**Assumptions:**

* All data for MVP will be correctly input by the user, thus validation is not needed (for sake of time, in a production environment I would always add validation on user input).
* The MVP is being hosted on an un-secure environment.
* There is no database, the json files provided will be updated accordingly.

**Question to raise to the ScrumMaster/PM or Business Stakeholders/Product Owners:**

Hosting/Access:

* Where will this API be hosted?
  + Local server?
  + On the cloud? (ex. Azure or AWS)
* Will the API be hosted on a secure or non-secure environment, or both?
  + If a secure environment, are any certificates required for environment access?
* Is there a designated base URL for the API? (ex. [www.google.com](http://www.google.com))
* What is the expected usage of the API (how many users will use it per day/month/year)?
* Do calls need to be synchronous or asynchronous?

Security:

* Is an authorization token needed?
* Is an API token needed?
* If yes, what is the preferred method of providing those tokens?
  + In the header?
  + In a cookie?
* Where should errors be logged to?

Data:

* Is a database needed, or is everything up-kept in the JSON files provided?
  + A database to store that information would be more secure.
  + If a database is requested for the data:
    - Who has access to the database?
    - What type of authentication is needed for the database?
    - Would you like to utilize Entity Framework for the solution? (highly recommended)
* To create accurate objects, what are the types for each variable in the json documents provided?
  + Are any variables read-only? Static?
  + Are there any constraints on any of the variables, like would age have a min of 0 and a max of 100?
  + Can we alter the objects schema to enhance code reuse? (ex. Both agents and customers have an address, we could make an address object that both can use)
* There is a phone number and an address for both Agents and Customers, would you prefer a table for both with foreign keys to the Agent or Customer the address/phone number represents, or keep the schema as is?