**Individual Contribution Report - Person 3**

I oversaw the Neo4j graph database implementation for this project, as well as the writing of SQL scripts to validate the database and make analytical queries.

I contributed by using SQL queries, for example, to ensure that the relational data warehouse was intact and consistent. That meant seeing to the foreign key integrity, counting data discrepancies between fact and dimension tables and prewriting headline queries to decode later.

To write up the graph database side, I ginned up the Neo4j model by:

* Make a node for each major entity, such as Products, Categories, Vendors.
* Defining relationships like BELONGS\_TO\_CATEGORY, BELONGS\_TO\_SUBCATEGORY and SUPPLIES.
* Using Cypher scripts to import structured CSV datasets into Neo4j.

I wrote up seven Cypher (CQL) queries, chosen carefully to echo complex SQL JOIN operations while being conceived through graph traversals. These queries showed how graph databases allow multi-hop relationships to be queried with fewer lines of code and execute faster.

Working on it, I buffed up my skills in graph modeling, Cypher Query Language, relational-to-graph data transformation and advanced SQL querying. The difficulties were translating SQL JOIN logic into Cypher patterns and optimizing the imports of graphs for speed.

This project also taught me how to work with non-relational databases, a skill that’s important now for modern data analytics and big data ecosystems.