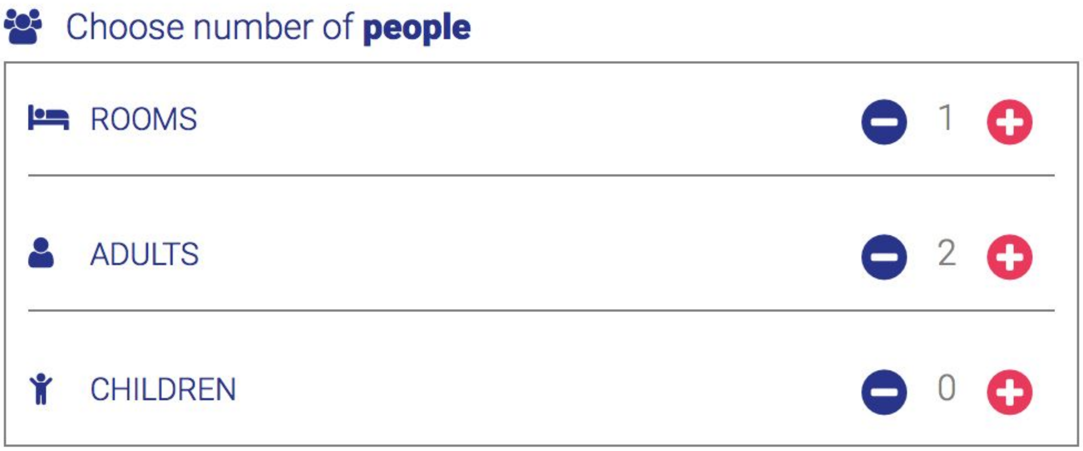
The programming exercise is to build the following component



Choose number of **people**

ROOMS ⊖ 1 ⊕

ADULTS ⊖ 2 ⊕

CHILDREN ⊖ 0 ⊕

The + and - buttons for each of Rooms, Adults and Children should be enabled/disabled and the value updated, based on the following constraints:

- Each of the + or - buttons can be clicked independent of the other
- At no point of time can an invalid combination of data be possible in the UI
- # Rooms can be minimum 1 and maximum 5
- # of persons in each room (i.e. total of Adults and Children) can be minimum 1 and maximum 4
- # of Adults can be >= 1 (i.e. a booking should have at least one adult)
- # of Children can be >= 0
- # of persons (Adults + Children) can be >= # of rooms (i.e. there should be at least one person per room)
- If Adult or Child count is increased, the room count can be increased, if required, to meet the constraint
- If Adult or Child count is decreased, the room count can be decreased, if required, to meet the constraint
- No count can automatically change unless required to meet the constraint
- If room count is increased, the adult count can be increased to the extent required to meet the constraint

	Current	Room + clicked
Rooms	2	3
Adults	2	3
Children	0	0

- If room count is decreased, if # of persons exceed maximum allowed, first # of children can be reduced upto 0, if present; then # of adults can be reduced to the extent required to meet the constraint

	Current	Room - clicked
Room	2	1
Adults	7	4
Children	1	0

- Buttons should be disabled if clicking them breaks the constraints. There should be no alert messages.

Examples

- - button should be disabled for room when 1 is the room count
- If + button of room is clicked, room count should change from 1 to 2
- If + button of room is clicked again, room count should change to 3 and the Adults count should also change to 3

Other Information

- The solution should be implemented and checked into a github repository.
- The repository should have a README.md which contains any information that is required to run the application
- While it is desirable to have the same look and feel as the above design, it is ok to make variations so long as the functionality is unaffected
- Unit test for the component would give bonus points