

# Análisis de pruebas de Etherpad-lite

Curso 2018-2019

Carlos Sousa González

## Pruebas Lite Connection Test

Estas pruebas son funcionales ya que son de unidad

- 1) `api_url_need_to_be_absolute()`  
Caja negra(solo comprueba que los parametros están mal)/negativa/dinamica  
No se prueba que sea null el otro caso.
- 2) `domain_with_trailing_slash_when_construction_an_api_path()`  
Caja blanca(sabe que tiene que probar con la / al final)/ positiva/dinamica
- 3) `domain_without_trailing_slash_when_construction_an_api_path()`  
Caja blanca(sabe que tiene que probar sin la / al final)/ positiva/dinamica
- 4) `handle_internal_error_from_server()`  
Caja blanca(conoce la respuesta del servidor)/negativa/dinamica
- 5) `handle_invalid_key_error_from_server()`  
Caja blanca(conoce la respuesta del servidor)/negativa/dinamica
- 6) `handle_invalid_parameter_error_from_server()`  
Caja blanca(conoce la respuesta del servidor)/negativa/dinamica
- 7) `handle_no_such_function_error_from_server()`  
Caja blanca(conoce la respuesta del servidor)/negativa/dinamica
- 8) `handle_valid_response_from_server()`  
Caja blanca(conoce la respuesta del servidor)/positiva/dinamica
- 9) `query_string_from_map()`  
Caja blanca(sabe como contruye el string)/positiva/dinamica  
No prueba con una lista de mas de 2  
No comprueba la exception
- 10) `unexpected_response_from_the_server()`  
Caja blanca(conoce la respuesta del servidor)/negativa/dinamica
- 11) `unparsable_response_from_the_server()`  
Caja blanca(conoce la respuesta del servidor)/negativa/dinamica
- 12) `url_encoded_query_string_from_map()`  
Caja blanca(sabe como contruye el string)/positiva/dinamica  
Lo mismo que la 9

13)valid\_response\_with\_null\_data()  
Caja blanca(conoce la respuesta del servidor)/positiva/dinamica

## Pruebas Lite Client Integration Test

En todas se usa mock así que son pruebas simbólicas

- 1)create\_and\_delete\_group()  
Caja blanca(conoce las respuestas)/positiva
- 2)create\_and\_delete\_session()  
Caja blanca(conoce las respuestas)/positiva
- 3)create\_author()  
Caja blanca(conoce las respuestas)/positiva
- 4)create\_author\_with\_author\_mapper()  
Caja blanca(conoce las respuestas)/positiva
- 5)create\_group\_if\_not\_exists\_for\_and\_list\_all\_groups()  
Caja blanca(conoce las respuestas)/positiva
- 6)create\_group\_pads\_and\_list\_them()  
Caja blanca(conoce las respuestas)/positiva
- 7)create\_pad\_and\_chat\_about\_it()  
Caja blanca(conoce las respuestas)/positiva
- 8)create\_pad\_move\_and\_copy()  
Caja blanca(conoce las respuestas)/positiva
- 9)create\_pad\_set\_and\_get\_content()  
Caja blanca(conoce las respuestas)/positiva
- 10)create\_pads\_and\_list\_them()  
Caja blanca(conoce las respuestas)/positiva
- 11)validate\_token()  
Caja negra/positiva

En este apartado no se hace ninguna prueba negativa.  
No se prueban todos los tipos de constructores de la clase Client.

## Conclusion

La cobertura que se propone es por de instrucción, por lo cual, no se prueban muchos caminos de entrada en los ifs y el resto de bucles. En las pruebas de integración no se realiza ninguna negativa por lo que puede ser que se realicen operaciones que quizá no deberían y desconozcamos estos fallos, tampoco se prueban los constructores con distintos parametros.