Modernisation Technical Exercise

A technical exercise for .Net Software Engineers

# The Task

What is required for this technical exercise

An interface (e.g., class library) that exposes the option to create, update, and return a list of items.

# Considerations

Things to look out for and implement when completing this technical exercise

1. It is NOT important that the exercise is completed nor polished; it is important that the exercise showcases the ability to think about considerations for production ready quality code in a modern distributed cloud native environment.
2. The task is simple, and the scope is purposely broad, to encourage the Software Engineer to add features to showcase their talent and knowledge.
3. Examples of showcasing of work may include, but is not limited to:
   1. having methods (within classes) that throw “not implemented” exceptions, and are covered by failing Unit Tests
   2. having a CI/CD pipeline
   3. implementing a solution that uses cloud native features
   4. using a modern User Interface
   5. adding options to allow for scale or performance (e.g., data stores, event driven/sourcing, messaging, redundancy, & etc.)
   6. implementing automated Integration tests
4. The overall task should NOT take more than a few hours.

# Evaluation

How the technical exercise will be assessed on completion

1. A solution based on .NET
   1. Where the source code placed on a repo on GitHub or Azure DevOps
   2. With a README.md file that includes a list of Features that were
      1. implemented
      2. partially implemented (if any)
      3. considered that should be implemented (if any)
2. A Q&A session where the developer would be expected to showcase the features chosen, and explain and justify their technical decisions.