# **Object Oriented Component Documentation**

Server-side of this web application as built on NodeJS and ExpressJS, which comprised of middleware, functions and logic from npm packages. As such, it did not rely on an object-oriented component structure.

Conversely, the client-side of the web application was built on React using a combination of functional and class-based components. Data was passed through local state and interpolated onto the browser as either page elements or form data.

Below is a tabulated break own of the components used on the client-side.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name of Component** | **Class or**  **Functional** | **Data being passed through state** | **Parent components** | **Child components (direct)** |
| WithState | Class | Sets up global state for the rest of the App | N/a | App |
| App | Class | Global state from WithState | WithState | Nav, Foot, Loader, Sign up, Edit Profile, Log in, View All Profiles, |
| Nav | Functional | isLoggedIn flag (stored in local storage which triggers different nav) | App | N/a |
| Foot | Functional | n/a, static | App | N/a |
| Loader | Functional | n/a, static | View All Profiles | N/a |
| Home | Functional | n/a, static | App | n/a |
| About | Functional | n/a, static | App | N/a |
| Edit Profile | Class | studentData object | App | N/a |
| Sign up | Class |  | App | N/a |
| Log in | Class | Dynamic; state (email, password) | App | N/a |
| Profile (view single profile) | Class | studentData object, isLoading flag | View All profiles | Loader |
| View All Profiles | Class | isLoading flag,  filter parameter, studentData object | App | Profile Entry, Loader |
| Profile Entry | Functional | studentData object | View All Profiles | N/a |
| Unauthorised (401) | Functional | n/a | N/a | N/a |
| Error (404 catch all) | Functional | n/a | N/a | N/a |