

# React Firebase Dashboard

**Weight: 20**

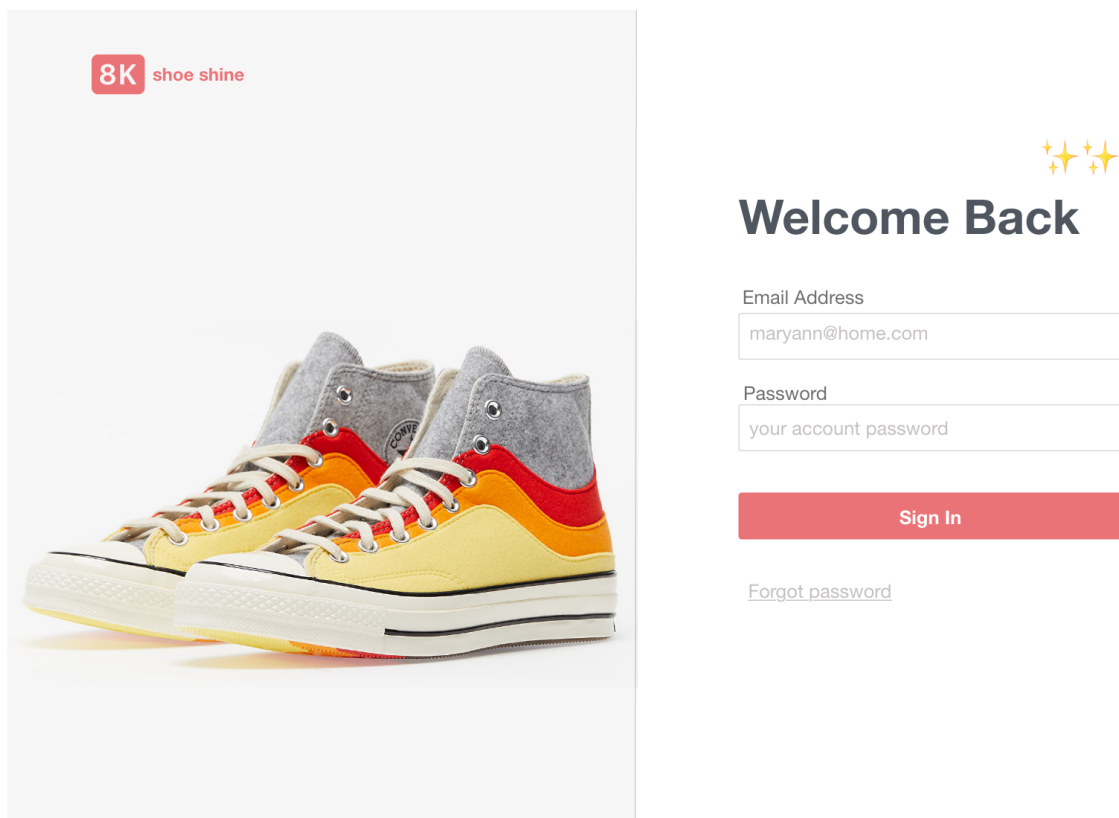
**Due Date: Friday March 4th 2022**

**Submit Before: 11:59pm**

## Introduction

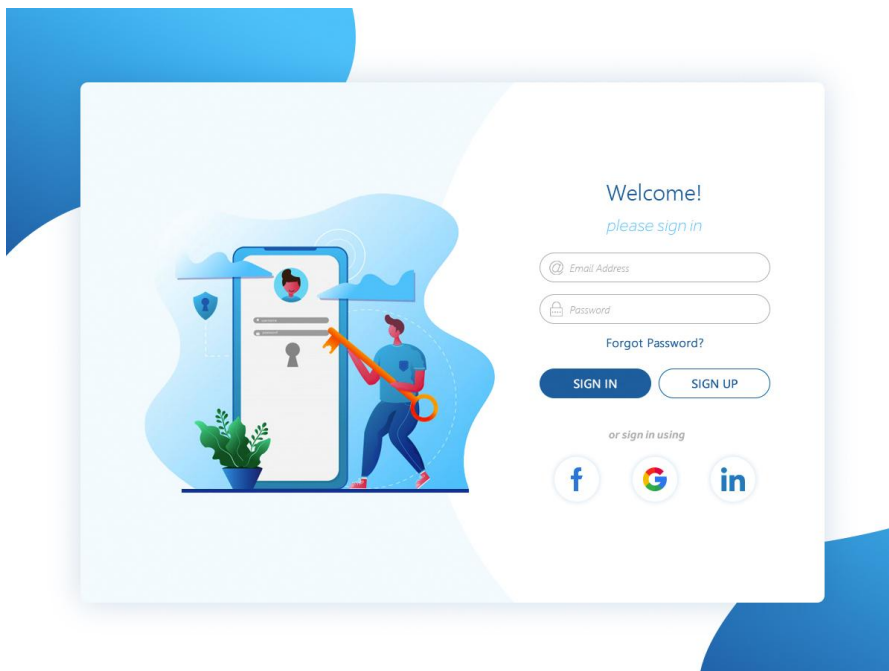
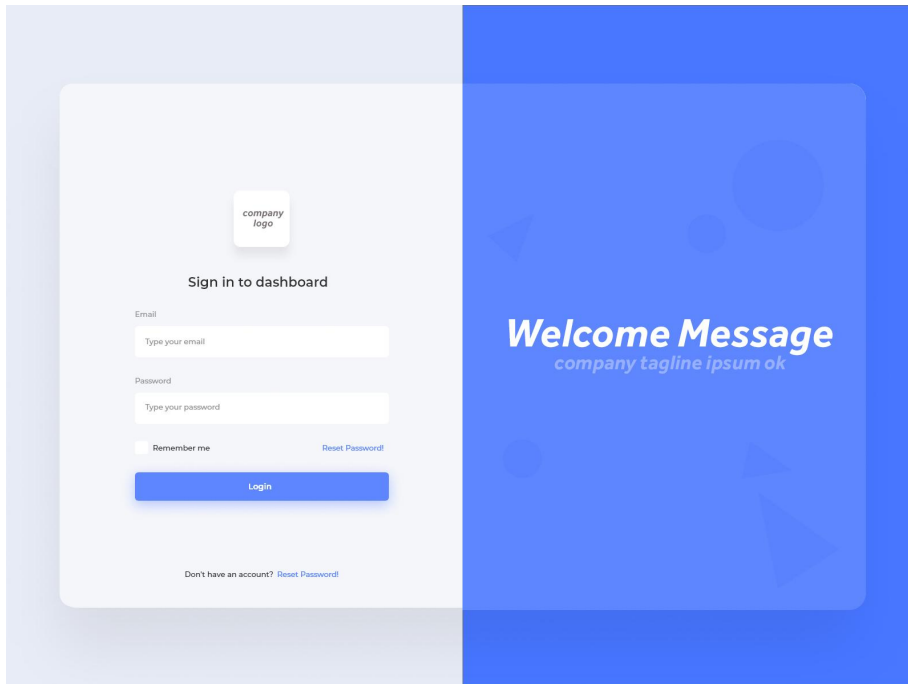
For the first assessment you will be creating the login screen for your dashboard. In class you were shown several ways to style components with React. You may pick any method that suits your comfort level with CSS. Choose to use CSS, CSS Modules, SASS, SASS Modules or Style Components. You may even want to use a pre-build UI component library such as Chakra or React Bootstrap.

Below is an example login page. Your page should have an email and password field and a sign in button including a link to recover email and password.



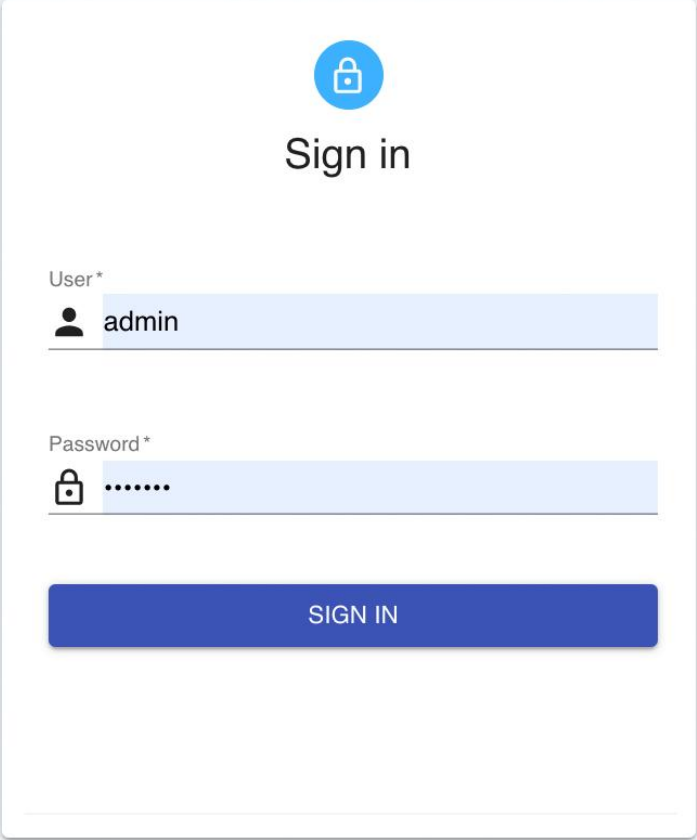
## Examples Of Acceptable Login Screen

At large viewport sizes (desktop) your login screen should contain two columns. One column will contain your form for user sign in. The second column should be used for additional messaging such as branding or company mission statements.



## Examples Of Unacceptable Login Screen

Just having the login screen as shown below is not acceptable. This would be ok at small viewport sizes but for large viewport sizes your login screen should be more akin to the previous examples.



### Note

Your login form does not have to be validated. There isn't time to completely build out every feature that the form should have. A minimum requirement is that you add the input field attribute required to both the user and password fields.

## Form Scripting

The sign in button is there only to trigger the form submit event. In class you were shown how react handles events. Your form script should prevent the default behavior. Add a redirect to the DashboardPage.js component to simulate a login.

## DashBoardPage.js

Your DashBoardPage component must have the appBar, sidebar and panels components laid out.

### AppBar Component

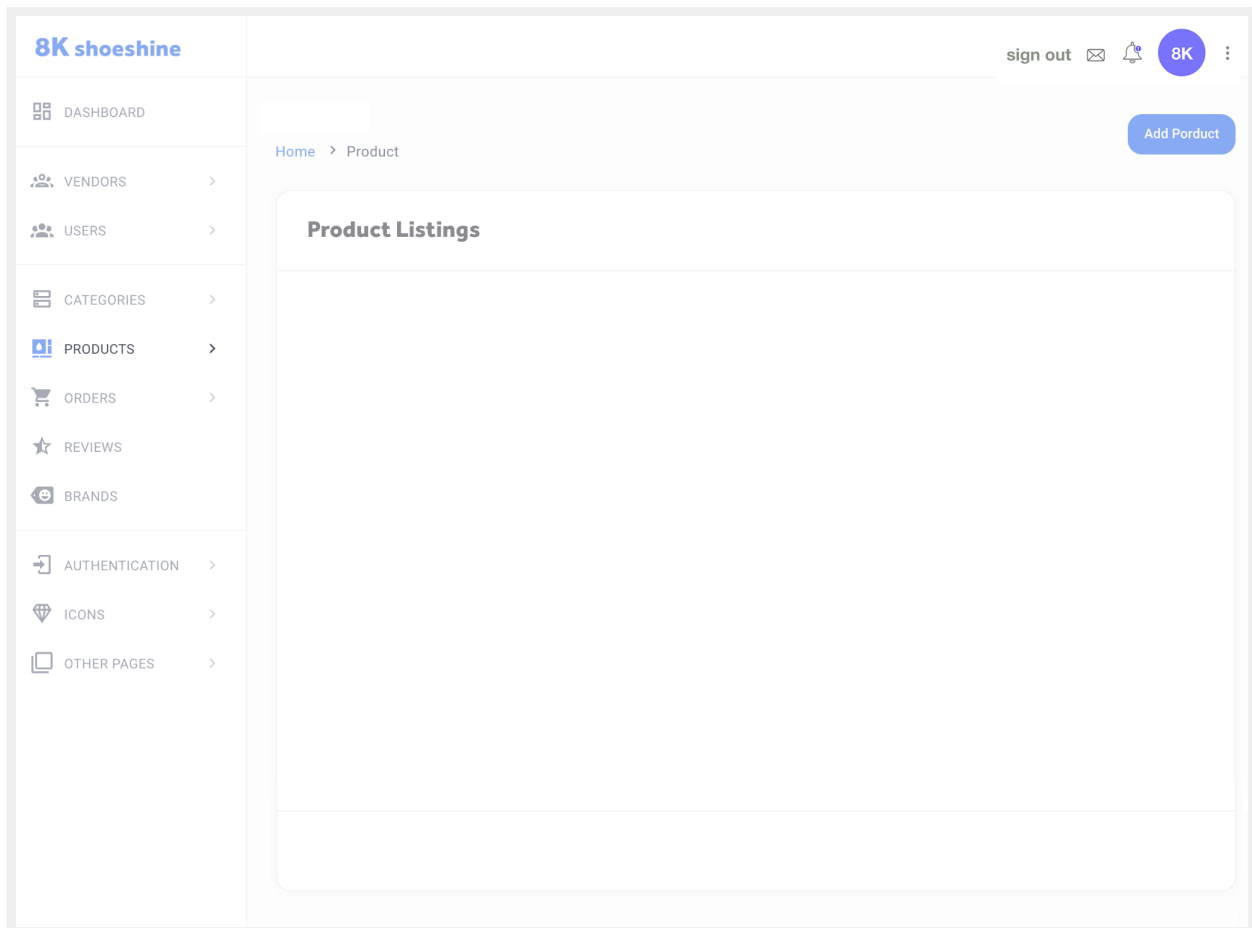
The app bar should have the company branding on the left. On the right side of the appBar you should have a minimum of 4 icons buttons, one for the settings, avatar, notifications and email.

### SideBar Component

The SideBar.js component mustv contain the link to sign the user out. This link will return the user to the LoginPage.js component. The sidebar must have a minimum of 12 icon buttons representing different features of the dashboard see the example provided. You can make these up to suit your dashboard style and content.

### Panels Component

The Panels.js component should have a single panel with the heading Product Listings and any other features you wish to add at this time.



## Routing And Routes



In class you were shown how to set up and configure react-router-6 with create react app. Use the template you created in class to add routing between the LoginPage and the DashBoardPage.

The LoginPage and the DashBoard page should be in a folder called pages.

```
components
pages
  LoginPage.js
  DashBoardPage.js
ui
App.js
index.js
scss
```

### Index Route

The index route will be the login page. React router has a specific way that it wants you to handle the index route. The path to match will be / and when this appears in the browser address bar, react router will display your LoginPage.js component.

### Dashboard Route

For the DashBoardPage.js component to show we will tell, react router that when the path in the address bar is /dashboard then we want to display the DashboardPage.js component. Your dashboard doesn't have to be laid out completely yet you will do that in the next assignment. For now your dashboard should contain the appBar, sidebar and panel components.

For the DashBoardPage.js you have to create a sign out button. This button will be used to mock a sign out request using react router to push a route and load in the LoginPage.js component.

### Page Not Found Route

For all other paths that are entered into the address bar for our app we will display a 404 Page, which is the standard web error for a web page not found. You can design your own custom 404Page.js to display. Make it thematic with the content you are building.

## Submission

If you haven't already created a github repository for your project then go ahead and make one.



Do not use the in class repo that you have been using while learning React. This repo is for your product dashboard.

Deploy your site to netlify.

### **Moodle**

Upload the link to your github repo

Upload the link to your Neltify Site Deployment.

Late Assignments will not be graded.

Any site that displays a 404 error will receive a grade of zero



## Marking Key

Task		Task Value
<b>LoginPage.js</b>  Form components email, password and sign in button. Sign in button links to dashboard page. Front page is styled thematic to the product contents. The Login page must be branded with the name of the company. Form redirects to the DashboardPage.js		<b>6</b>
<b>App.js</b>  Router is added to the App entry component correctly. Components are properly imported and each route contains the correct path as per assessment instructions (see routing) LoginPage.js form submit event handler programmatically redirects to DashBoardPage component. Sign Out Button in the dashboard returns the user to the login page.		<b>3</b>
<b>Dashboard Page</b>  Dashboard has sidebar, appbar and panel components displayed. App Bar Logo, user icon settings icon 3 other icon choices. Sidebar contains sign out link to return user to the LoginPage.js Sidebar contains a minimum of six icon buttons for different sections of a dashboard. These are content only and do not have to work. Panel component simply needs a header Product Listings.		<b>9</b>
<b>Deployment and Github Repo</b>  Site deployed to Netlify using a CI/CD integration with Github. GitHub repo is well organized and contains a professional looking Readme.md file. (No assignment 2 naming) Naming should be branded to the company/organization that is selling the product.		<b>2</b>
		<b>20</b>



## Marking Rubric

Marks	5 Marks Criteria
<b>5 [0]</b>	Task was completed with the highest proficiency, adhering to best practices. The tasks demonstrate a high level of proficiency implementing the subject matter and the presentation of the content is to a professional standard.
<b>4 [-1]</b>	Task was completed well, with some minor mistakes. Well above average work, shows good understanding of the implementation details of subject matter. Tasks show a high degree of competence.
<b>3 [-2]</b>	Task was completed satisfactorily. Some features are missing or incorrectly implemented. Shows a moderate level of understanding in the task with room for improvement.
<b>2 [-3]</b>	Task completion is below average, the task was poorly completed. Shows understanding of the task and the requirements to implement, but implementation was poorly executed.
<b>1 [-4]</b>	Some of the tasks were completed. Shows a lack of understanding in the subject matter and very poor execution.
<b>0 [-5]</b>	Not completed.

Marks	3 Marks Criteria [minus]
<b>3 [0]</b>	Task was completed well, adhering to best practices, and shows a high degree of proficiency with the subject matter guidelines.
<b>2 [-1]</b>	Task was completed satisfactorily. Some features are missing or incorrectly implemented. Shows a moderate level of understanding in the task with room for improvement.
<b>1 [-2]</b>	Some of the tasks were completed. Shows a lack of understanding in the subject matter and very poor execution.
<b>0 [-3]</b>	Shows a little to no degree of competence in completing the task; not completed.

Marks	1 Marks Criteria
<b>1</b>	Task completed satisfactorily
<b>0</b>	Task was not completed