**Brief overview of solution :-**

1. Solution is built in .net Core 3.0 as a REST Web API.
2. Swagger contract is created that can be shared to consume.  
     
   
3. Solution has been structured into folders for better readability
4. aController Layer  
     
   a) 3 different APIs have been written to fetch results as per the requirement.   
     
   b) It injects IAddressBookService and ILogger to implement the business logic in the service layer and to log errors/information when required.   
     
   c) Async/Await has been used to indicate asynchronous data fetching (for database layer in future)
5. Service Layer

a) It injects – IAddressBookLoader to load the address book and ILogger to Log errors/information when required.

b) IAddressBookLoader/CsvReader has been implemented to isolate Logic of Loading AddressBook from a file.

5. Unit Tests

a) Unit test implemented for Controller Layer, Service Layer and CsvReader to demonstrates TDD approach using XUnits and Moq.

**Improvements/Extensions :**

1. **Provide API to Refresh AddressBook** - so that if the address book is updated then the application is not required to be restarted. While providing this API we need to handle concurency issues that might arise due to simultaneous execution of multiple API requests.

2. The filename to load the address book should be maintained in AppSettings.json.