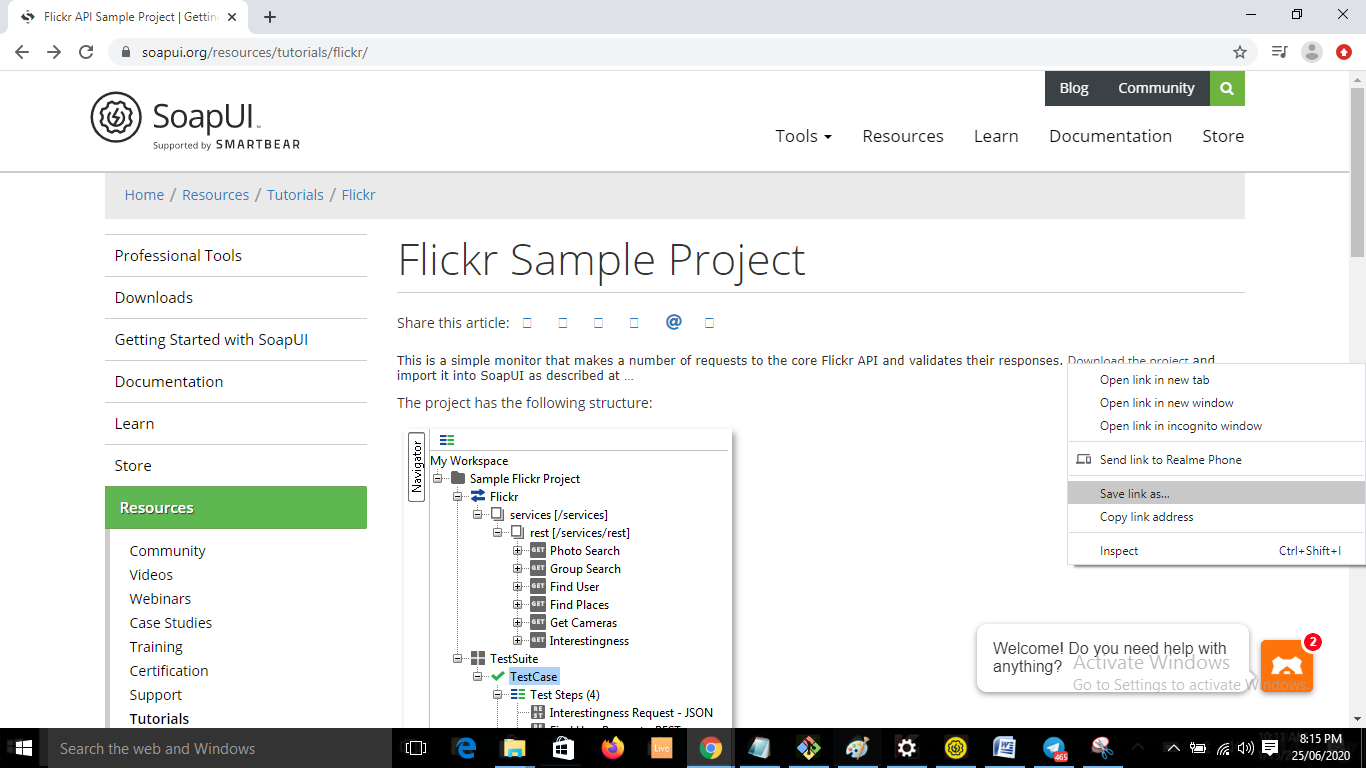
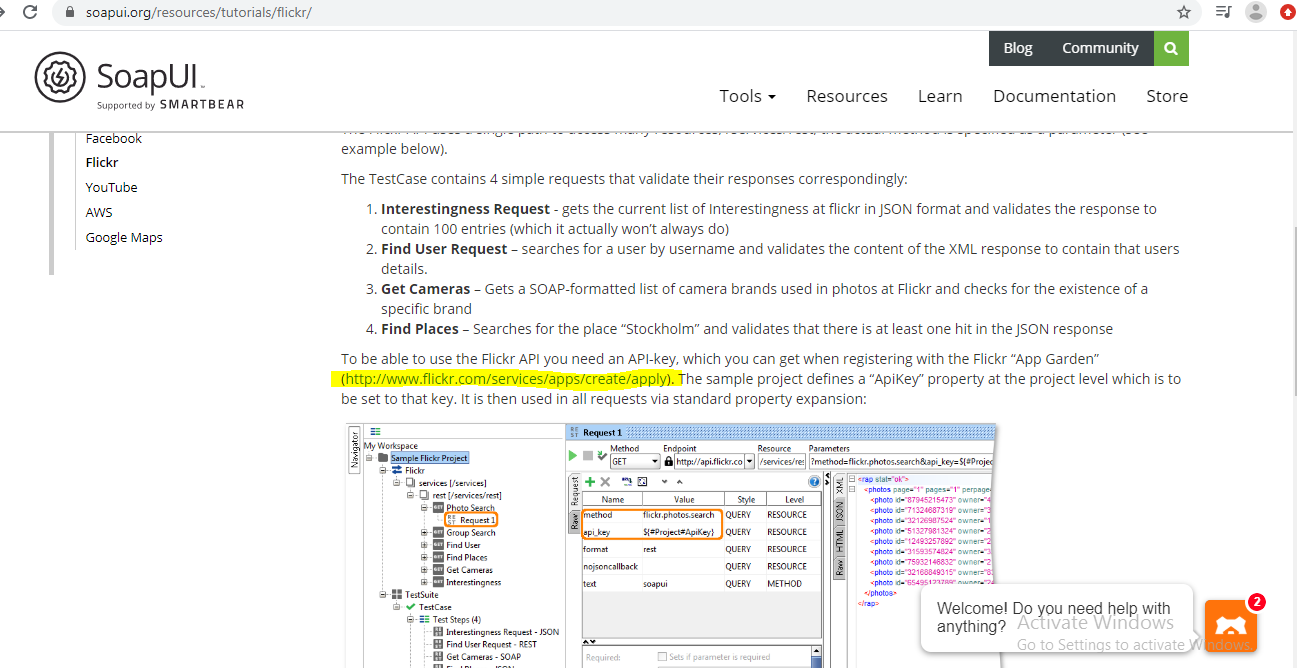
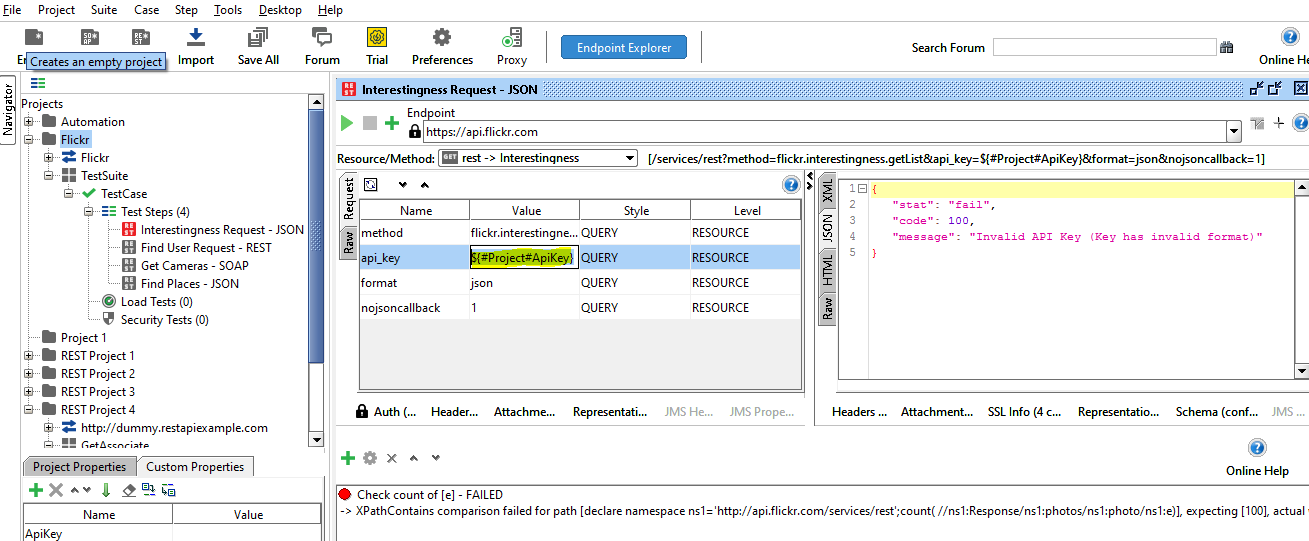
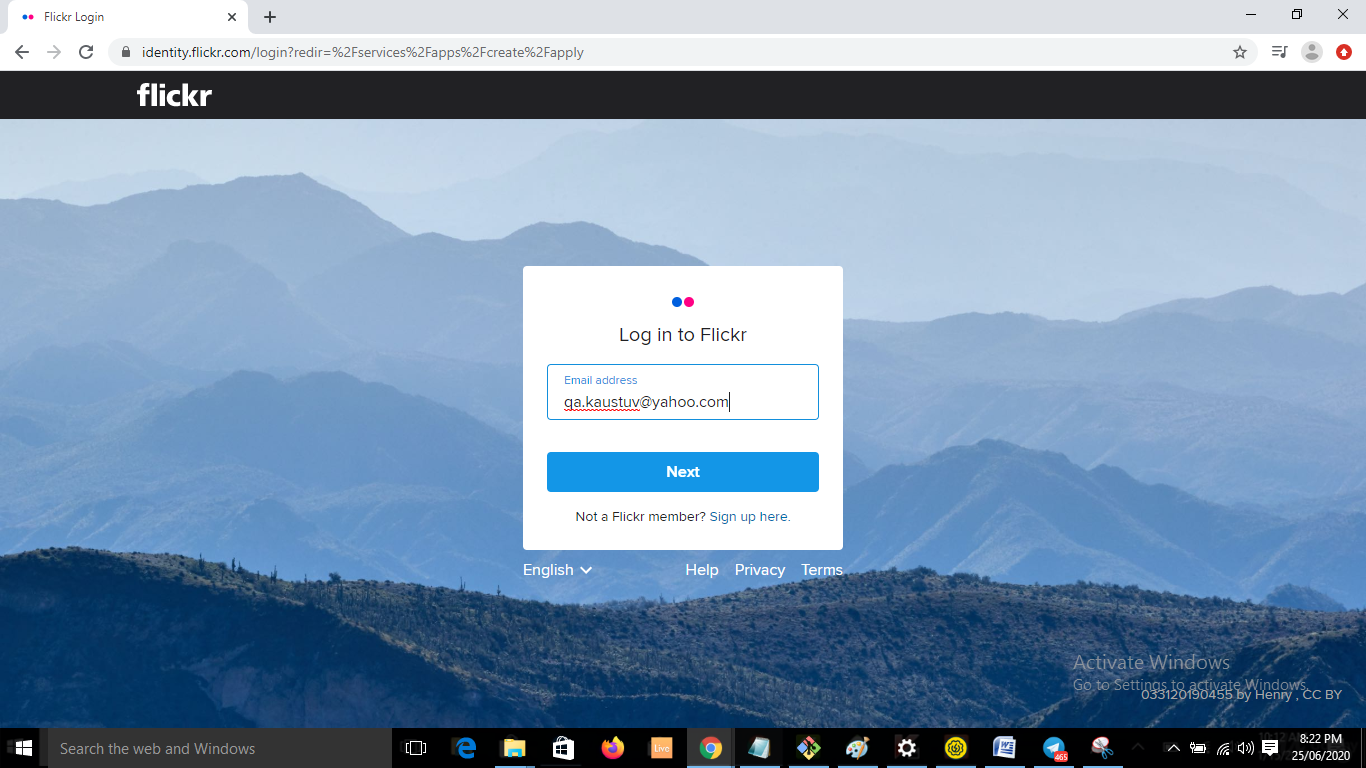
**43. Flicker API Introduction**

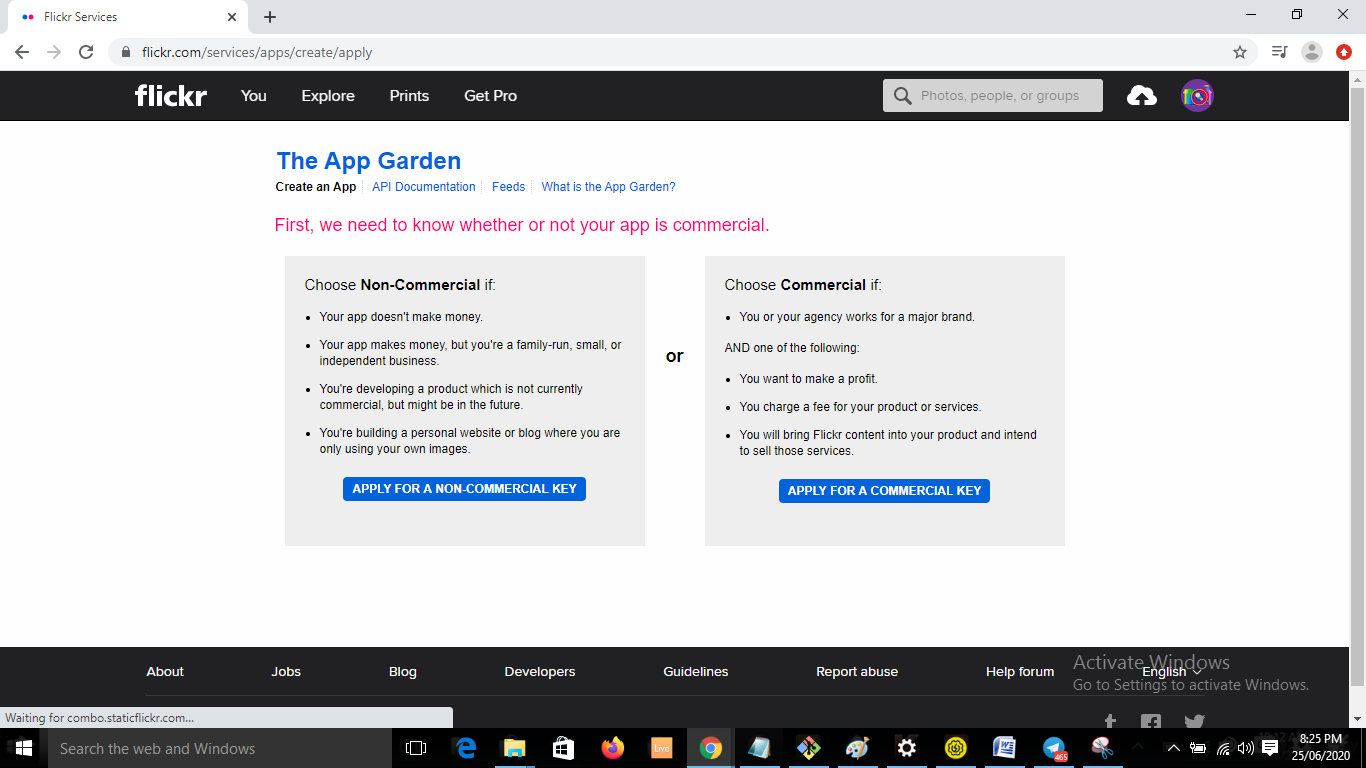
* Go to this: <https://www.soapui.org/resources/tutorials/flickr/>
* Download the project.

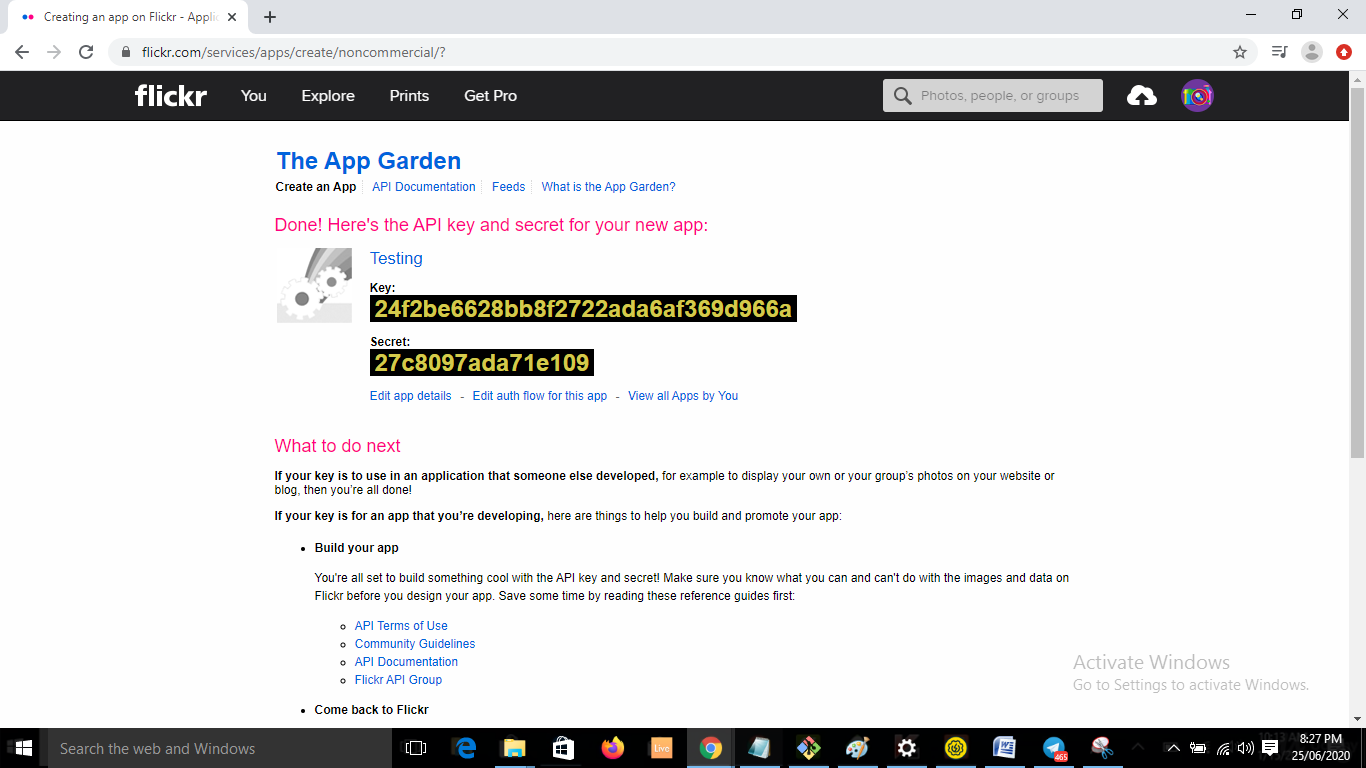
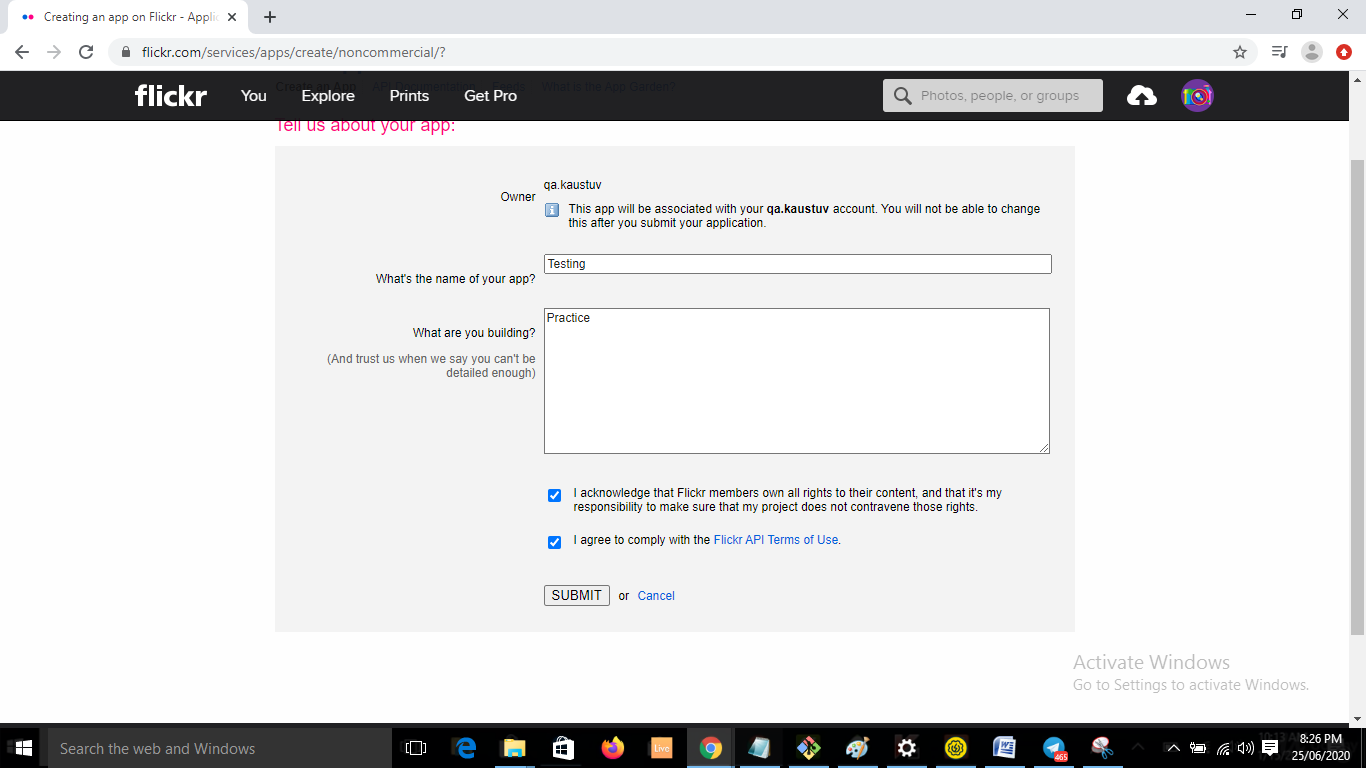


* Import the project.
* You need to provide the api key.

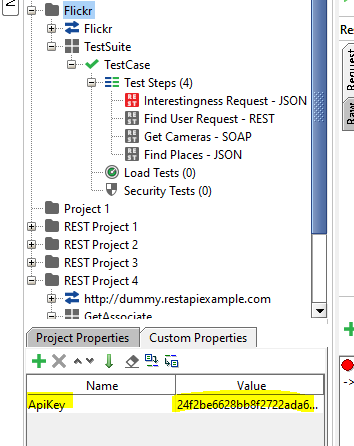


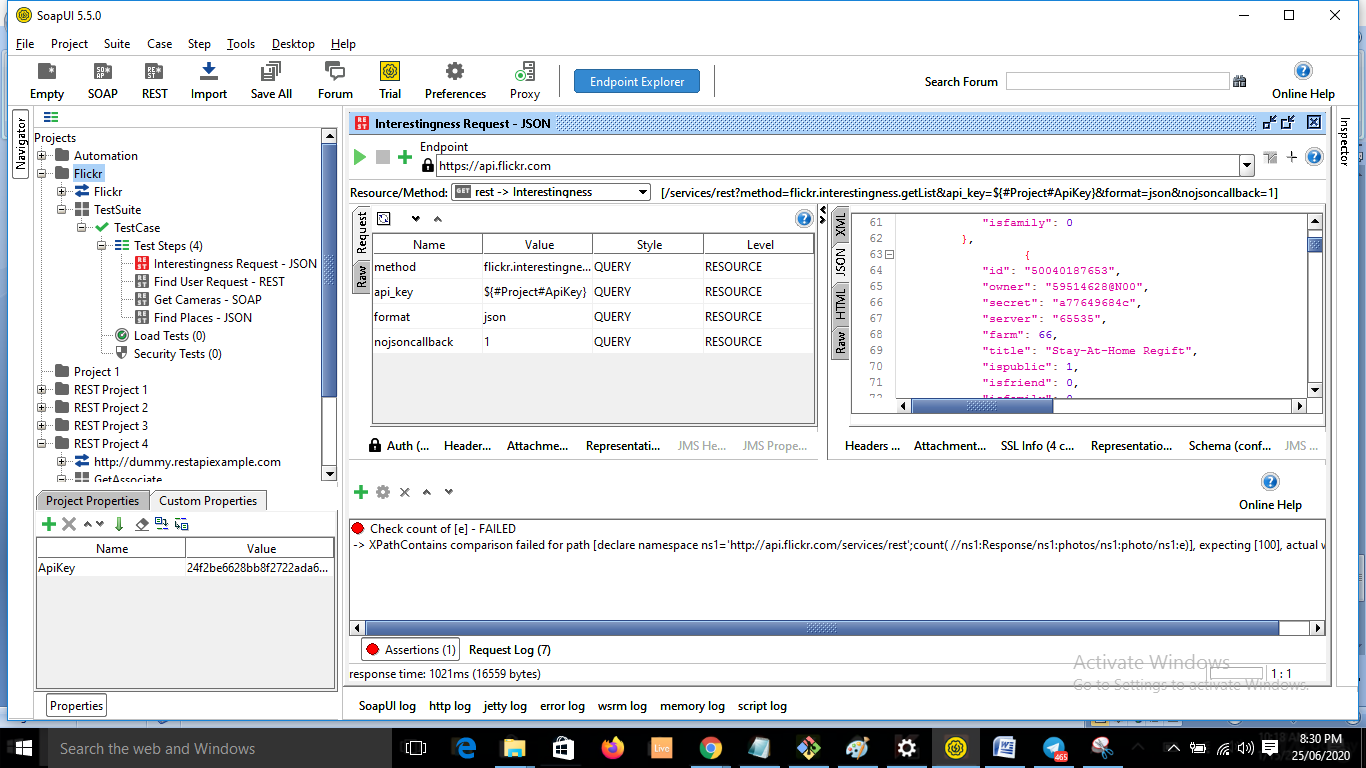




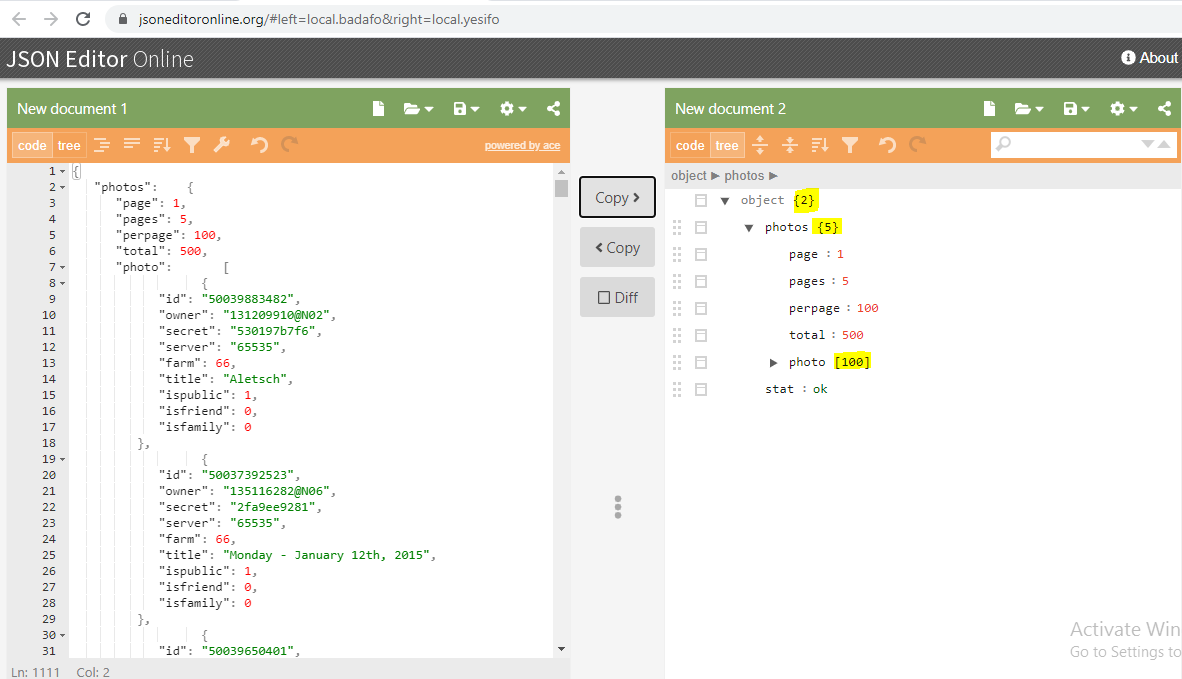


24f2be6628bb8f2722ada6af369d966a



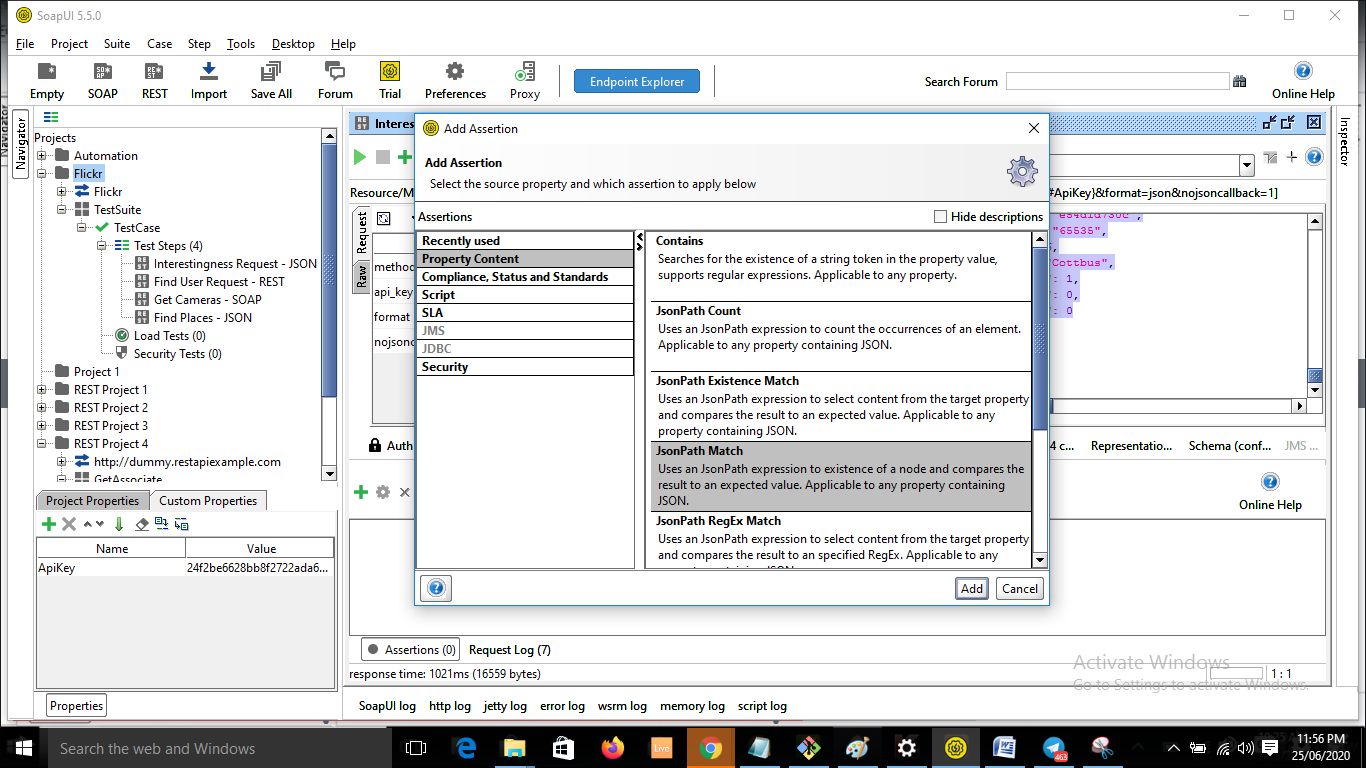


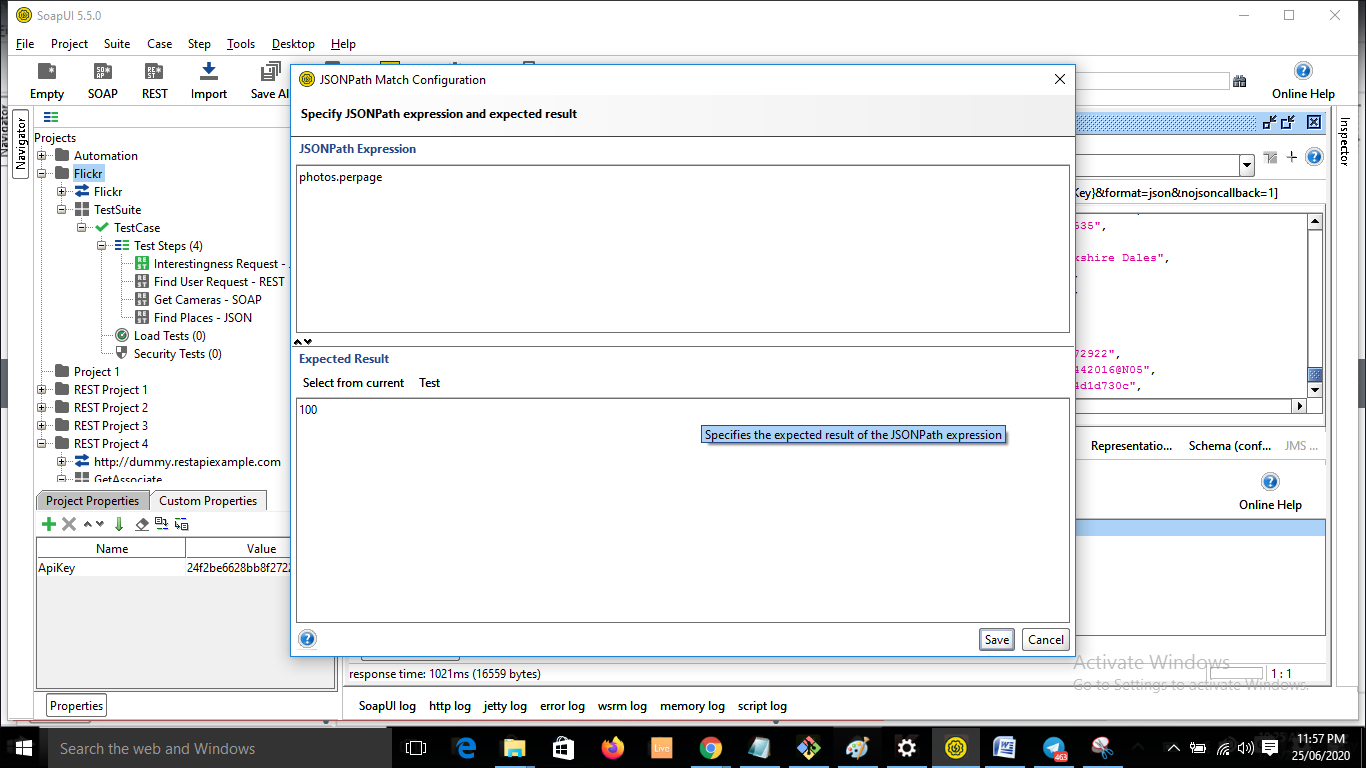
**44. Json Path extraction from responses**

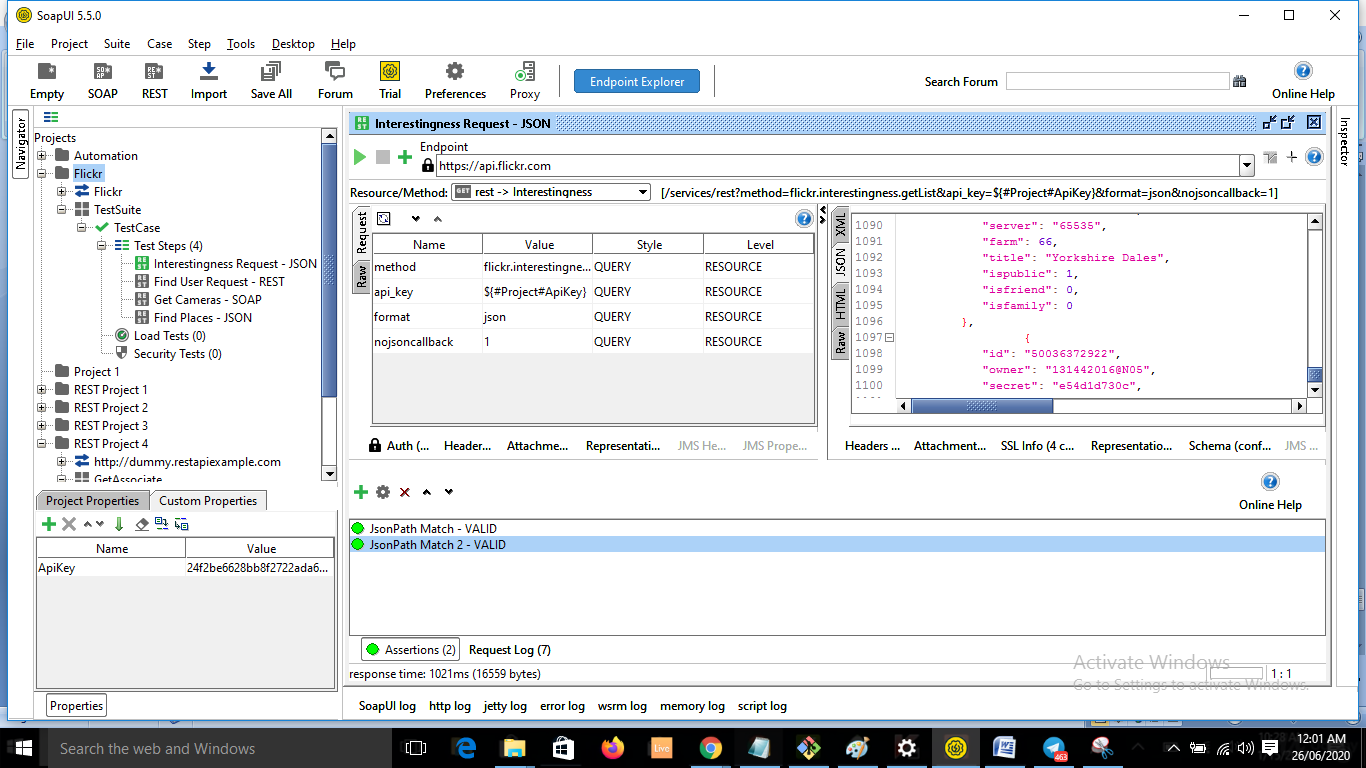
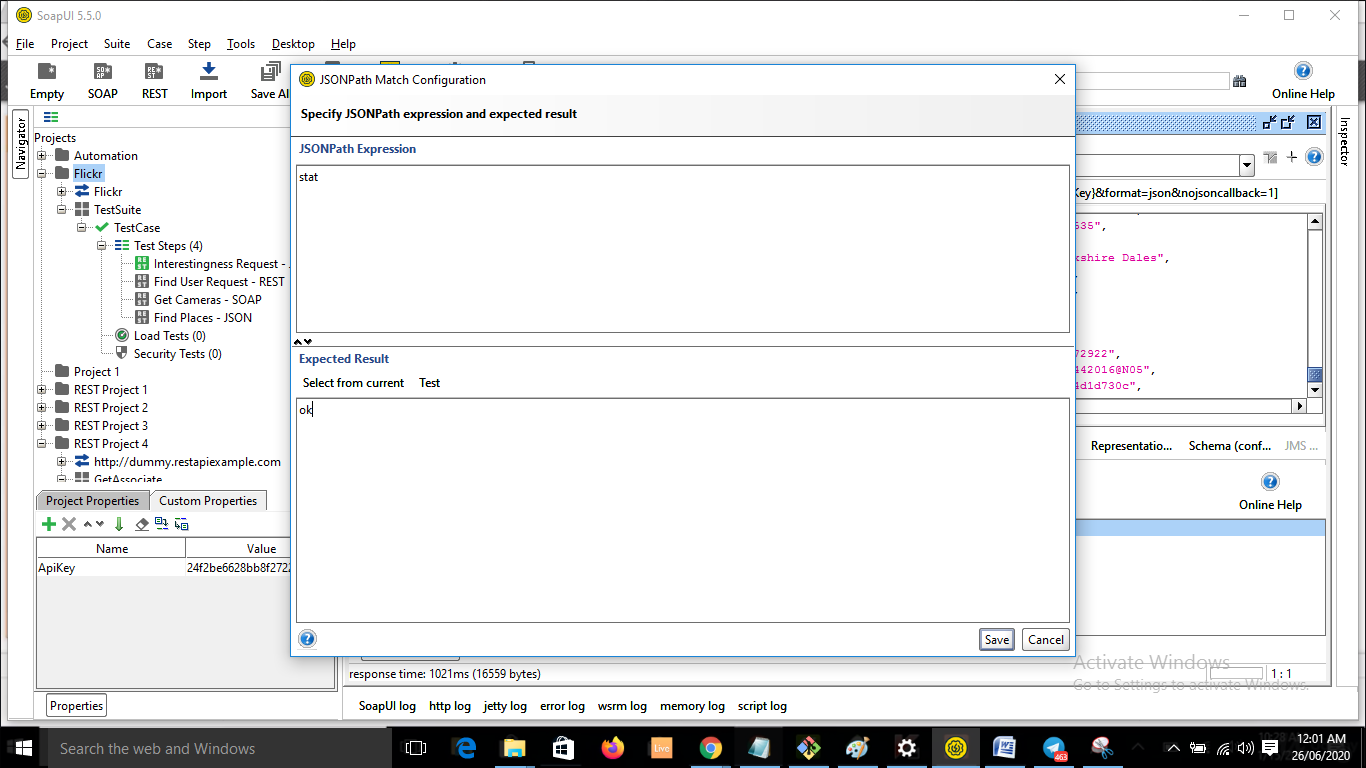
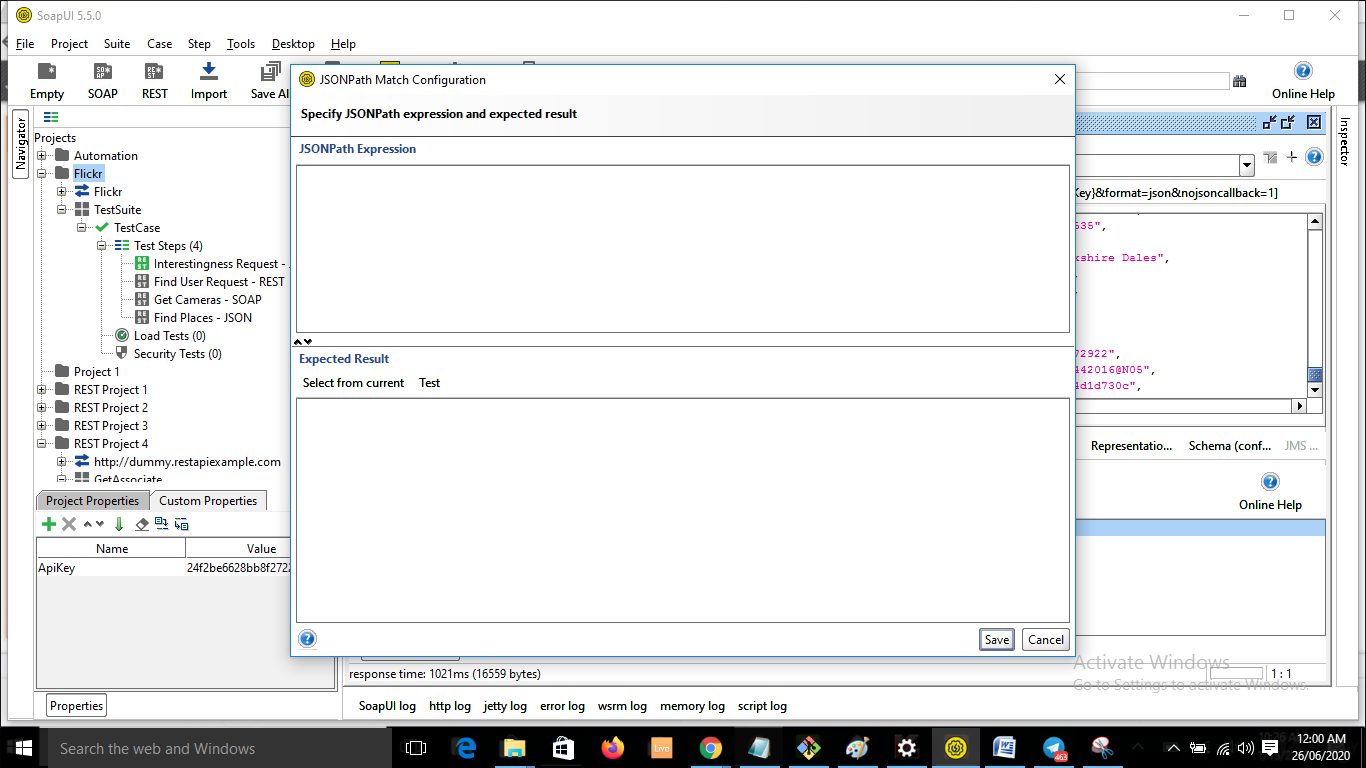
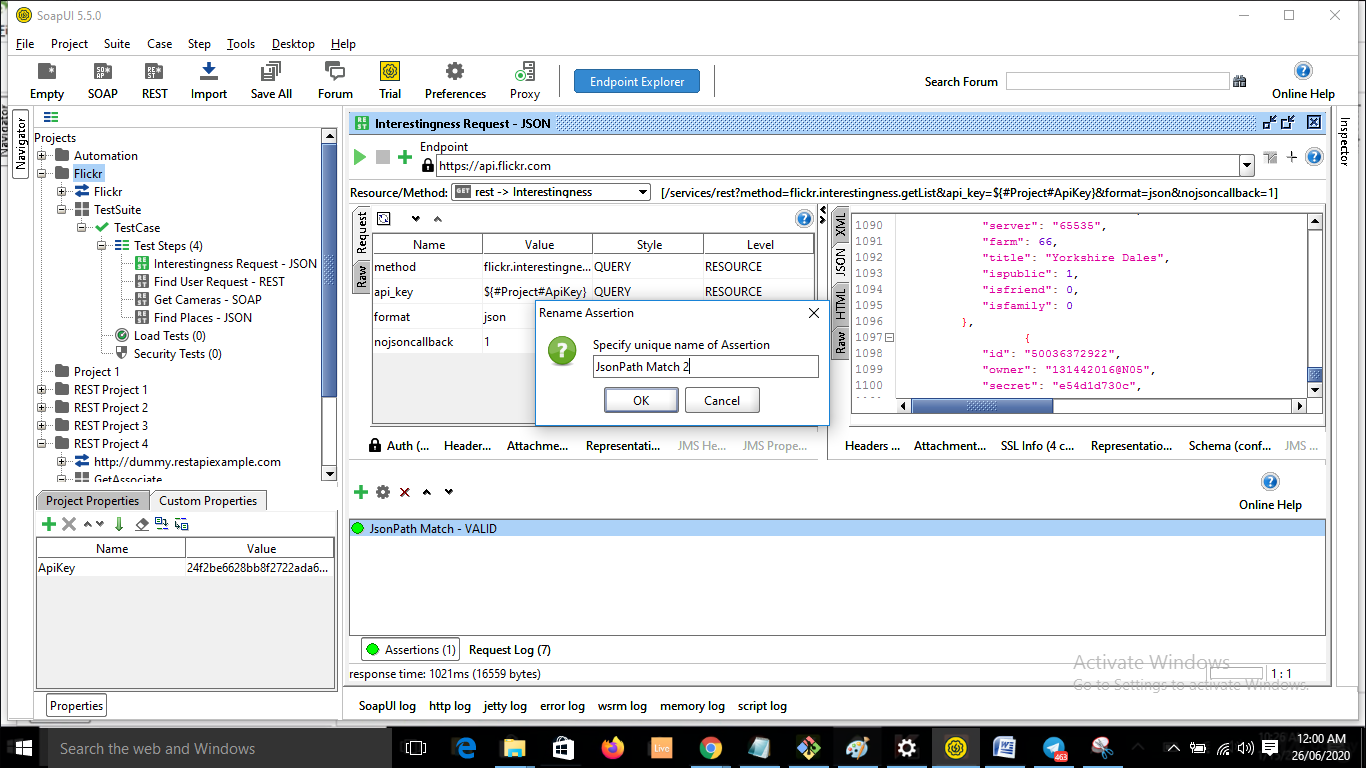
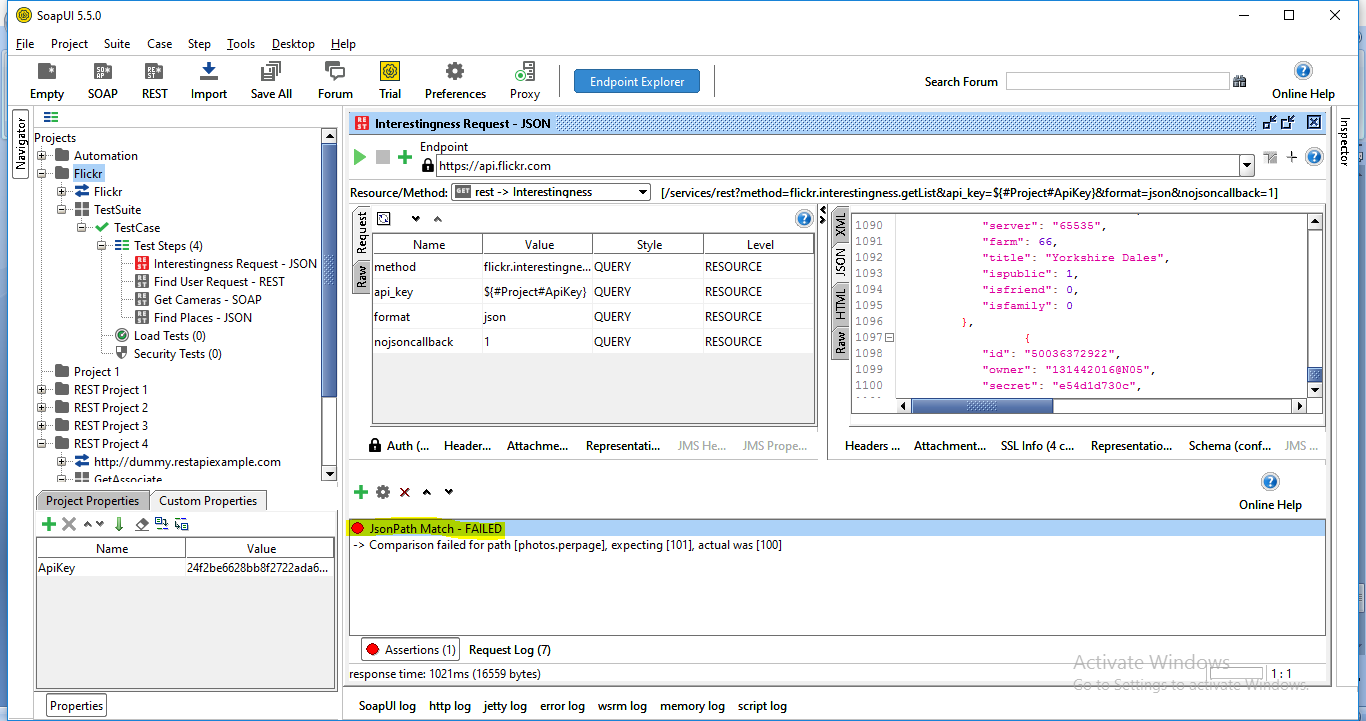
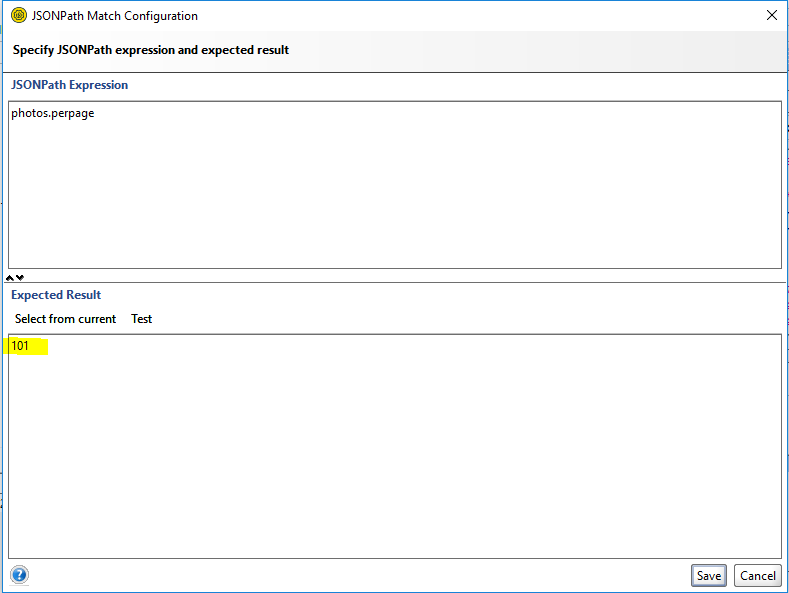
****

* {}🡪Objects

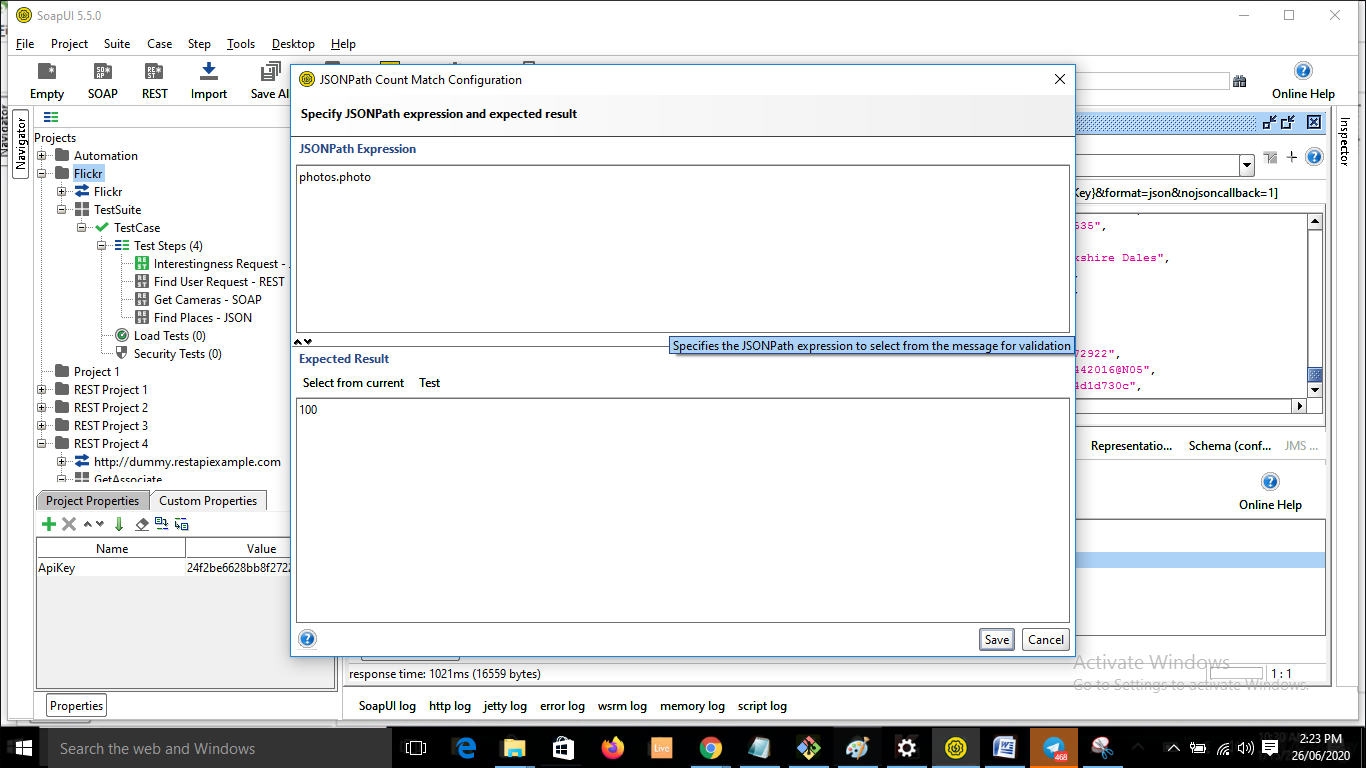
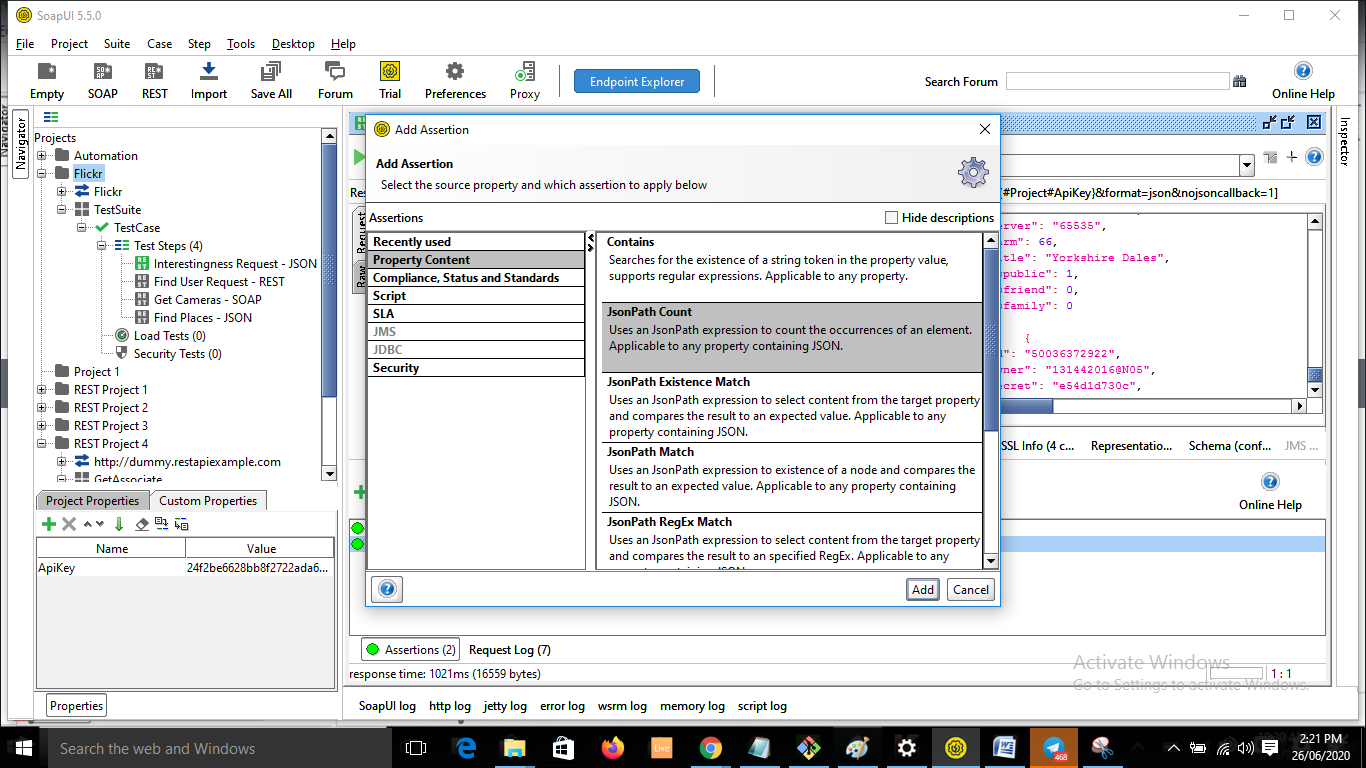
[] 🡪 Arrays/Indexes

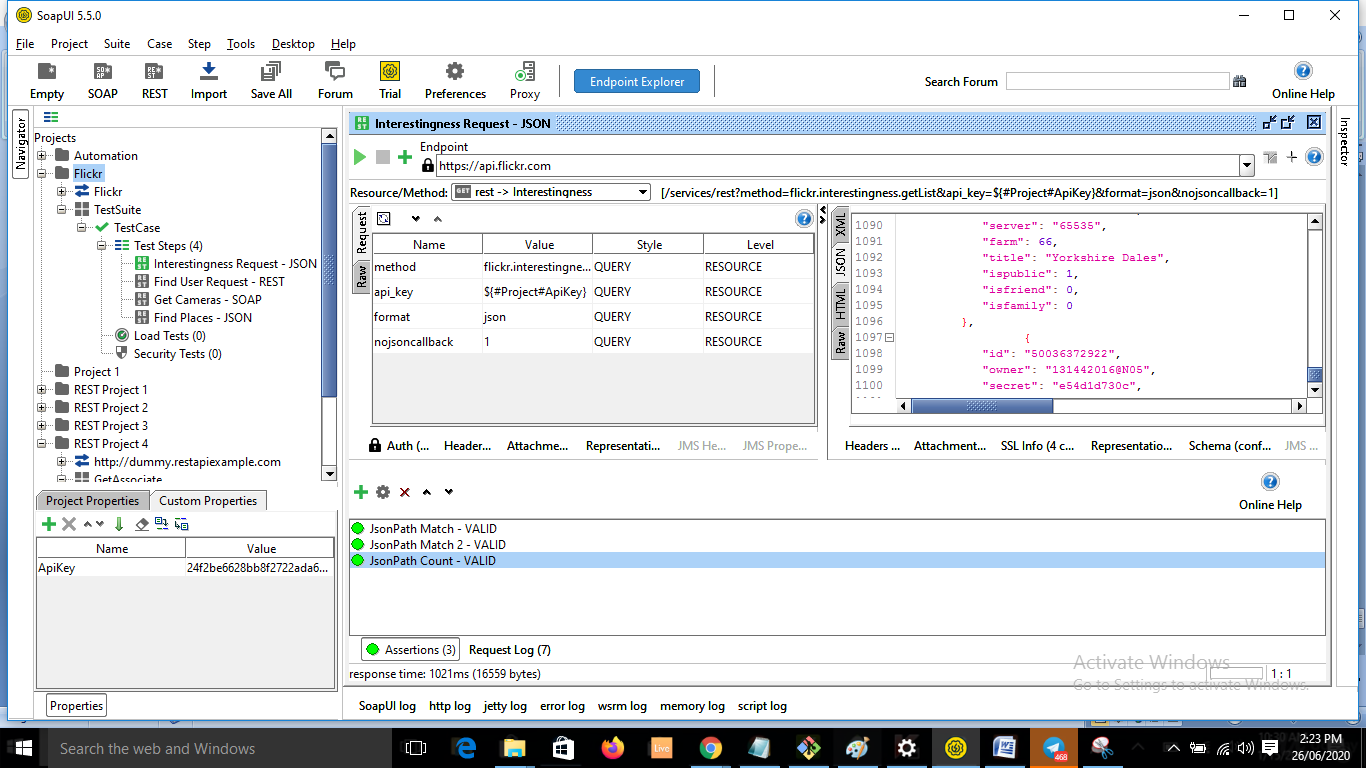


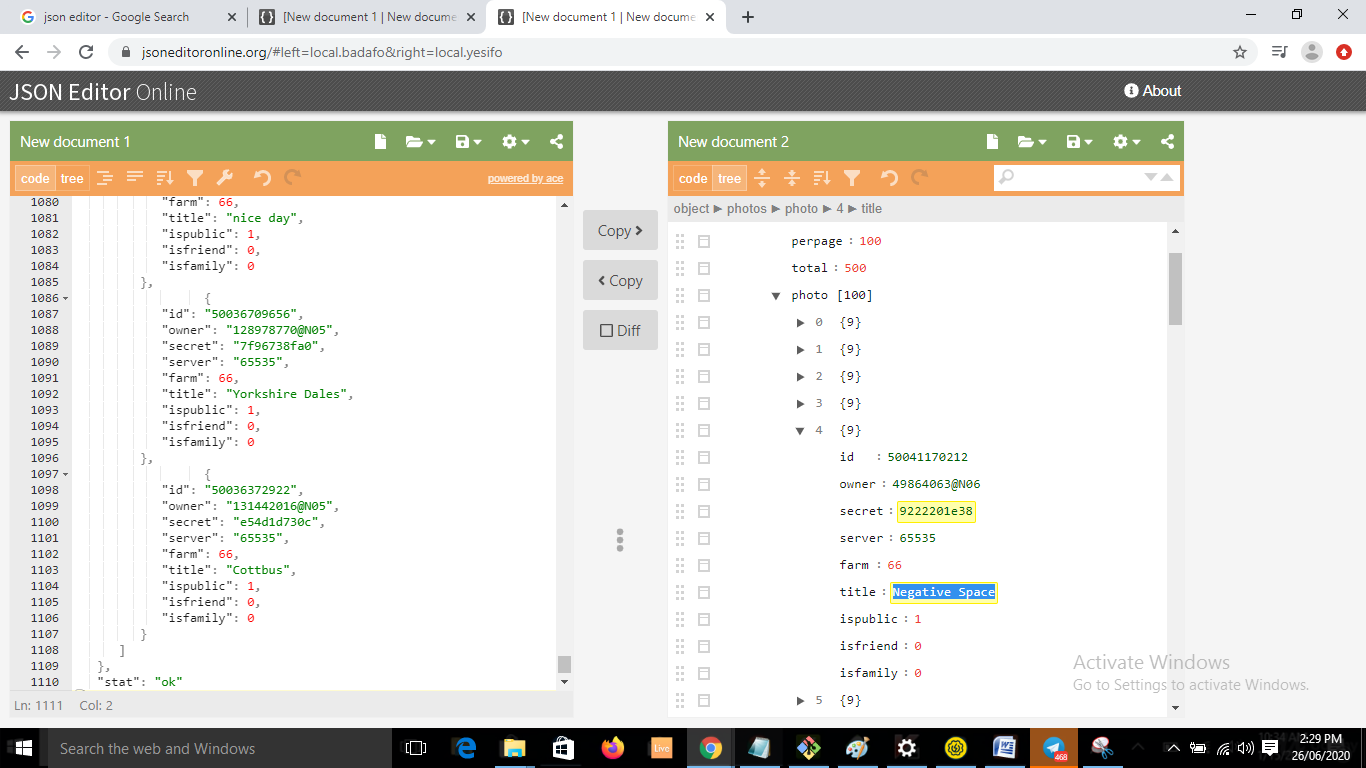


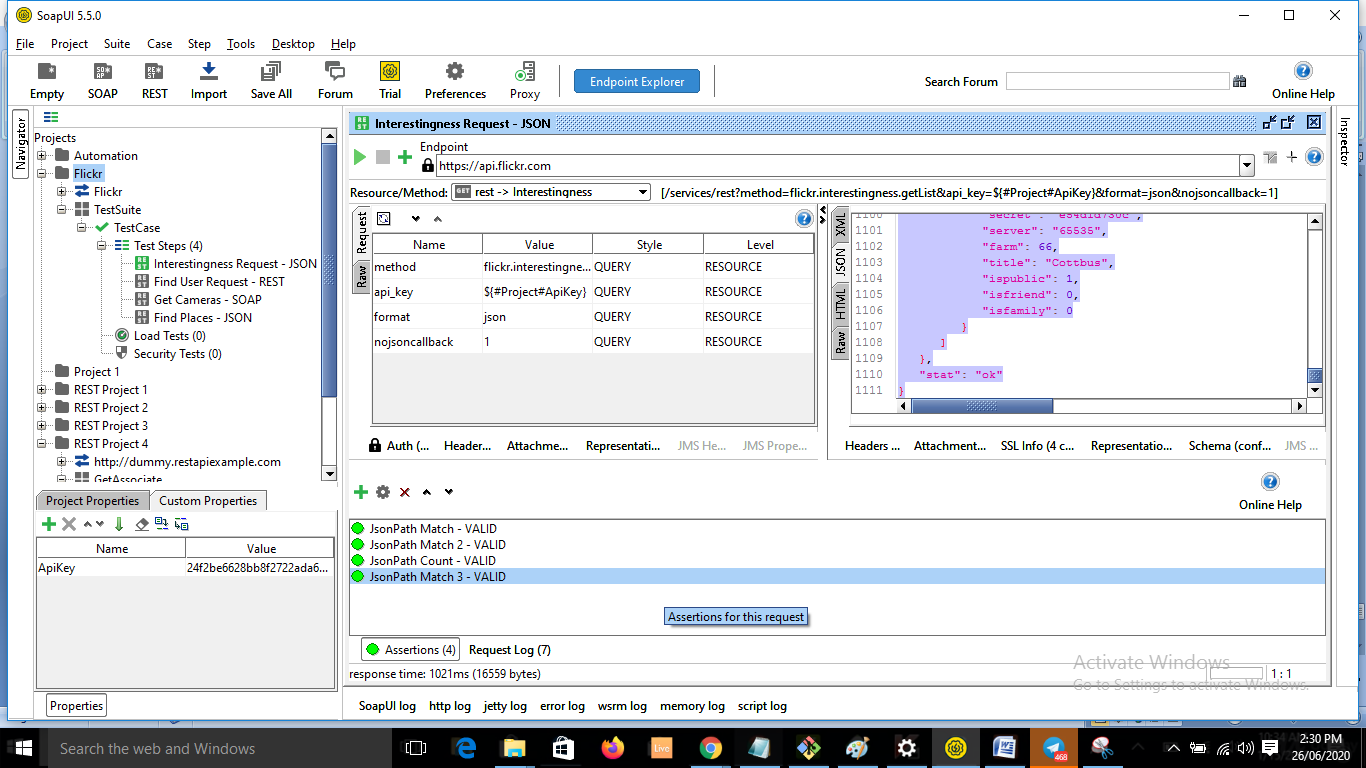
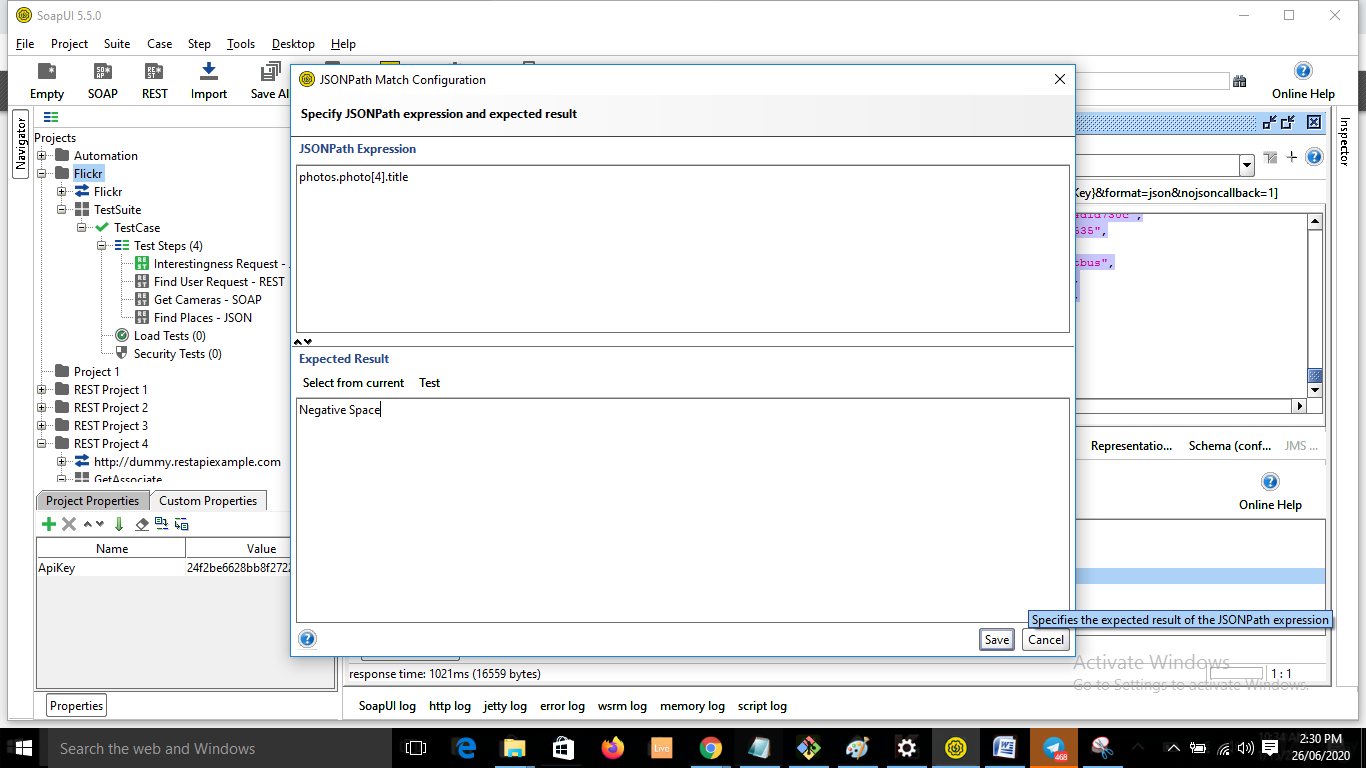
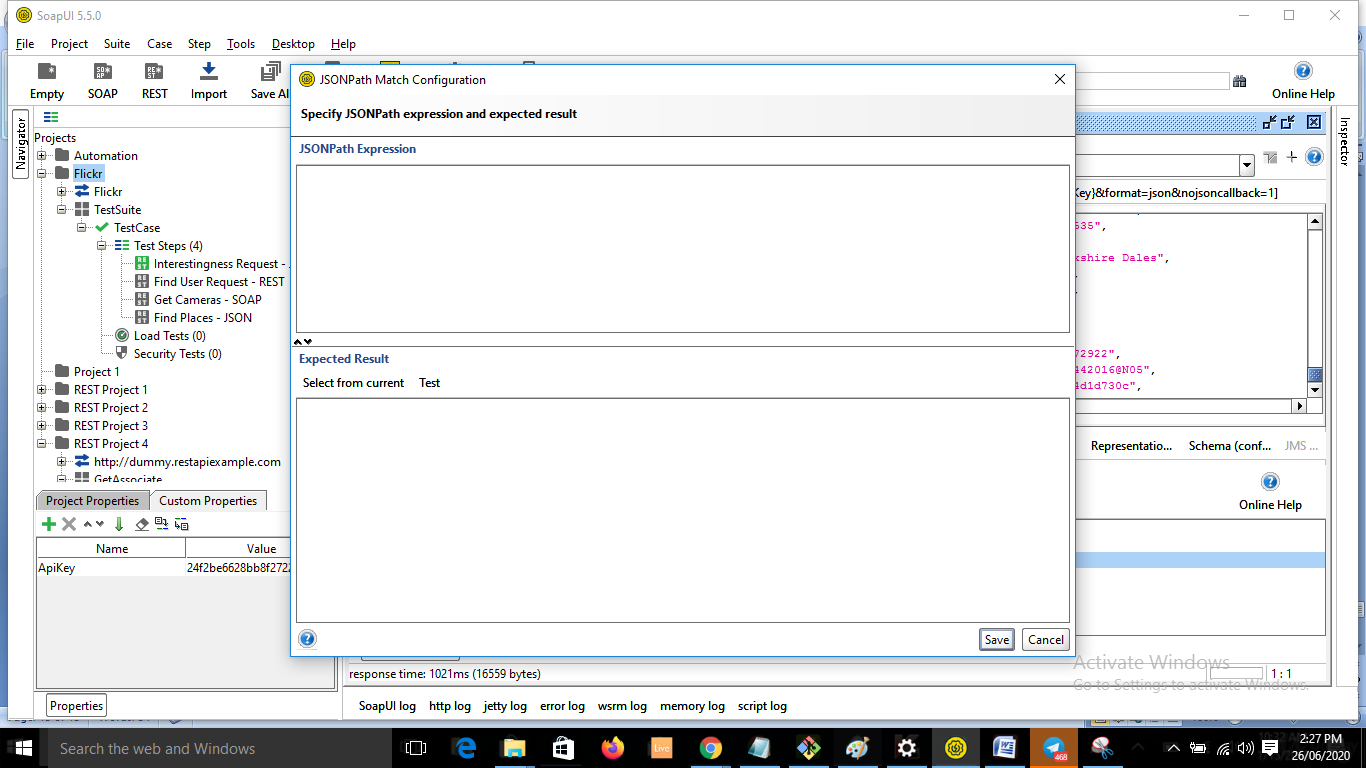
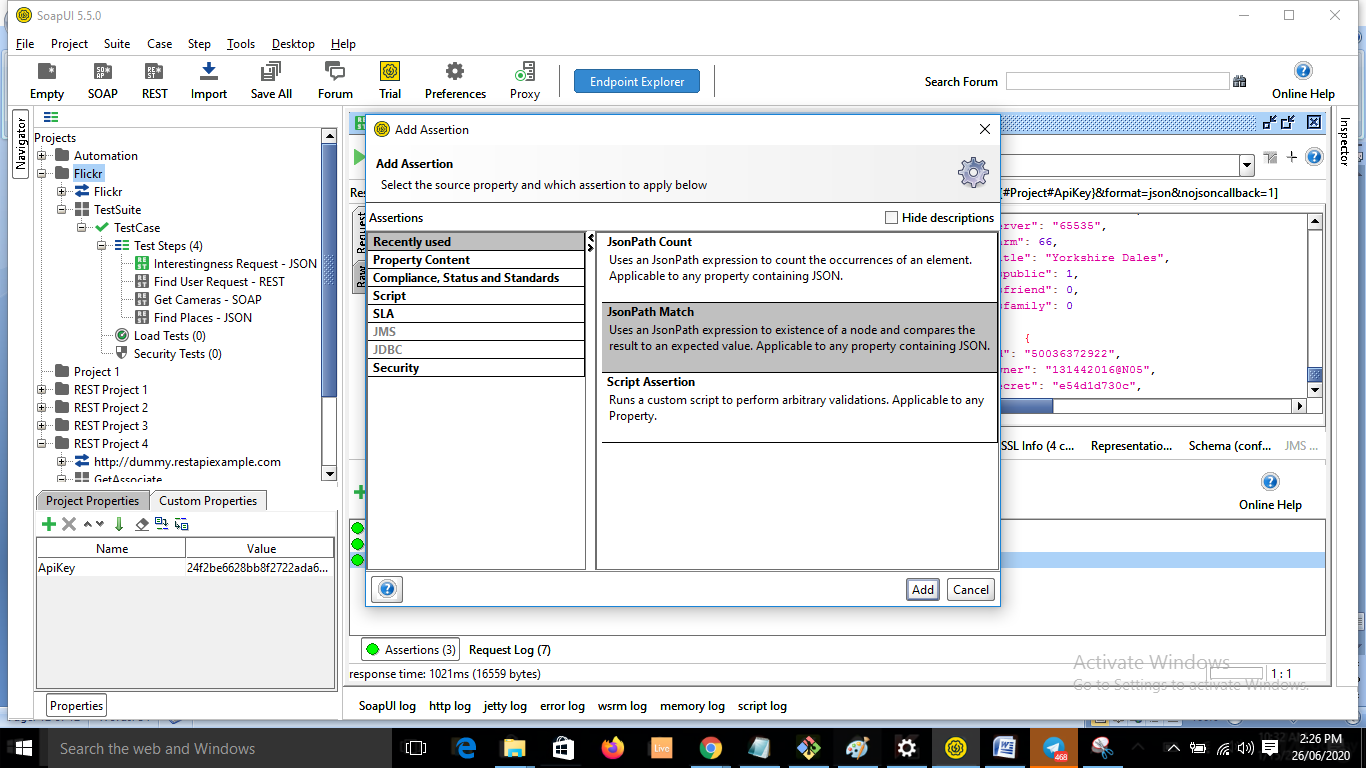


**45. Json Assertions on parsed responses**

* Json Path Count Assertion: 

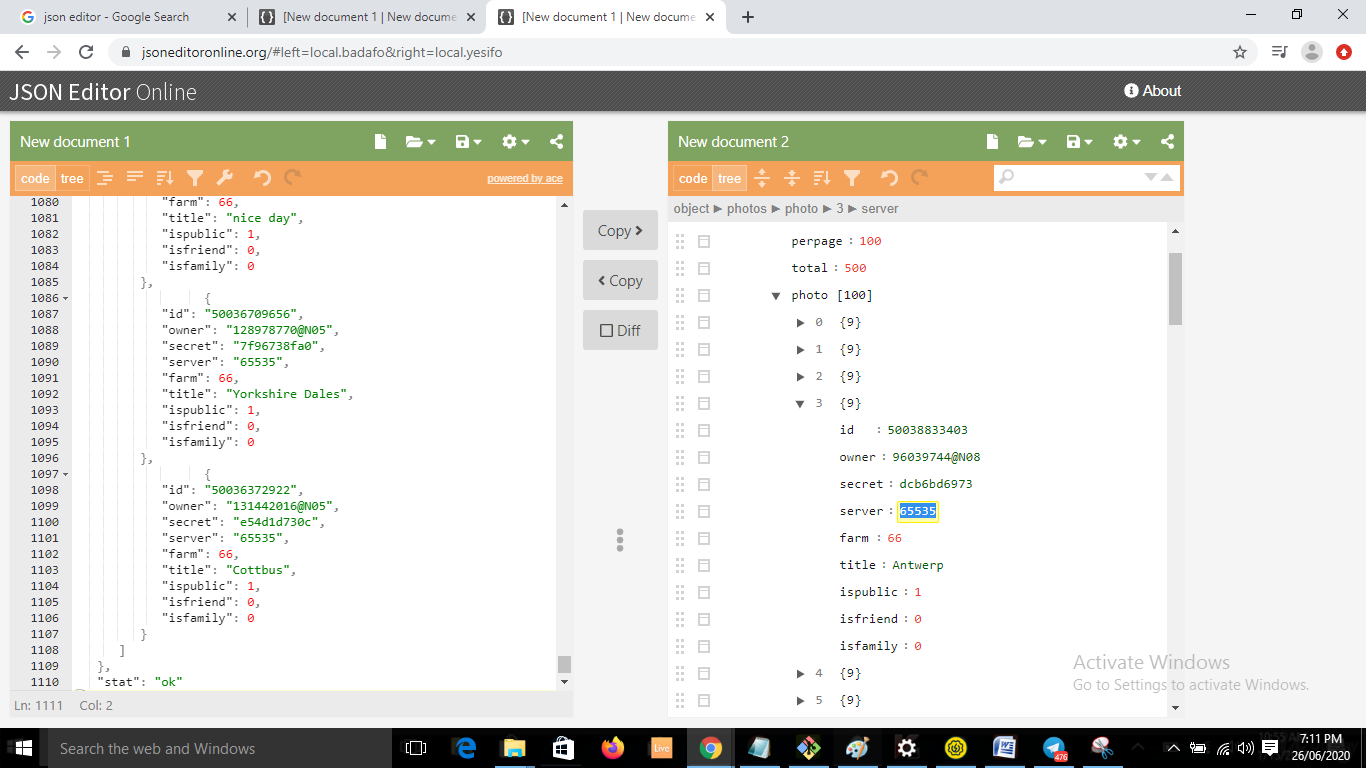


* Check name is Negative Space. 

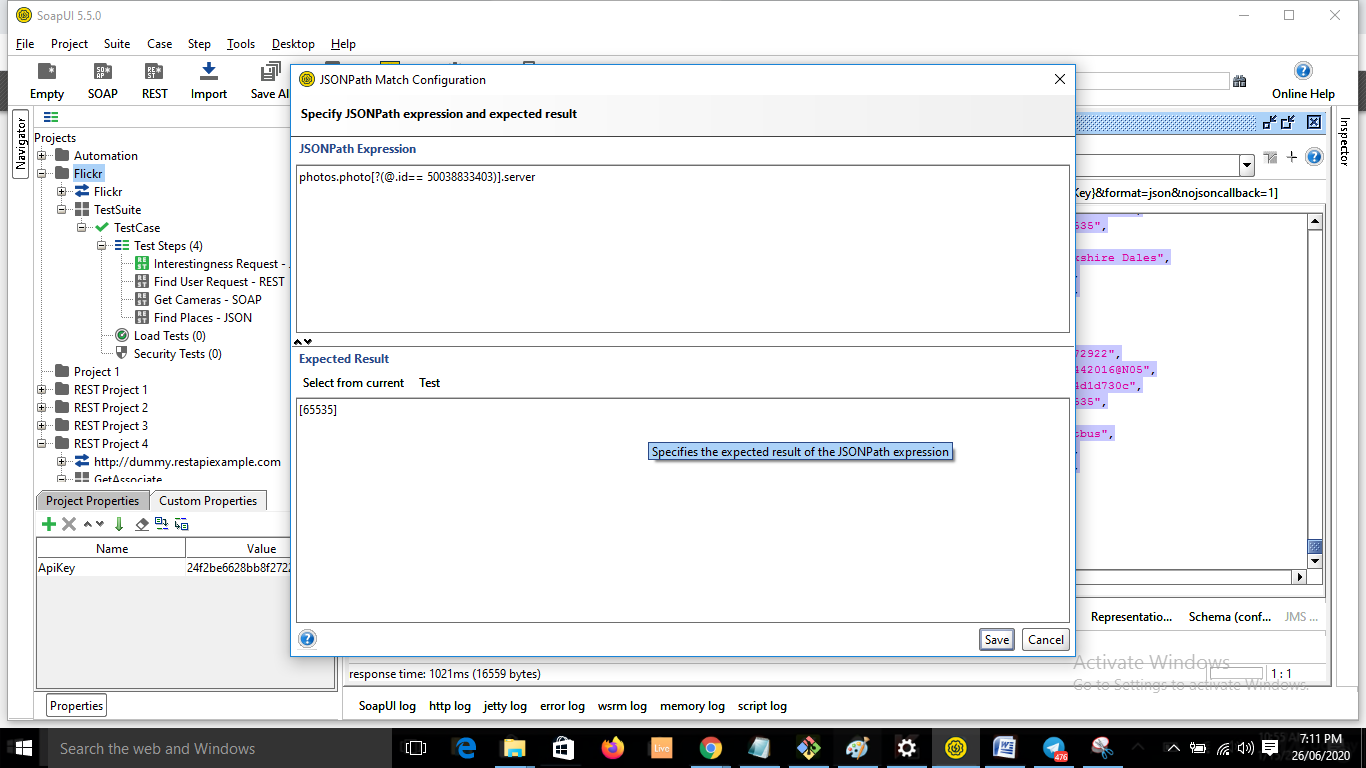


**46. Advance example on Json Assertions**

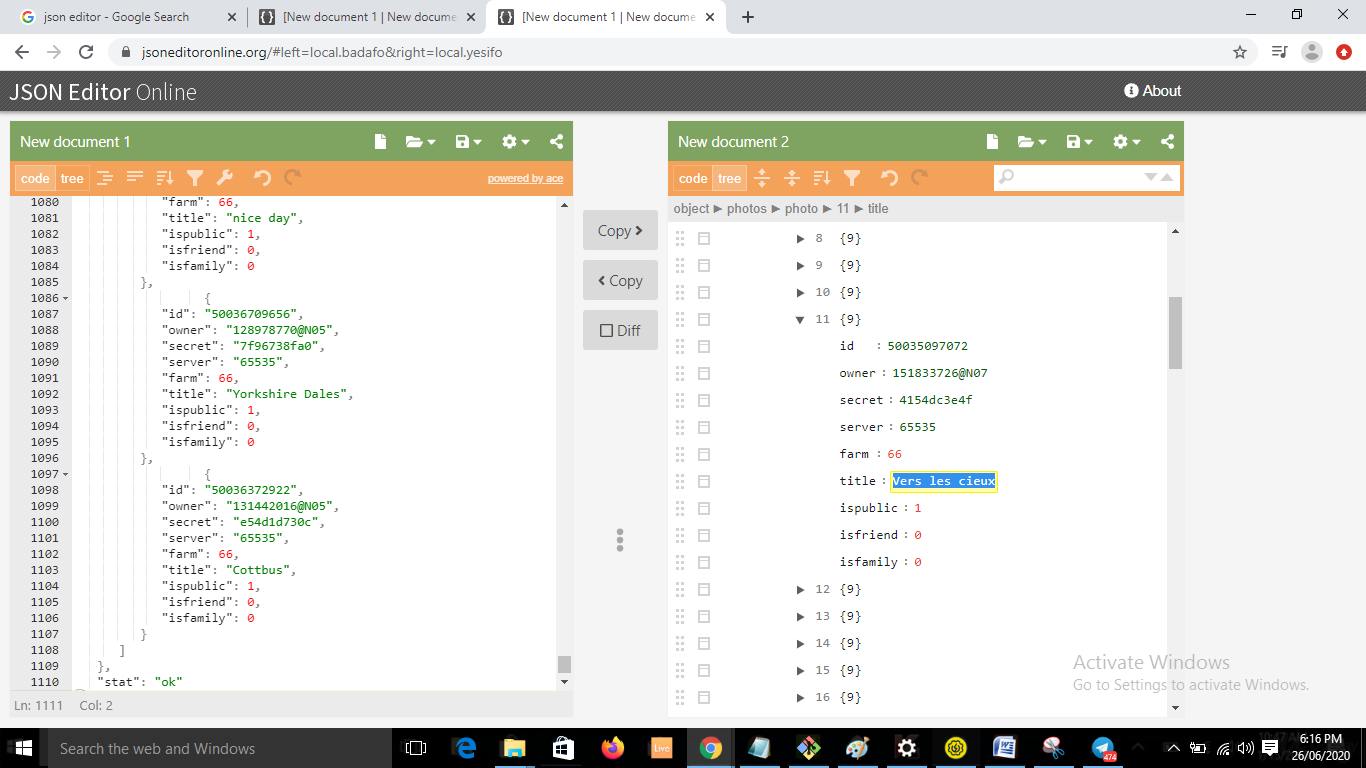
* Grab the server value with the id 50038833403



photos.photo[?(@.id== 50038833403)].server



* Or grab the id value with title called Vers les cieux

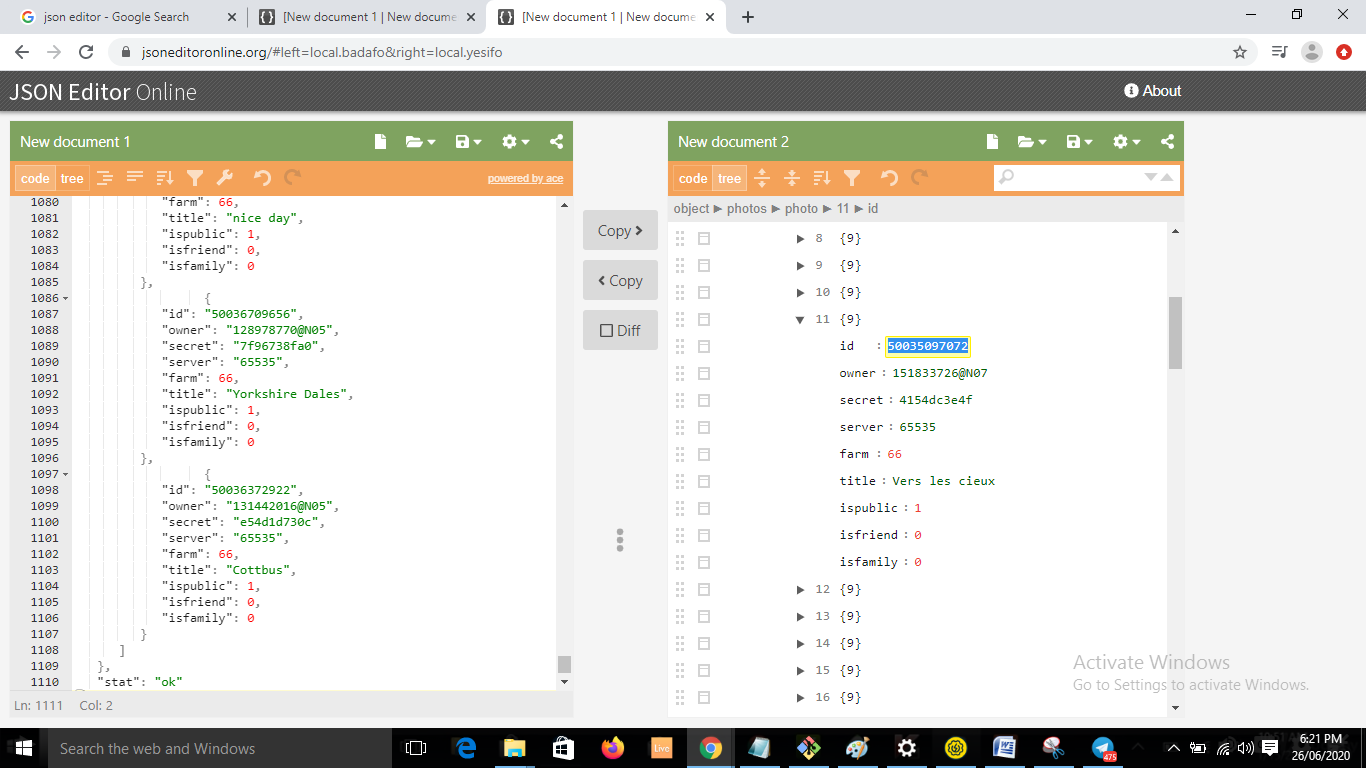


* Syntex for this:

Photos.photo[?(@.object==’value’)].object

‘?’ stands for regular expression.

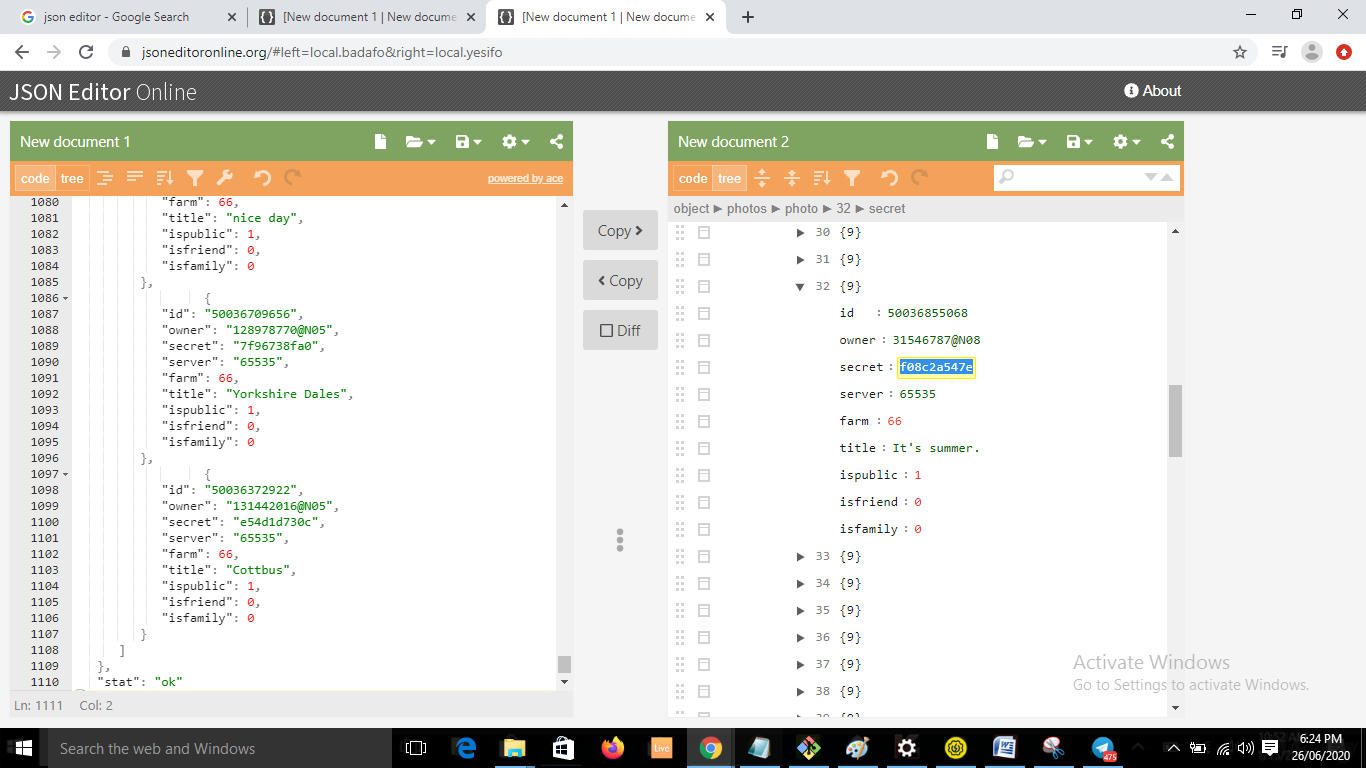
For us it will be:



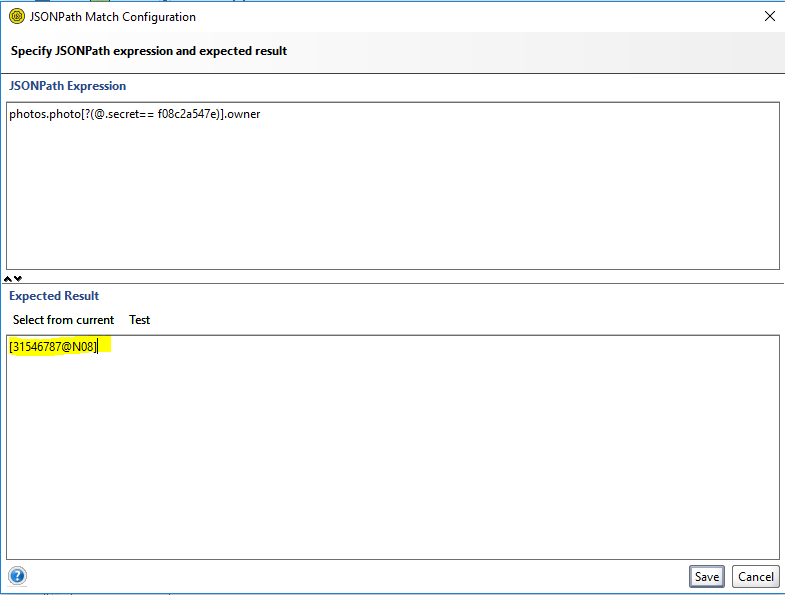
photos.photo[?(@.id==50035097072)].server

It will scan all ids and reach to id=50035097072. After that it will pick the server.

* Pull the owner who has secret value=f08c2a547e.



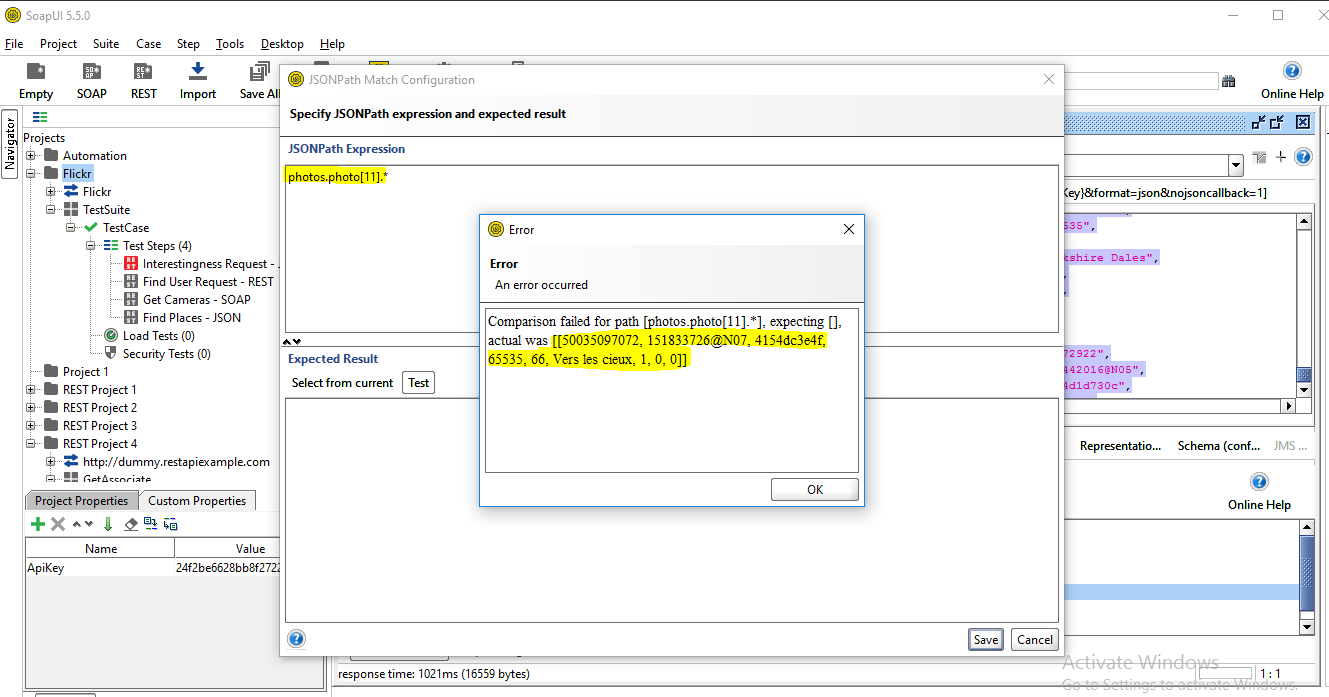
Photos.photo[?(@.secret== f08c2a547e)].owner

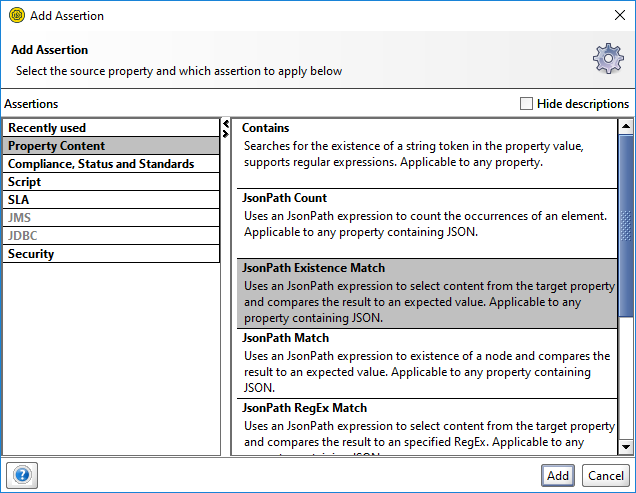


We need to give [] for regular expressions.



* Ie. Instead of hard coding you can find our objects using sibling object if you know the value.
* To get all the values:





* It will check whether Path is there.
* To check perpage exists or not:

