The maintainability and the readability issues with the code sample provided were addressed in the following manner.

* To ensure that that we adhere to the single responsibility pattern, I’ve moved the ***GetFailOverLotEntries()*** logic from the **LotService** to a separate **FailoverLotEntryDataLoader** class.
* The **ArchivedRepository, FailoverLotRepository, LotRepository** and **FailoverLotEntryDataLoader** classes all implement their respective interfaces.
* All the methods within these repository and loader classes now implement the **async/await** pattern, allowing them to be invoked in an asynchronous manner.
* The ***LoadCustomer()*** method from the **LotRepository** class was renamed as ***GetLotAsync(int id)***, since it was a more appropriate name for the operation performed by it.
* The **LotService** no longer relies on concrete implementations for any of its repository or data loader dependencies, allowing us to mock these types and unit-test the underlying service logic.
* The **LotService** logic has been unit tested via a simple **LotServiceTests** test fixture.