



**Certified Tech Developer**

The Ultimate Degree

## Front End II

# Requerimientos funcionales

## To-Do App v.4

- ☐ Dentro del script **tasks.js** procederemos a crear una función dedicada al renderizado.
  - ☐ Esta función debe recibir como parámetro un **listado**.
  - ☐ En cada iteración se debe evaluar si la tarea corresponde al listado de "tareas-pendientes" o "tareas-terminadas".
  - ☐ En cada contenedor del html se debe insertar una un ítem que respete la estructura de la siguientes plantillas:
  - ☐ Tarea terminada

```
<li class="tarea">
    <div class="done"></div>
    <div class="descripcion">
        <p class="nombre">${tarea.description}</p>
        <div>
            <button><i id="${tarea.id}" class="fas
fa-undo-alt change"></i></button>
            <button><i id="${tarea.id}" class="far
fa-trash-alt"></i></button>
        </div>
    </div>
</li>
```

- ☐ Tarea pendiente

```
<li class="tarea">
    <div class="not-done change" id="${tarea.id}"></div>
    <div class="descripcion">
```



```
                <p class="nombre">${tarea.description}</p>
                <p class="timestamp"><i class="far
fa-calendar-alt"></i> ${tarea.createdAt}</p>
            </div>
        </li>
```

- ❑ Posterior a la función anterior crear una nueva función que se encargue del **cierre de sesión**, la misma deberá sacar al usuario loggeado y quitarlo de la vista. Utilizar el botón correspondiente de la interfaz para implementar con la última función.

## Requerimientos no funcionales

- ❖ Mantener un código medianamente ordenado en bloques para poder identificar las distintas funcionalidades.
- ❖ Llamar a la función de renderizado en todos los momentos que se considere necesario.
- ❖ Una vez terminado con todo el desarrollo, pueden tomar como desafío el desarrollo de una función dedicada al cambio de estado de las tareas, de pendiente a terminada y viceversa.

## DOCUMENTACIÓN DE LA API

<https://ctd-todo-api.herokuapp.com/>