|  |  |  |  |
| --- | --- | --- | --- |
| **INTERVIEW FEEDBACK FORM** | | | |
| **Candidate Name** | Sanduni | **Interviewer** | Dileka |
| **Position Interviewed** | Software Engineer | **Position Recommended** | Software Engineer |
| **Referral / Campus** | University of Moratuwa | **Interview #** |  |
| **Date and Time** | 2021/01/07 | **Venue** |  |
| |  | | --- | | **CANDIDATE ASSESSMENT** |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** | | **Analytical Skills** |  |  |  |  |  | | **Creativity** |  |  |  |  |  | | **Learning Ability** |  |  |  |  |  | | **Passion** |  |  |  |  |  | | **Communication** |  |  |  |  |  | | **Team Work** |  |  |  |  |  | | **Attention to Detail** |  |  |  |  |  | | **Accountability** |  |  |  |  |  | | **General Attitude** |  |  |  |  |  | | **Cultural Fit** |  |  |  |  |  | | **Domain knowledge** |  |  |  |  |  | | **Respective skills to perform the job** |  |  |  |  |  | | **Adaptability and Flexibility** |  |  |  |  |  | | | | |
| **FINAL RATING** | | | |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Superior** | **Very Good** | **Good** | **Standard** | **Below Standard** | |  |  |  |  |  |  |  | | --- | | **STATUS** |  |  |  |  |  | | --- | --- | --- | --- | | **Recruit** | **Reject** | **On-hold** | **Next Interview** | |  |  |  | **yes** | | | | |

|  |
| --- |
| **REMARKS** |

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | | **STRENGTHS** |   **The candidate has proper knowledge on Java, REST API and SQL and I would like to have a longer technical interview with the candidate**   |  | | --- | | **WEAKNESSES** |  |  | | --- | | **COMMENTS** | |

**Candidate Level**

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Beginner Intermediate Advanced**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **FUNDAMENTALS** | | | | | |
|  |  |  |  |  |  |
|  | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **OOP Concepts** |  |  | Good |  |  |
| **Language fundamentals** (Evaluate language proficiency while relating language principles to core computer science concepts) |  |  |  |  |  |
| **Algorithms** (Queues, Stacks, Linked Lists) |  |  |  |  |  |
| **Operating Systems** (Linux, kernel, threads vs processes) |  |  |  |  |  |
|  |  |  |  |  |  |
| INTERMEDIATE CANDIDATES ONLY | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **Design Patterns and their application** |  |  |  |  |  |
| **UML** |  |  |  |  |  |
| **Unit testing** |  |  |  |  |  |
|  |  |  |  |  |  |
| **SERVER SIDE/ENTERPRISE** | | | | | |
|  |  |  |  |  |  |
|  | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **Evaluate conceptual knowledge on server-side frameworks and libraries** (e.g. JEE (JSP, Servlets, EJB), Spring, Hibernate Struts, Node.js, Express.js) |  |  |  |  |  |
| **Transactions concepts** (ACID, JDBC auto commit. Save points and rollbacks (if applicable)) |  |  |  |  |  |
|  |  |  |  |  |  |
| INTERMEDIATE & ADVANCED CANDIDATES ONLY | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **Transaction attributes and transaction propagation** |  |  |  |  |  |
| **REST** |  |  | Good |  |  |
| **Web services** |  |  |  |  |  |
|  |  |  |  |  |  |
| ADVANCED CANDIDATES ONLY | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **Enterprise Patterns** |  |  |  |  |  |
| **Distributed transactions** (XA) |  |  |  |  |  |
| **Transaction isolation levels and their behavior** |  |  |  |  |  |
| **Unit and integration testing** |  |  |  |  |  |
| **Security** (e.g. secure coding practices, SQL injection, symmetric vs asymmetric encryption, salted hashes, etc.) |  |  |  |  |  |
| **Enterprise Integration Patterns** |  |  |  |  |  |
|  |  |  |  |  |  |
| **DATABASE & SQL** | | | | | |
|  |  |  |  |  |  |
|  | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **Normalization** |  |  |  |  |  |
| **ER concepts** |  |  |  |  |  |
| **SQL Exercise** (Identifying entities and their relationships for a simple use case) |  |  | Good |  |  |
|  |  |  |  |  |  |
| ADVANCED CANDIDATES ONLY | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **Dimensional Modelling** |  |  |  |  |  |
| **Query optimization** (Explain plans, hints, etc.) |  |  |  |  |  |
| **NoSQL** |  |  |  |  |  |
|  |  |  |  |  |  |
| **FRONT-END** | | | | | |
|  |  |  |  |  |  |
|  | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **HTML/5** |  |  |  |  |  |
| **CSS** |  |  |  |  |  |
| **JavaScript Fundamentals** |  |  |  |  |  |
|  |  |  |  |  |  |
| ADVANCED CANDIDATES ONLY | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **SPA framework proficiency** (Ember, React, Angular or a framework of their choice) |  |  |  |  |  |
| **Cross Site Scripting** (XSS) |  |  |  |  |  |
|  |  |  |  |  |  |
| **MISCELLANEOUS - FOR THOSE WHO ARE MORE CONVERSANT** | | | | | |
|  |  |  |  |  |  |
|  | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| **DevOps: Containerization, Jenkins, AWS** |  |  |  |  |  |
| **Functional programming** |  |  |  |  |  |
| **Architectural questions** |  |  |  |  |  |
|  |  |  |  |  |  |
| **SOFTWARE DEVELOPMENT METHODOLOGY** | | | | | |
|  |  |  |  |  |  |
|  | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| Familiarity with Agile/Scrum concepts |  |  |  |  |  |
| ADVANCED CANDIDATES ONLY | **Excellent** | **Very Good** | **Good** | **Standard** | **Below Standard** |
| Story Point based estimations |  |  |  |  |  |
| If PO asks..., what would you do..., questions |  |  |  |  |  |