Flask application for XML conversion

Create a simple Flask application with one endpoint /export that will convert data from the XML downloaded from Rossum to a XML in another format. The endpoint should:

- Validate that the request came from an authorized source, using Basic Authentication. The right credentials should be stored as environment variables in the environment that your application can access. The username shall be *myUser123*, password *secretSecret*.
- Accept the user request where the user passes the ID of the annotation they want to get the data for and the queue ID of the annotation where the annotation is located in Rossum. The way how the users shall pass this information is up to you. Validate that the request format is correct and it contains everything necessary.
- Within the endpoint, use the passed values to download data for a sample annotation in your Rossum account, using a call on /export endpoint of [Rossum API](). ([Create a new Rossum account here.]() To create the Rossum account, you can use some temporary email, for example from [this website]().) The content of the response will be similar to [this]().)
- Convert the data that you have received, so that the output data has [the following format]().
- Send the converted content as a REST API POST request to dummy URL [https://my-little-endpoint.ok/rossum](), with no authentication details and data as JSON with the keys *annotationId* and the value of the original annotation ID and the key *content* with the data in XML as a base64 string.
- Return a JSON with a key *success*, filled by *true* or *false*, depending on whether the request could be posted successfully or not.

Add Dockerfile to build your application and create one test that will test the whole pipeline for the scenario where everything goes without issues. Push your solution to a Git repository of your choice where Rossum can access the code.