# Introduction

This short assignment is intended to evaluate the basics of software development and the basic skill levels of the candidate. You are required to:

1. Create a solution that is optimal and easy to test.
2. Develop a [Solution File](https://learn.microsoft.com/en-us/visualstudio/extensibility/internals/solution-dot-sln-file?view=vs-2022) with as many projects as you think are needed. Languages accepted are C# and Javascript.
3. Follow [.NET/C#](https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/coding-style/coding-conventions) and [Javascript](https://google.github.io/styleguide/jsguide.html) Coding patterns while developing code/program solutions.
4. Write Tests wherever possible and comment on the code that needs an explanation

# Scenario

**Digital Corp is an enterprise that manufactures & sells consumer products. The company intends to build an online solution for its vendors where they can manage orders based on its product catalog. The product catalog is managed by its internal employees and a notification is sent to all active vendors when there’s a change in some key properties of a product like the Pricing or Availability.**

**For a new vendor to register, they have to go through a 2FA Authentication, Email, and Phone OTP. The OTP generated will be sent via an SMS Gateway. SMS will be:**

1. **6 Characters in Length (Numbers)**
2. **No more than 2 repeat characters (AABBCC is valid, AAABBC is not valid)**

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

# Tasks

1. Design an optimal database with necessary Primary Keys, Foreign Keys
2. Build a C# solution demonstrating Interfaces and Classes for the entities that you define in the project.
3. Use an ORM or any Native Data Access library to demonstrate:
   1. Creating the Tables at Run-Time when the application runs for the first time
   2. Update the Database Table Schema at Run-Time when a column is added to the table
4. Vendor registration and Product CRUD Feature.
5. Send automated messages to registered vendors when there’s a change in Specific Properties like the Price or Availability of the product.
6. A Basic product search feature. Sort the products based on pricing(Without using LINQ or any native sort method, **this is very important**)
7. A front-end/page to demonstrate the [pagination capabilities](https://datatables.net/examples/basic_init/alt_pagination.html) of the products list
8. Write a few unit tests for important methods

# Nice to have(Not mandatory)

1. WEB APIs for all public methods
2. Database Migration code when there’s a change in Database Schemas, like incremental versioning.
3. Use any front-end library like VueJs for the front end to demonstrate UI reactivity.

Create a Read Me file(.md or .txt) and mention the instructions to run the app(if any). You can push the code to any online Git repo and share the Repo URL, OR zip the solution project and email us back. Use SQL Joins where ever possible to optimize the SQL Execution performance.

NOTE: You are free to use any pattern/technique to build the solution. We expect the solutions to be as short and concise as possible.