**HCA Software Development Organization**

**API Development Challenge**

**Overview**

This development challenge is designed to evaluate your knowledge of C#, problem solving, API design, security, and documentation. Some details are left vague for you to decide the best direction to take. If you have questions, please send them to [Gregory.Billings@HCAHealthcare.com](mailto:Gregory.Billings@HCAHealthcare.com) so we can address them.

**Introduction**

You have been tasked with building a basic lab test API in C# using .net core. Lab Tests are a common test run by hospitals and could consist of glucose tests, complete blood count, lipid panel, urinalysis, etc. The API has the following user stories:

**Lab Report API Requirements:**

1. As a user, I need the ability to create lab reports.
2. As a user, I need the ability to read lab reports.
3. As a user, I need the ability to modify lab reports.
4. As a user, I need the ability to delete lab reports.

**Patient API Requirements:**

1. As a user, I need the ability to create patients so I can store lab results for them.
2. As a user, I need the ability to read patients.
3. As a user, I need the ability to modify patients.
4. As a user, I need the ability to delete patients.
5. As a user, I need the ability to show all patients that have had a certain type of lab report in a datetime range.

**Tasks:**

1. Create the models for a patient and a lab test.
   * For Patient, you should include common attributes that you'd find for a patient such as name, DOB, gender, etc.
   * For the Lab Results, you should include common attributes that you'd find in a lab result such as type, result, time of test, entered time, etc.
2. Create the API endpoints for the CRUD operations listed in the user stories above.
3. Using an in-memory cache, create the logic that supports the crud operations in the user stories above.

**How we will evaluate your code:**

We will evaluate your code using the following methods:

1. Cloning your repository and running the code to ensure it compiles, runs, and addresses the user stories above.
2. Examine the code for readability and best C# coding practices such as KISS, DDD, Testability, etc.
3. Examining your thought process, reading through your comments, and asking follow-up questions during our de-brief session.
4. Reviewing any creative liberties, security controls, or other additions you made beyond the basic tasks & requirements.

This is your opportunity to show us your talents and skills. You are not limited to just completing the tasks above. Feel free to express yourself and add additional features & parts that you believe would be helpful for a lab test API to provide.

**Completing Time**

We would appreciate if you would complete the development challenge 1 week after it has been sent.