¡Felicitaciones!

Si estás leyendo esto, es porque llegaste a una estapa muy importante de nuestro proceso de selección.
Te invitamos a desarrollar nuestro Challenge Técnico para la posición de **Tech Lead Back End.**

¿Por qué esta etapa es importante?
Porque nos ayuda a realizar la próxima etapa (entrevista técnica) con mayor objetividad, pero principalmente nos aporta información muy valiosa sobre tus hard skills.

¿Cuánto tiempo tengo para realizar el Challenge?
Tienes 5 días corridos para realizarlo. Está pensado para invertir una hora al día, considerando que también tienes otras responsabilidades laborales como personales.

¿Qué sucede si no reaalizo el Challenge?
Lamentablemente, no podremos continuar con el proceso, ya que se trata de una instancia de las más importantes y definitorias.

Mucho éxito!

Challenge:
N5 company requests a Web API for registering user permissions, to carry out this task it is necessary to comply with the following steps:

- Create tables to manage employees , permission and permission types.
- Your system must allow that you have employess with "N" count of permissions type.
- Create a Web API using net core on Visual Studio and persist data on SQL Server.
- Make use of EntityFramework.

- The Web API must have 3 services “Request Permission”, “Modify Permission” and “Get Permissions”. Every service should persist a permission registry in an elasticsearch index, the register inserted in elasticsearch must contains the same structure of database table “permission”.
- Create apache kafka in local environment and create new topic where persist every operation a message with the next dto structure:
- Id: random Guid
- Name operation: “modify”, “request” or “get”.
- (desired)
- Making use of repository pattern and Unit of Work and CQRS pattern(Desired). Bear in mind that is required to stick to a proper service architecture so that creating different layers and dependency injection is a must-have.
- Add information logs in every api endpoint and log the name of operation using serilog as log library.
- Create Unit Testing and Integration Testing to call the three of the services.
- Use good practices as much as possible.0
- Prepare the solution to be containerized in a docker image.
- Prepare the solution to be deployed in kubernetes. (desired)
- Upload exercise to some repository (github, gitlab,etc).

Resources:

- Elasticsearch:https://www.elastic.co/guide/en/elasticsearch/reference/current/docker.html,https://www.elastic.co/guide/en/elasticsearch/client/net-api/7.x/nest.html
- Sql server express: https://hub.docker.com/_/microsoft-mssql-server
- Kafka:https://www.notion.so/n5now/Kafka-242a5fd883bf49ffa77190fb16eb4ecf#74a1076feed24ea482c804f54483773d
- Serilog: https://serilog.net/
- CQRS:https://docs.microsoft.com/en-us/azure/architecture/patterns/cqrs
- EF: https://docs.microsoft.com/en-us/ef/core/