Initial Task Brainstorming

1. Solution Architecture based on requirements

* API Project – Hosting
* Persistence
  + EF Scafolding
* Services
  + CQRS – MediatR, DI
* Domain
* Repository

2. Understanding the Domain Objects

Creation of Domain Objects and Understanding their relationships:

2.1 Events (I will treat event as same with appointment). Only entered into the system by internal users. Authenticated and Authorization to be applied when CRUDing. Assume that Events do not overlap days. Starting time and Ending time of events and appointments is 08:00 to 16:00

2.2 Attendees – these may book online (via a UI app). Need to have a valid email address. No authentication and Authorization attributes to be applied to this endpoint.

2.3 Doctors – Assume that the doctor only works on this practice and nowhere else. Also that all doctors do the same thing ie specialisation.

2.4 Bookings – for both Events and Appointments

2.5 Users – perform booking internally – after receiving a call, email etc

3. Making a Decision on ORM to use - ie EF, Dapper etc

- EF because of exntensive background. To decide on light weight or full

- In-Memory DB (can be easily)

4. Mapping - accelerate delivery of code

4. Logging and Auditing

4. Authentication and Authorication