# Assignment

# The challenge is a Software Engineer Intern.

### Technology

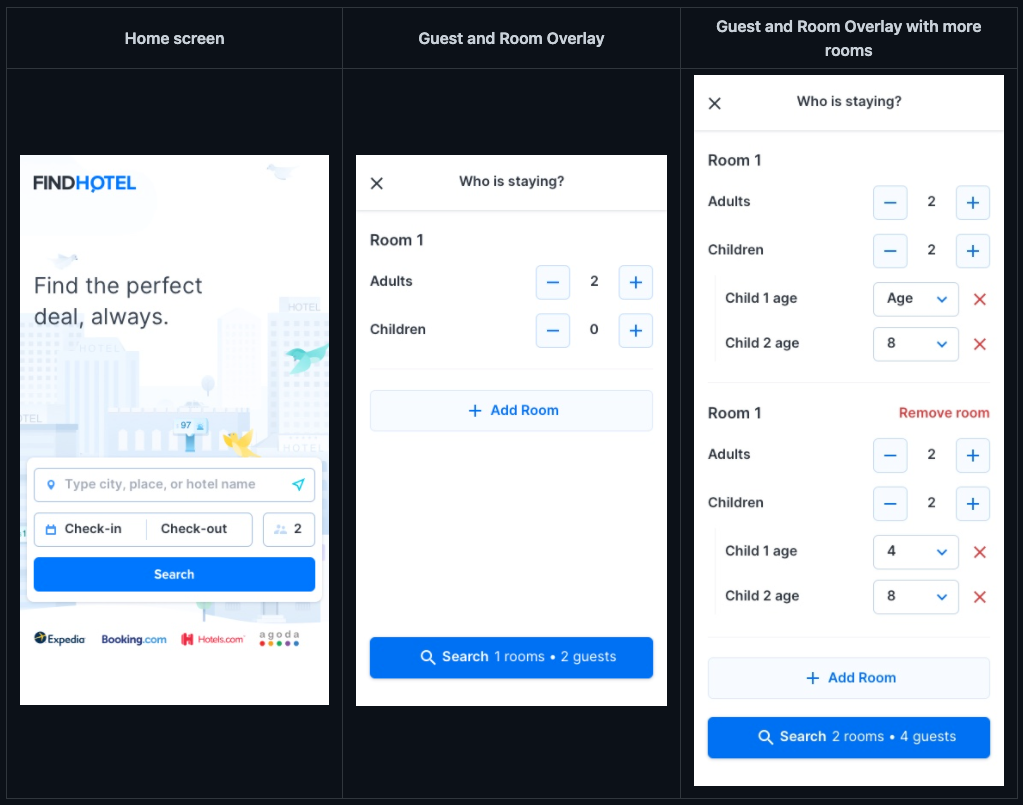
# setup project using React

# Material as a CSS tool

# Prefer using [TypeScript](https://www.iamtk.co/a-mental-model-to-think-in-typescript) (or Flow)

### UI

### The idea is to create a Guest and Room Overlay component. The user can open it, add different rooms, select any limited number of adults and children, and select the children's ages.



**Input rules**

The component should be able to pass a string as the default data. These are the rules:

* Rooms are separated by pipe |
* Adults and children are separated by a colon :
* Children's ages are separated by a comma ,

Examples:

* "1:4,6|3" → Two rooms, one with one adult and two children ages four and six and the other with three adults and no children
* "3" → One room with three adults and no children
* "2:4" → One room with two adults and one child aged four
* "1:0,13,16" → One room with one adult and three children (aged zero, thirteen, and sixteen)

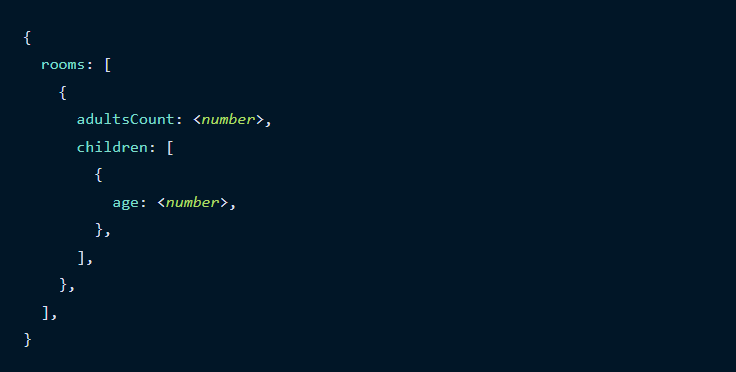
**Functional Requirements**

* Up to eight rooms can be added
* Each room has at least one adult and up to a maximum of five
* Each room has zero or more children up to a maximum of three
* Each child needs to have their age supplied, so we know what kind of bed or cot to provide and what to charge for the room
* Each room has a maximum occupancy of five. This is, adults plus children per room
* The Guest and Room selector should always yield a valid room occupancy, use button disablement to avoid invalid configurations

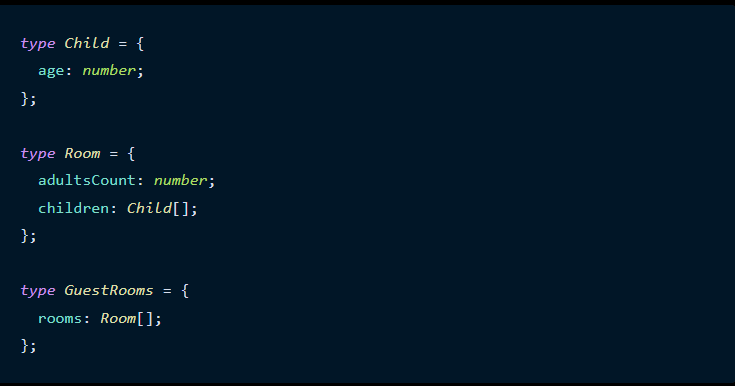
### Solution:

### Data Structure and State management:

The solution data structure for the above UI would look like this.



#### Type Script Implementation for the same would be:



#### Behavior of the second room page features would be like:



#### Context Provider: to handle states, we use context API and provide the state’s data structure and function APIs to every component that is wrapped by the context provider. Provider just wraps our component and provide a value for it.



In the above code guestRooms is a state and setGuestRooms is the func API to update the state.

Better data structure along with separators would look like:

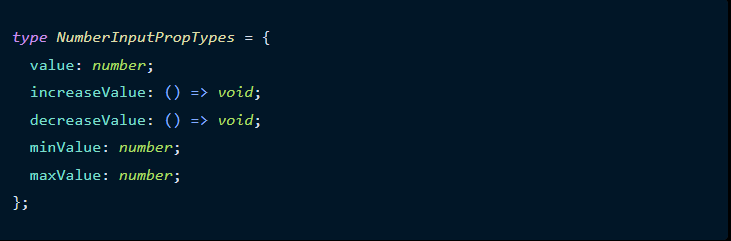
Using separators to get each meaningful data and returning the GuestRooms data structure.

We can easily test it as a pure function:

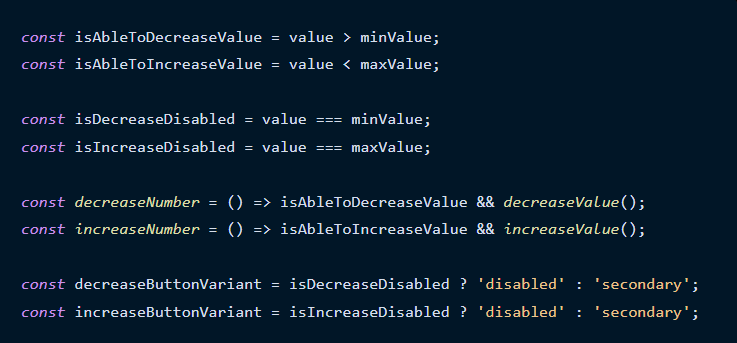


Number Input:

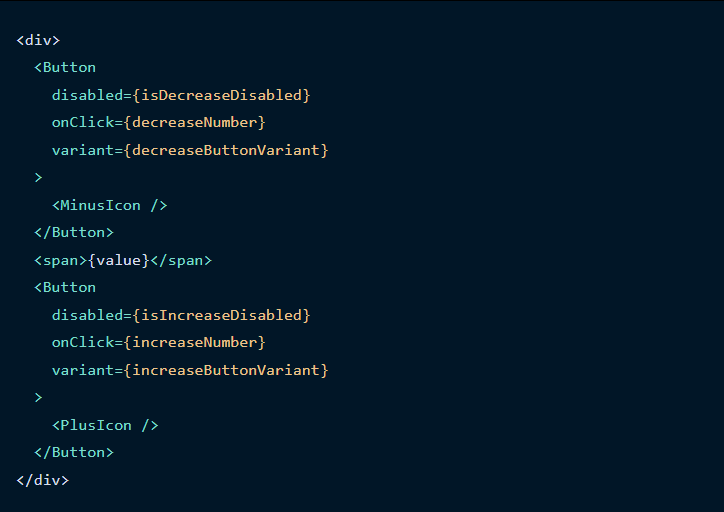
Block for adult count input and children count input.



To disable (or not) the increase and decrease buttons.



UI implementation for above implementation:



# 