# To TopTal screeners:

## Administration notes

1. In my eyes, one of the most important elements of the development process is the constant communication between the developers (team) and the side defining the requirements (the client or the product team). I avoided any professional questions or communication, since it's a screening process. It's important to me to emphasize that this should NOT be the normal way of managing the process.

## Technical notes

1. I am using .Net Core for my solution.
2. I did not secure the connection string in appsettings so that it could be changed by you. The best practice in this case is to secure it (for example by using the secrets encryption mechanism).
3. I did not apply any concurrency or thread-safety model in the solution. Even though it was not mentioned in the requirements, it should be an unsaid assumption. Nevertheless, I chose to Keep-It-Simple-Stupid. I usually use the Optimistic Locking mechanism.
4. I did not use any logging in my solution. It also does NOT reflect a real project.
5. I would probably make a few changes, given the time:
   1. Use less tables (possible with 2 instead of 5)
   2. Implement a bulk post to create multiple resources
6. The proper way to implement query filters in .Net Core is through use of IActionFilter. But in this case the requirements are very basic and were targeted only to one method. I decided to implement the filters in one switch and not in a generic way. The advantage is that the code is more readable.
7. The proper way to implement authorization in .Net Core is through Policy mechanism. I did not use the mechanism (time constraints).
8. The proper way to implement tests (unit tests, component tests, integration tests, end-to-end tests) is through designated test projects with mocking and assertion support. Hopefully I will have time to start this project at least. An alternative: saving complete test scenarios from Postman.
9. In very few cases of error (BadRequest or Conflict for example), I returned simple text or an error structure in response. Since this is an API server, the proper way to do it is to use unique error response json and fill it with as much information as necessary (time constraints).