Unit tests have found some bugs in the system. Mainly focus on the ‘relation’ part.

1. Description and impact

In the GET method of ‘/todos’, the user could add the URL parameters on the request. The result would be a filtered data items. But when the team uses the non-existing parameter like ‘titles’, the result should be an empty list or the error message to tell us the parameter is wrong, but we received a list without any filtration. If this bug exists in the system, it could not handle the typo made by the user correctly

Then, In the GET method of ‘/todos/{id}/categories’, the user would receive the categories that todos with id have a relation with. But if we use a non-existing id in this request, the result should be an error message with error code 404 or 400. But the actual result is 2 copies of categories information. This mistake exists on every relation-related API such as ‘/todos/{id}/categories’,’/projects/{id}/categories. If this bug exists in the system, the user could be confused that why there are so many objects in the backend, even he does not create

Also, In the post method of ‘/todos/{id}/tasksof’. The user could add relationships between todos and projects by sending the id of the project. When no id is posted, the server should do nothing. But in fact, the server creates a project automatically with the default value and add a relationship between todo with id and the new project. If this bug exists in the system, the user might create a lot of project objects in the backend even without knowing it

1. reproduction

For the first bug, after starting the server, the user could send a GET request of ‘/todos’ API with a URL parameter {‘titles’: ’scan paperwork’} to the server, and retrieve the response body, he would find that it could still receive results even there is no such a field.

For the second bug, after starting the server, the user could send a GET request of ‘/projects/10/categories’ API to the server, and retrieve the response body. He could find that it receives all the categories information two times.

For the third bug, after starting the server, the user could send a POST request of ‘/todos/1/tasksof’ with nothing, and retrieve the response body. He could find that it is a new created project. And send a GET request of ‘/todos/1’, he could find that a new relation has been add.