Jairo Flores AI Interview - RPG, Java and DB2 Development

Question: Java Development

Can you describe a project where you used Java to develop a solution? What was the project about, and what were the key challenges you faced?

Answer:

In my role at Datapro, I worked on the e-IBS banking system implementations for several client banks. The system's frontend was developed using J2EE technologies, including HTML, JavaScript, JSPs, and Servlets. It connected to a backend developed in RPG modules, which processed requests and generated data in DB2, connected to external databases, performed ETL or ELT processes, and produced reports using the Java POI package for end users. My responsibilities included adapting the baseline system to meet client requirements or comply with country-specific laws, and connecting or migrating bank products.

Each project involved creating new JSP pages with client-specific requirements, modifying servlets, and developing Java Beans to handle API and web service requests as well as socket connections. Additionally, I worked on the RPG backend to implement business rules, modify and save data into databases, generate Java POI reports, and program ETL processes.

In this position, I collaborated closely with business implementers and client bank personnel to deliver solutions that met all requirements, leading to high client satisfaction and repeated requests for my involvement in new projects.

Question: DB2 Database Development

Can you describe a situation where you had to design and develop a DB2 database from scratch? What was the project, and how did you approach the design and development process?

Answer:

I led a project to migrate an insurance system from Oracle to IBM i, using RPG and DB2 as the backend and web technology for the front end and transactions for the banking teller system.
The initial phase involved analyzing the Oracle catalog to identify components for duplication, rewriting, and data migration. This included reports, data objects, and forms.
We designed and implemented the DB2 schema, including tables with relationships, constraints, indexes, and views. Existing data was then successfully migrated.
We then developed stored procedures, functions, and backend programs to interact with the data.
I made a proof of concept and proposed to use JavaScript ExtJS framework for the frontend, to not use the RPG classic green screen. The proposal was approved, and I started to built the controller programs to attend the http requests from the JavaScript ExtJS frontend and MQ Series controllers to handle the requests from the .NET for bank teller forms. At the end, I developed the web forms.

The project was completed on time, fully tested, and, as a weird fact, it included a stored procedure to provide data for an Excel workbook used by insurance personnel.

Question: RPG and SQL Optimization Technical Expertise

Tell me about a project where you debug and modify an SQLRPGLE to optimize database performance. What specific techniques did you use?
What were the challenges, and how did you address them?

Answer:

In my current role, and for the past two years, I've been optimizing components using reports on system performance produced by IBM Healthchecks during high-intensity processes. One report indicated that a program handling electronic branch services was consuming excessive database resources. To address this, we analyzed the program's db access and found missing indexes, and that it needed to implement blocking reading techniques. We optimized the queries by adding recommended indexes and applied blocking reading techniques, but I alerted my leader that we'll need to test functionalities that weren't modified. For testing, I used a MQ tester to send XML documents as request to the service. Given the numerous instances of MQ controller jobs, I set service entry points to halt the program for debugging purposes. Upon sending the XML documents, the program initiated, allowing me to debug it effectively. I proposed to use Rational Open Access and wrote a brief proof of concept, which was approved. Using this product we needed to modify only one line of code with the file declaration lines and create a service program to handle database optimized reading with logical blocking of 1,000 records at once. This approach reduced processing time by 45% while preserving the legacy code and requiring only regression and stress tests. In the next presentation report, IBM recognized and promoted this project as a model for future optimizations.

Question: Problem-Solving Skills

Can you describe a time when you encountered a significant obstacle while working on a project? How did you overcome it?

Answer:

In the same project where we achieved a 45% reduction in processing times, we encountered an issue in another country, where the program produced incorrect order results. Despite retesting all, using testing and staggin systems, we couldn't find any problem. I contacted the person who reported the issue and asked him to reproduce it while we monitored the database components. We discovered an index specific to that country affecting the results. The IBM database was prioritizing the SQL Query Engine (SQE) over the Classic Query Engine (CQE), leading to incorrect order of results. We created a new SQE index with the correct order, which resolved the issue and ensured the program worked correctly across all systems.

Question: Technical Expertise and Team Work

Describe a situation where you had to integrate new technologies (like Webservices or APIs) into an existing IBMi system. How did you approach this task?

Answer:

The bank I worked for needed to integrate with a US government web service to retrieve and apply payments for retirees using their .NET teller system. This involved a new type of teller transaction that required significant data exchange between the .NET system and the IBM i backend.

My task was to collaborate with a .NET developer to design and implement this transaction. I coordinated with my .NET coworker to plan and define the necessary fields for the transaction in both systems. The teller transactions communicated with the IBM i using MQ Series.

I gathered detailed data about the government SOAP web service and data samples. I chose to use an open-source API to integrate with the web service. Initially, I created tests for the internal transaction, then for the external web service, and finally for the integration of all systems.

While my colleague handled the frontend in .NET, I developed the program to handle the teller transaction, incorporating accountability and payroll applications, and tested it with realistic test data. Next, I created a parametric framework to register web service endpoints using keyed data queues to manage multiple instances, balance loads, handle timeouts, and invoke appropriate programs based on transaction IDs.

After registering the government web service data in the framework's tables, I developed and tested the program to handle the transaction code and established communication with a dummy web service. We conducted thorough testing together with the accounting department to validate the transaction's accuracy and reliability. I ensured that the IBM i system could handle the volume of data and process multiple transactions efficiently.

We performed integration tests using transactions from the .NET teller interface to identify and fix weak points or errors. Once installed in production, we tested the entire process with a few transactions, resolved some application errors, and completed the project. This framework subsequently became the standard for similar needs in the bank.

Question: Problem-Solving

Describe a situation where you had to solve a difficult problem. What did you do?

Answer:

In my previous role at a bank, we faced a significant bottleneck in our mobile air time charging framework. The system relied on a single job to handle transactions from multiple channels, causing delays and timeouts.

My task was to re-engineer the framework to handle multiple transactions concurrently and efficiently. I collaborated with the business product owner to understand the desired functionality and requirements.

A thorough analysis of the transaction flow from various channels identified the bottleneck. I gathered technical details about how transactions were processed and how our program connected with cellular companies' APIs.

Based on these insights, I designed a new flow using asynchronous processing with keyed data queues. I implemented a solution where each transaction message in the data queue was handled by a created job, allowing multiple transactions to be processed simultaneously. The system could dynamically manage the number of created jobs, simulating the behavior of web servers like Express or Apache.

This re-engineering significantly improved the system's performance by reducing transaction delays and timeouts, thereby enhancing customer satisfaction and operational efficiency.

Question: Project Management and Organizational Skills

Describe a time when you had to handle multiple responsibilities. How did you organize and prioritize your tasks?

Answer:

In my role as a Data Center Supervisor, I was responsible for a broad range of duties, including applying production software changes and updates, overseeing data center operations, and managing various administrative tasks.

Juggling these responsibilities effectively was key. I supervised data center operators, handled tape storage, applied software changes, coordinated system backups, and planned the department schedule.

To keep the team well-prepared and organized, I created a detailed monthly schedule for the department, ensuring all tasks were planned and discussed with my supervisor weekly. I established a routine that prioritized tasks in advance, enabling a proactive approach.

I supervised the rotation of data center operators and ensured the availability of tape storage for daily operations, applied software changes and updates, controlled system backups, and troubleshooted any issues that arose due to incompatible changes errors.

Finally, I coordinated with team members to resolve high-priority issues and ensure smooth operations across multiple countries with varying banking laws and requirements.

This approach fostered a well-organized department where everyone knew their duties a month in advance. It resulted in excellent performance metrics and increased efficiency in handling daily operations and unexpected issues.

Question: Communication Protocols (SFTP, AS2, HTTPS)

Can you explain a situation where you had to set up and troubleshoot a communication protocol such as SFTP, AS2, or HTTPS? What steps did you take to resolve any issues?

Answer:

At Datapro, we had a client bank that generated and transmitted files manually to an external client. Our task was to automate this process using secure FTP (SFTP).

We were provided with the public keys for the destination test system and created our system's public keys, which we provided to the SFTP server administrator. We then configured the IFS user directory for the local donwload/upload files and validated the passwordless SFTP connection.

Next, we modified the backend to automatically generate the data using the job scheduler. During testing, we encountered issues with automatic sending. By analyzing the SFTP log, I identified and fixed the IFS directory permission problem. The result was a successfully automated process, significantly improving efficiency for the client bank.

Question: Web Services and XML Technologies

Can you give an example of a time when you had to work with SOAP or REST web services? What was the project, and what challenges did you face?

Answer:

"The bank I worked for needed to connect to a client server using HTTPS over TCP/IP to retrieve an XML document, apply ETL rules to the transactions, and return the results. The client server exposed a SOAP web service. My first task was to analyze the WSDL and XML schema provided by the client.

Using an open-source RPG API, we developed a framework to consume web services. We registered the client and developed the necessary program to handle this process. To test our framework, we created a Node.js local server to expose a web service, providing example data for our tests.

After passing the integration tests, we deployed the solution in production, initially pointing to the client's test server. Once certified, we updated the parameters to point to the production server. The result was a satisfied client with a seamless and efficient process."

The bank I worked for needed to connect to a client server using HTTPS over TCP/IP to retrieve an XML document, apply ETL rules to the transactions, and return the results. The client server had exposed a SOAP web service. My first task involved analyzing the WSDL and XML schema provided by the client.

We developed a parametric framework of services to consume web services using an open-source RPG API. This framework involved registering the client and developing the necessary program to handle the process. To test our framework, we created a local Node.js server that exposed a web service, providing sample data for our tests.

After passing the integration tests, we deployed the solution in production, initially pointing it to the client's test server. Once certified, we updated the parameters to target the production server. The result was a satisfied client with a seamless and efficient process.