**Microsoft .NET Technical Experience Evaluation Questionnaire**

Full Name:Sadashiv Byakod

Date:07-08-2025

Please note that the completion of this assessment/questionnaire should solely rely on your own experiences, knowledge, and capabilities. The use of ChatGPT or any AI assistance is strongly discouraged. Any indication of AI usage may result in the termination of the hiring process.

**Q1)** Name three software projects you took part in the last three years? Please mention your role in the project as well as the project duration. (Please write three projects at most.)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name of the project** | **Date/ Duration** | **Your role** | **Client** | **Team Size** |
| **Worley R3** | 3 years | Sernor developer | US | 12 |
| **ConnectMe** | 2 years | Anyalist | **American Airlines** | 8 |
| Custom Translator | 3 years | system Engriner | Microsoft | 5 |

**Q2)** Which technologies did you use in the projects you mentioned above?

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of the project** | **Prog. language** | **Database** | **Reporting system** |
| **Worley R3** | reactjs ,c#, | sql |  |
| **ConnectMe** | angular | sql |  |
| Custom Translator | c#, Angular,typescript | sql |  |

**Q3)** List projects where you have used Angular with project duration and your role in project.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of the project** | **Duration** | **Role** | **Angular Version** |
| **ConnectMe** | 3 years | developer | v6.0 |
| Custom Translator | 2 years | developer | v14.0 |
|  |  |  |  |

**Q4)** Have you ever worked in a project where there were branches of development in a source control application? If yes, what was the name of the project, how many branches were in use?

<< yes Custom Translator braches -5>>

**Q5)** How do you handle database schema migrations and versioning in your projects? Describe your approach to managing schema changes across different environments (e.g., development, staging, production). What tools or practices do you use to ensure smooth transitions and minimize downtime?

<<Using migration tools, version control, a robust testing process, and a clear rollback strategy is crucial to effective schema management. Regular reviews and documentation help keep our migration strategy relevant and understandable for our team.>>

**Q6)** Did you ever use cryptography in your projects? If yes, which algorithms did you use, for what purpose and what was your key management strategy such as the storage place for keys?

**Q7)** What are the things you would do or avoid doing for improving performance of a web application? Please list top 5 most important things for you.

<<**Use Minification and Compression**, **Implement Lazy Loading,** **Optimize Database Queries,** **Enable Caching, Reduce HTTP Requests:>>**

**Q8)** What are the things you would do or avoid doing for improving security of a web application? Please list top 5 most important things for you.

<< **Sanitize and Validate User Input**:,

İmpletetaing azure Ad ,jwt,etc

**Regularly Review Security**

**Set Up a Web Application Firewall (WAF)**:,

**Choose a Mature Framework**:>>

**Q9)** Assume that you are developing a web application that shows currency exchange rates for a bank web site. The rates are stored in an IBM Mainframe in the bank and rates are available through web services. Each time a user accesses the rates page, the page makes a request to the mainframe. This generates too much load on the mainframe. Especially most of the time the rates delivered is the same. But at the same time rates may fluctuate within the day and if rates have changed the web page should not display the cached values but make another request to the web service. The rates table at mainframe supports triggers where each update to the rates table can invoke a trigger where the trigger may further call a web service. How would you design such a caching architecture and make sure it is invalidated on rate changes? **Please explain by drawing a diagram.**

**--------------------+**

**| User |**

**| Request Rates |**

**+--------------------+**

**|**

**v**

**+--------------------+**

**| Check Cache |**

**+--------------------+**

**|**

**+-------+-------+**

**| |**

**| Cache Miss Cache Hit**

**| |**

**v v**

**+--------------------+ +---------------------+**

**| Request to | | Return Cached Rates |**

**| Mainframe Web |----------| to User |**

**| Service | +---------------------+**

**+--------------------+**

**|**

**v**

**+--------------------+**

**| Update Cache |**

**| with Latest Rates |**

**+--------------------+**

**|**

**|**

**v**

**+--------------------+**

**| Trigger on |**

**| Mainframe Update |**

**| calls Invalidate |**

**| Cache |**

**+--------------------+**

**|**

**v**

**+--------------------+**

**| Clear/Update |**

**| Cache |**

**+--------------------+**