

```
dynamicComponents () {  
  let input, actionButton;  
  const {editMode, todo} = this.state;  
  
  if(editMode) {  
    input = (  
      <input  
        type="text"  
        className="form-control"  
        value={todo}  
        onChange={(e) => this.handleChange(e)}  
      />  
    );  
    actionButton = (  
      <button  
        type="button"  
        className="btn btn-warning text-white"  
        onClick={this.onEditMode.bind(this)}  
      >  
        Actualizar  
      </button>  
    );  
  }
```

```
    },  
  } else {  
    input = todo;  
    actionButton = (  
      <button  
        type="button"  
        className="btn btn-warning text-white"  
        onClick={this.onEditMode.bind(this)}  
      >  
        Editar  
      </button>  
    );  
  }  
  return {input, actionButton};  
}
```

- Unificamos los diferentes cambios realizados con la lógica que establecimos en cada componente y estructuramos en el `jsx`.

Complexity is a it's time to do something...

```
render () {  
  const {index, onRemove} = this.props;  
  const dynamicComponents = this.dynamicComponents();  
  
  return (  
    <tr style={{textAlign: 'center'}}>  
      <th scope="col">#</th>  
      <th scope="col">{index + 1}</th>  
      <th scope="col">{dynamicComponents.input}</th>  
      <th scope="col">  
        <div className="btn-group" role="group">  
          {dynamicComponents.actionButton}  
          <button  
            type="button"  
            className="btn btn-danger"  
            onClick={onRemove}  
          >  
            Eliminar  
          </button>  
        </div>  
      </th>  
    </tr>  
  );  
}
```