

Compte Rendu

Boutayeb Mohamed Taha

pour cette semaine c'était la phase du testing en utilisant Jasmine Framework , dont le but était de tester si certaines événements se déclenchent ou non lorsqu'on clique sur un bouton. Par exemple pour le bouton de suppression, on a opté à créer un spy sur la fonction « Window.confirm » ,cette fonction qui affiche un modal avec un message optionnel, en d'autres termes, on attend que notre fonction soit appelé après le clic sur notre bouton

Pour le code :

```
it( 'should expect a confirmation when click delete button', () => {  
    spyOn(window, 'confirm').and.returnValue(true);  
    const compiled = fixture.debugElement.nativeElement;  
    const ButtonDelete = compiled.querySelector('button');  
    ButtonDelete.click();  
    expect(window.confirm).toHaveBeenCalled();  
  
});
```

pour créer un spy on utilise la fonction spyOn() qui prend deux paramètres, le premier paramètre est le nom de l'objet alors que le deuxième et la fonction qu'on doit espionner, elle remplace la méthode à espionner et n'exécute pas réellement la méthode réelle
la fonction toHaveBeenCalled() retournera **true** si le spy a été appelé sinon elle retournera **false**.

Références :

<https://jasmine.github.io/2.0/introduction>

<https://angular.io/guide/testing>

<https://coderwall.com/p/elevha/stubbing-out-confirm-dialogs-in-jasmine>