BoxCommerce

Technical Assessment

C# Senior Engineer

Premise

A vehicle manufacturer, who produces customised vehicles for order, is looking for a distributed, highly scalable, system to automate their inventory and manufacturing processes. The manufacturer has 3 independent factories to produce various vehicle components (Engines, Chassis & Option packs) on demand, as well as a warehouse for storing the finished components and assembled vehicles.

You have been selected as a candidate to design and develop a first-class solution to meet their requirements. As such the system will need to fulfil the basic tenets of modularity, maintainability, and scalability.

High-level system process

At a high-level, customers place orders for vehicles. They can customise the vehicle using 3 different components (Engines, Chassis & Option packs). The system needs to receive customer orders, then check for available stock in the warehouse - either individual components or pre-assembled vehicles. If any of the components are not available, they need to be scheduled for production. Once all the components are available, the vehicle is assembled and delivered to the warehouse ready for collection.

Throughout the process, the customer needs to be kept informed of the order progress. They also have the option to cancel an order before collection.

Required solution

The scenario outlined is a rough guideline and candidates are free to take liberties within reason. It is deliberately open-ended, allowing you the freedom and flexibility to showcase your abilities whichever way you feel appropriate. A fully featured UI is not necessary, and a barebones UI or API to trigger the workflows will suffice.

Your solution should be .NET Core based and written in C#. It is strongly recommended to include aspects from some of the following elements in your solution.

- Containerisation
- Microservice/Serverless architecture patterns
- Mediator patterns
- CQRS patterns
- DDD patterns
- Event based patterns

Additional items that would also be worthwhile to demonstrate

- Structured Logging
- Request Correlation
- Error handling
- Configuration