**CDR API Development**

I am excited to present the CDR API that I developed using .NET Core version 8.0 for the API layer and PostgreSQL 8.7 for the database. My approach was focused on creating an efficient, scalable, and user-friendly API to handle and process CDR (Call Detail Record) data.

**Technologies Used:**

- **API**: .NET Core version 8.0

- **Database**: PostgreSQL version 8.7

For this project, I opted for PostgreSQL, and although a NoSQL option might have suited certain use cases, PostgreSQL offers excellent scalability and flexibility for handling large datasets.

**Development Approach & Challenges:**

I have previous experience developing WCF web APIs using older versions of the .NET Framework and SQL Server. Due to the unavailability of Visual Studio 2012, I developed this project in Visual Studio 2022, which presented some challenges. For example, I didn’t have full permissions to execute in the Package Manager Console, so I employed the Database-First approach, bypassing the automatic migration generation.

**Assumptions:**

1. The CDR data will be provided as a CSV file.

2. The API will be primarily accessed by internal teams for programmatic data retrieval.

**Key Features and Considerations:**

- **Error Handling:** I’ve implemented comprehensive error handling throughout the system, including for file uploads, database interactions, and API requests.

- **Scalability:** The design anticipates the handling of large datasets. Techniques like data partitioning and background processing for uploads have been incorporated to handle potential future growth.

**Future Enhancements:**

I see opportunities to further improve the API and make it more valuable to users:

- **Logging:** API requests, errors, and database transactions will be logged to facilitate monitoring and debugging.

- **Advanced Analytics Endpoints:** Adding features like call trends over time and cost analysis by destination.

- **Data Visualization Integration:** Integrating with visualization tools for enhanced data exploration and reporting.

- **Cloud Deployment:** I am exploring options for deploying the API in the cloud to improve scalability and manageability.

I believe that this CDR API has the potential to provide a solid foundation for efficient data management and retrieval, while also being easily extensible for future needs. I would be excited to discuss how my experience and this API could be leveraged to meet your assignment goals.

Thank you for considering my application.

Best regards,

Aishwarya Gayake