1. Component Structure:

**breedsearch**/

```
├-------------------- src/
|                  |── components/
|                         |---------- searchBar.js
|                         |---------- favorite.js
|                         |---------- favoritecontext.js
|
|                  ├── style/
|                         |── searchBar.css
|                         |-----favorite.css
|
|                  ├── App.js
|                  ├── App.css
|                  ├── index.js
```

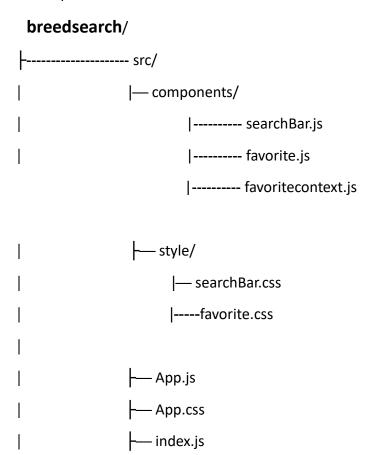**2. searchBar.js** : Fetching Dog Images

- The "DogImages" component is responsible for fetching and displaying dog images based on the entered breed.
- Users can enter a breed name in the search bar and click the "Search" button.
- The "fetchImages" function sends an API request to the Dog API using the entered breed.
- If the API call is successful, the first 10 images are displayed.
- If the API call fails or the entered breed is invalid, an error popup is displayed.

**3. FavoriteContex.js** : Managing Favorites

- The "FavoriteContext" context manages the state related to favorite images.
- The "FavoriteProvider" component wraps the app to provide the `favorites` state and functions to child components.

- The "addToFavorites" function adds a new image URL to the `favorites` state if it's not already present.
- The "removeFromFavorites" function removes an image URL from the `favorites` state.

**4. Favorites.js** : Displaying Favorite Images

- The "Favorites" component displays a list of favorite images.
- It uses the "useFavoriteContext" hook to access the "favorites" state and "removeFromFavorites" function.
- Favorite images are displayed along with a "Remove" button.
- Clicking the "Remove" button removes the corresponding image from the favorites list.

**5. App.js** : Routing and Navigation

- The "App" component sets up the routing for the application using `react-router-dom`.
- It provides the "FavoriteContext" to its child components through the "FavoriteProvider"
- Users can navigate between the "Home" and "Favorites" views using the navigation bar.

**6. Styling**

- CSS styles are used to style the components and create a user-friendly interface.
- The UI includes a search bar, image display, "Favorite" button on hover, navigation bar, and favorite image list.

# Workflow:

1. The user enters a dog breed in the search bar and clicks the "Search" button.

2. The "searchbar" component fetches dog images based on the entered breed using the Dog API.

3. The fetched images are displayed in a list, and users can mark images as favorites by hovering over them and clicking the "Favorite" button.

4. Users can navigate to the "Favorites" view using the navigation bar.

5. In the "Favorites" view, users can see their list of favorite images and remove images from the list by clicking the "Remove" button.