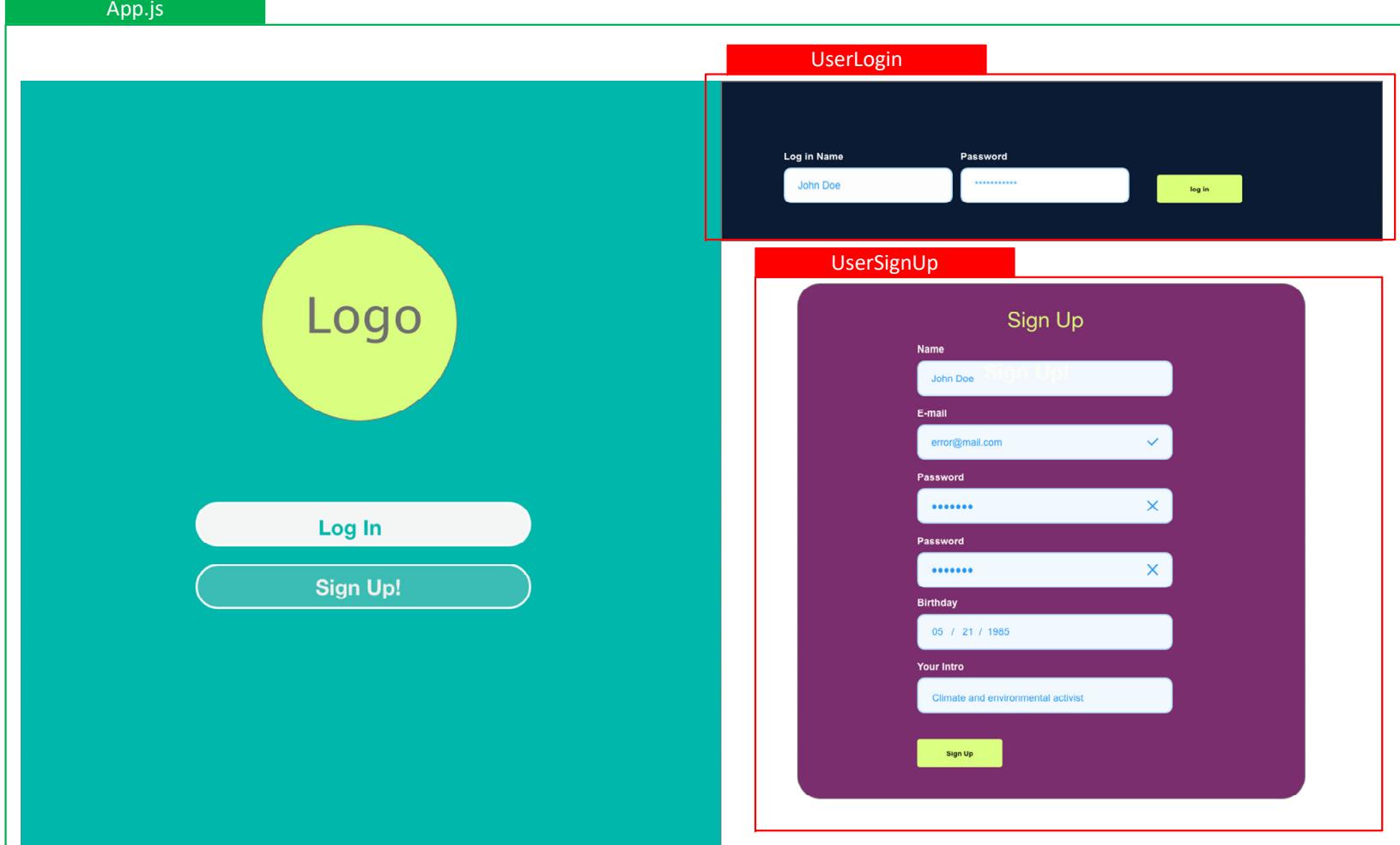


App.js



UserProfilePage.js

SiteNav

UserLogin

log in Sign Up!

Search

User Profile

Name Surname
Climate and environmental activist

E: name.surname@mail.com
Bday: April 1, 1985

UserProfileSummary

UserProfileDetail

About User.

109 Posts 1.5M Followers 71 Following

User Posts

See Posts

TimelinePage.js

UserLogin

log in Sign Up!

UserProfileSummary

Name Surname
Climate and environmental activist

E: name.surname@mail.com
Bday: April 1, 1985

UserPosts

UserPost

Post 1

Culpa qui officia deserunt mollit anim id est laborum. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi ropeior architecto beatae vitae dicta sunt.

share

Post 2

Culpa qui officia deserunt mollit anim id est laborum. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi ropeior architecto beatae vitae dicta sunt.

share

Component	UserSignup	UserLogin	UserProfilePage	UserProfileSummary	UserProfileDetail	UserPosts	UserPost	TimelinePage	Search	SearchPage
Props (inputs)			username	username	username	username postCount	username fullName emailAddress profilePicture postTime postText	username		searchTerm
Internal State	username fullName emailAddress password1 password2 birthday profilePicture briefSummary	username password			userInterests (external API GET)				searchTerm	[searchResult]
Store actions	READ username WRITE username password fullName emailAddress birthday profilePicture briefSummary isLoggedIn	READ username password isLoggedIn WRITE isLoggedIn		READ username fullName emailAddress birthday profilePicture briefSummary		READ username fullName profilePicture posts				READ username
Output/ Expected action	Create the user on completion by adding all the variables to Store, and automatically log user in. Route user to UserProfileSummary page or TimelinePage.js (TBC)	Set the isLoggedIn value to true for the user. Route user to UserProfileSummary page or TimelinePage.js (TBC)	Pass the prop containing active username to each of the child components, UserProfileSummary and UserProfileDetail. The user prop is received from React Router containing the username.	Display user information based on the data from the store.	Display userInterests state var (array of random interests drawn from the API), represented as a list Display UserPosts component. Pass username and postCount props to the Component.	Take optional username and postCount props to retrieve the store data. Pass data to UserPost component for display.	Display received data from the props and return list item.	Check that the username that's logged in corresponds to the username passed through via React Router. If not, reroute to Login page. If good, pass the username to different components on page to display information accordingly	Ensure that a valid search string is passed on to the SearchPage component.	Pass array of usernames that conform to the search string to the UserPost component with postCount = 1

