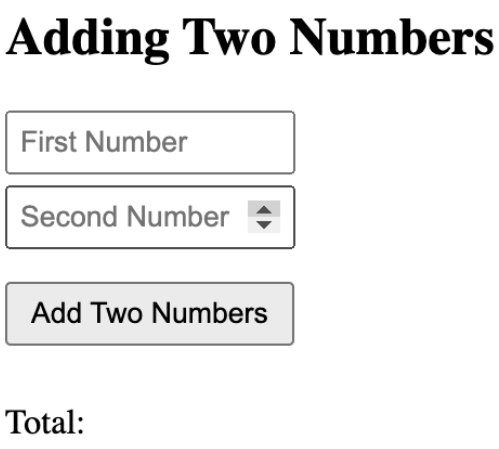
Coding Questions

1. Challenge 1: Create a **React Native** page with any design that works like a calculator that can add two numbers.

   Functionality: When user place numbers on first and second input and hit the button. The sum should appear on the `Total: ` as an output.

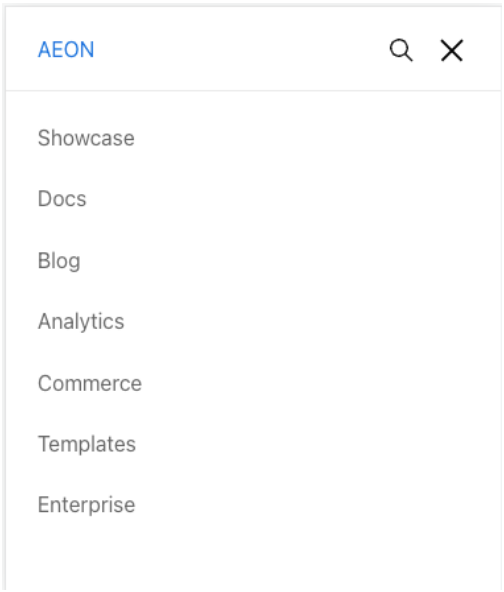
   Please refer to the image below;



2. Challenge 2: Create a blank page using **React Native** that contains the navbar as below.
    a. Navbar title can be any name. You can follow the sample image below
    b. Navbar should use <a> tag with href=”/”
    c. Please only create a search input without any functions.
    d. Mobile view is optional.
        i. The navbar should close when user click [X]
        ii. Replace X icon with Hamburger icon
        iii. The navbar should open when user click the icon

Desktop View

Mobile View (Bonus)



3. Challenge 3: Two Sum II - Input Array Is Sorted

Create a simple page using **React Native** that can cater for following functionalities;

Given a **1-indexed** array of integers numbers that is already ***sorted in non-decreasing order***, find two numbers such that they add up to a specific target number. Let these two numbers be numbers[index1] and numbers[index2] where 1 <= index1 < index2 < numbers.length.

Return *the indices of the two numbers,* index1 *and* index2*,* ***added by one*** *as an integer array* [index1, index2] *of length 2.*

The tests are generated such that there is **exactly one solution**. You **may not** use the same element twice.

Your solution must use only constant extra space.

You can use Javascript or Typescript. Please provide your answer as sample structure below.

```
/**
* @param {number[]} numbers
* @param {number} target
* @return {number[]}
*/
const twoSum = function (numbers, target) {
       // your code here
};

function twoSum(numbers: number[], target: number): number[] {
       // your code here
}

console.log(twoSum([4, 11, 17, 25], 21));
console.log(twoSum([0, 1, 2, 2, 3, 5], 4));
console.log(twoSum([-1, 0], -1));
```

Example 1:

```
Input: numbers = [2,7,11,15], target = 9
Output: [1,2]
Explanation: The sum of 2 and 7 is 9. Therefore, index1 = 1, index2 = 2.
We return [1, 2].
```

Example 2:

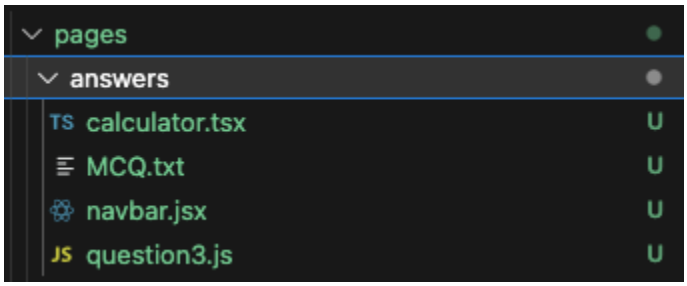
```
Input: numbers = [2,3,4], target = 6
Output: [1,3]
Explanation: The sum of 2 and 4 is 6. Therefore index1 = 1, index2 = 3. We
return [1, 3].
```

Example 3:

```
Input: numbers = [-1,0], target = -1
Output: [1,2]
Explanation: The sum of -1 and 0 is -1. Therefore index1 = 1, index2 = 2.
```

Notes:

1. Provide us with how to run your code in a readMe file
2. Please complete the test given in **Javascript or Typescript**
3. Please use **React Native** for the framework
4. Option 1: send us the answers in a zipped file via Google Drive link or Option 2: you can also provide your answers through Github. Please provide us with the repository link.
5. See example below:



6. If you are shortlisted, we are expecting you to **demo and walk us** through your solution from your machine in the interview after submission