Login/Sign up function
- YouTube subscribers list
- YouTube subscribed list
- Likes (If possible/viable)
- Build proof-of-concept for collecting data

View - Main Feed

- Template for suggestion items
- Documented names for assets (social icons) in a <!-- comment -->

View - Flash Notices

- Confirmation of sign up
- Confirmation of password recovery
- 404 and other errors

Notes: A very generic layout with a header and paragraph, with an obvious link to navigate to the home page.

Suggestion Feed

- Edit view for dynamic population of suggestions
- Load while scrolling
- Create necessary controllers

Suggestion System

- Create efficient algorithm
- Consider using Redis server for efficiency
- Determine necessary output for suggestions
-> UJS?
-> ERB?
-> JSON?

Tumblr API

- Basic Info
- Followers
- Followed
- Topics followed
- Likes (If possible/viable)
- Build proof-of-concept for collecting data

Pinterest API

- Basic Info
- Followers (users)
- Followed (users)
- Followers (boards)
- Followed (boards)
- Likes (If possible/viable)
- Build proof-of-concept for collecting data

View - Sign In

- Buttons to sign in with Facebook, Twitter, Google
- Form for:
-> email or username
-> password

View - Sign Up

- Buttons to sign up with Facebook, Twitter, Google
- Form for:
-> email
-> username
-> password
-> password confirmation

Sign Up/In System

- Implement clearance gem
- Edit view as necessary
- Modify necessary controllers
- Modify config as necessary
- Ensure models/migrations support clearance gem use
- Follow documentation's steb-by-step

Database

- Design
- Create necessary models
- Create necessary migrations
- Create columns based on social API's data

Twitter API

- Basic Info
Login/Sign up function
- Followers
- Followed
- Likes (If possible/viable)
- Build proof-of-concept for collecting data

Instagram API

- Basic Info
- Followers
- Followed
- Likes (If possible/viable)
- Build proof-of-concept for collecting data