# **Improvement**

## **Architecture**

1. Applying CQRS for better performance.
2. Applying Microservices if this Application will be Complex or there are Scalability Requirements.
3. Implement Message Queuing like RabbitMQ if we need Decoupling and Asynchronous Processing

Or Scalability and Load Balancing.

1. Can Use non-relational (MongoDB) stores data as JSON documents in a more flexible format.

## **Implementation**

1. Entity:

* We Can add User and deletion Signature.
* For PreferredCallTime we can make it Timestamp (from-to).
* Add Candidate Image or Candidate CV.

1. Generic Repository:

* Implement All basic Functionality (delete +add List+ update list + pagination ,….etc ).

1. Implement Authentication and authorization.
2. Log request and response for better tracking, especially if you want to implement Microservices to track Transferring data between micros.

## **Caching**

1. Database Caching (sql server in memory oltp)
2. Application-Specific Caching (Response Caching )
3. Redis

Point 1,3 I read about but I never implemented.