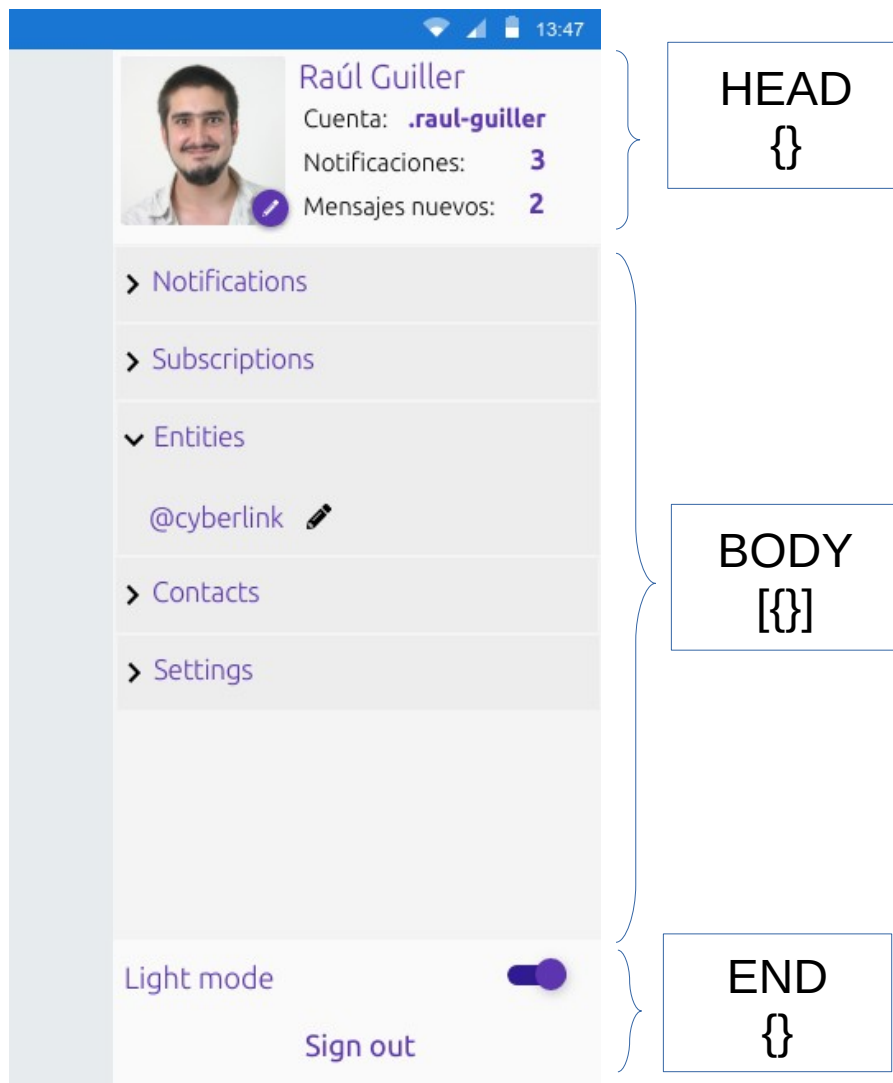


Drawer entity v 0.3

1. > PEP 8 -- Style Guide for Python Code (no soy exigente con esto, pero tenerlo en cuenta.)
 2. > Si se utiliza código estático de kivy crear archivo contenido.kv
 3. > Todo el contenido "estático" tiene que ser adquirido de un archivo
"nombre_modulo_spanish.json",. Entonces el botón que dirá "salir" lo podemos traducirlo fácilmente ;)
- "|" lo utilizo para indicar "or"
"&" lo utilizo para indicar "and"
"..." indica cantidad indefinida.
"*" argumento o key obligatoriamente necesario o utilizado



- > Desarrollar una clase partiendo de la clase Navigation Drawer (kivymd)
- > Por el momento no es necesario aplicar el ítem 3 (el de arriba), voy a ver si existe una librería o método práctico para hacer esto.

> crear metodos para actualizar head, body.

Estatic string :

```
"Cuenta:"  
"Notificacines:"  
"Mensajes nuevos:"  
"Ligh mode"  
"Sign out"
```

Propiedades y tipo:

Todos los keys cuando estan vacios se presentan no valor None

```
head['profile_picture'] : string, url de una imagen en carpeta  
head['person_name']: string, text len max (25).  
head['account_name']: string, len max (15).  
head['button_edit_pic'] : Boolean si es true, show button  
head['notifications']: numer entero.  
head['new_msg']: numer entero.
```

```
body['icon'] : string indica nombre icono kivyMD (tamaño a determinar)  
body['sub']['icon'] : string, indica nombre icono kivyMD (tamaño a determinar)  
body['text'] : string len max (15), size_font a determinar.  
body['sub']['text']: string len max (15), size_font a determinar.  
body['sub']['icon_button']: string, indica el nombre del icono kivyMD,  
el elemento a mostrar es un button con icono.
```

Drawer_entity_content =

```
*head = {'profile_picture':,  
         *'button_edit_pic':,  
         *'account_name':,  
         *'user_name':,  
         *'new_msg':,  
         *'notifications':}
```

```
*body = [{ 'icon':,  
          *'text':,  
          'sub':[{ 'icon':,  
                  *'text':,  
                  'icon_button':},  
                  ...]},
```

```
          { 'icon':,  
            *'text':,  
            'sub':[{ 'icon':,  
                    *'text':},  
                    ...]},  
          ... ]
```

```
*end = {'light_mode':True}
```

> El Drawer Navigation que queremos crear seria algo similar al que utiliza kivymd en su pagina de documentacion. <https://kivymd.readthedocs.io>, claro solo que esta si se pueda ocultar.

> El contenido del body, mostrar en una lista utilizando recycleview, utilizar Expansion Panel para cada item del root body .

Buscamos crear una vista de arbol o similiar.

> Prototipo en <https://pr.to/1UJNX3/>

Si nos basamos en el contenido del prototipo los datos dados a la clase serian estos .

```
head = { 'profile_picture': resource/profile/image.png',
        'button_edit_pic': True,
        'person_name': 'Raúl Guiller',
        'account_name': 'raul-guiller',
        'notifications': 3,
        'new_msg': 2}

body = [{ 'icon':None,'text':'Notifications', 'sub':None},
        { 'icon':None,'text':'Subscriptions', 'sub':None},
        { 'icon':None,'text':'Entities', 'sub':[{ 'icon':None,'text':'@Cyberlink', 'icon_button':'pen'} ]},
        { 'icon':None,'text':'Contact', 'sub':None},
        { 'icon':None,'text':'Settings', 'sub':None}]

end = {'light_mode':True}
```