MOBVEN

# .Net Interview Challenge

## Project Info

You are going to do a Movie Application with Service Oriented Architecture. The project should be a single page and the data in mssql read through the Asp.Net Api. There should be one movie datas on screen. The data displayed on the page should be brought through a 12 minute cache. Client can search any movie with Asp.Net API. If movie data is not in the database, then search request must go to omdbapi from backend. Then, you save the omdbapi movie search result to mssql.

Requests to the ASP.NET API through the frontend must be done with javascript. Movie data on mssql should be updated every 10 minutes without using windows service from omdb api. The structure in omdbapi integration should be easily interchangeable. (Loosely Coupled) So instead of omdbapi as a source of data, the logic layer should not be changed when another api added. Asp.Net Api must be have a documentation. You can use swagger or whatever you want. And logging in case of any error.

## Structures To Be Used

- Service Oriented Architecture
- Asp.Net Web API
- Microsoft Sql
- Cache
- Dependency Injection
- Log
- Authentication

## Bonus

- Using .Net Core Framework
- Cache on Redis or Memcached
- Log on MongoDb or another noSql tool.
- Using React or Angular.js in frontend.
- Clear all cache one keyword. For example api call like: [HttpGet]api/cachemanagement/clear

## Questions

- Which patterns did you use and why did you choose those patterns?
- Which DI tools did you use and why did you choose that tool?
- Which auth mechanism did you use and why?
- How did you fix a “10 minute data update issue” without Windows Service? Do you have any additional comments about this?
- What would you like to do if you had the more time?
- Do you have additional comments?

## Evaluation Criterias

- First of all, your application should be running.
- Implementation of the logic from SOLID development perspective.
- Clean and understable code.
- Loosely coupled structure.
- How much you show your developer skills.
- Your approach to the solution.

## Notes

- We don't expect the application to have a perfect design but it should be have a basic design. You can use free html templates.
- You should use the camelcase writing style in project.
- All comments, codes and documentation must be in English.
- This project must have a unit test.
- OMDb API Free and Documentation:
  http://www.omdbapi.com/

## Important

Please show your developer skills when developing this application.