**.NET Challenge**

Thank you for taking the time to interview with us! We're excited about moving you forward in the interview process. Below are details on the .NET engineering challenge, which is the next step in the process.

**The goal:**Our team is excited to see your programming style, skill, approach, and creativity applied to a single-page web app.

**The challenge:**

Carton Caps is an app that empowers consumers to raise money for the schools they care about, while buying the everyday products they love. In this hypothetical example, the app is already full-featured and publicly available. The Carton Caps team has completed the designs for a new feature that allows Carton Caps app users to refer their friends to install the Carton Caps app. The new feature will be powered by shareable deferred deep links, which allow the app to present a customized onboarding experience for the referred user after installing the app. Your challenge is to design the REST API endpoints necessary to power the new referral feature.  
  
**Deliverable #1: REST API Endpoints**  
Using the attached UI designs, create a specification for REST API endpoints necessary to power the new referral feature. The API endpoints will serve as the contract between the backend server and the app. You can use your preferred API spec format, but regardless of the format you choose, your deliverable should include:

* All the necessary details for the backend team to implement the endpoints, including error states.
* All the necessary details for the frontend team to integrate support for the endpoints into the app.

You can assume that we'll be using a third-party vendor for deferred deep link support.  
   
You can also assume the app uses an existing REST API that accounts for the following (you don't need to design these):

* User authentication.
* User profile details that include the current user's referral code.
* New user registration.
* Referral redemption as part of new user registration.

Considerations for new API endpoints:

* How will existing users create new referrals using their existing referral code?
* How will the app generate referral links for the Share feature?
* How will existing users check the status of their referrals?
* How will the app know where to direct new users after they install the app via a referral?
* Since users may eventually earn rewards for referrals, should we take extra steps to mitigate abuse?

**Deliverable #2: Mock API Service**  
Implement a mock API service (i.e., web service) for the new endpoints you designed such that a developer could use the mock service to test their implementation.

* Use the .NET Core runtime.
* Data provided by the mock API can be fake, but it should be realistic enough to use for front-end development.
* Tests for your code are required. They may take any form deemed appropriate for the service, but should demonstrate your testing expertise.
* Code produced by generative AI is prohibited.
* The service and associated API spec must be placed in a public repo on a site like GitHub or Bitbucket.
* We should be able to pull down the service from this repo and build, run, and test. (Note: We will use macOS to build, run, and test, so your solution must be fully compatible with macOS.)

**Due date:**

**XXX**

* We typically give candidates 1 week to complete the challenge, but if for any reason you need more time, tell us now, and we're happy to accommodate.

**Consideration:**

* One way we're able to assess engineering candidates on a level playing field is by giving every serious contender for a position on our team the exact same assignment with the exact same inputs to see how they approach their thinking and execution.  This helps us reduce bias in our selection and make the interview process fair for everyone.

**Hint:**

We're a senior group of folks who see software as a craft. We'll be looking for well-structured code that follows best practices, is tested, doesn't have errors, is commented appropriately (e.g., comment on your public interfaces), considers edge cases, and demonstrates your creativity and love of software.