Greetings,

I hope this email finds you well. I am Likhita and am writing to provide you with an update on the progress of the project and discuss next steps.

I worked on the Fetch rewards data and want to share my insights with you. I was provided with Receipts, Brands and users tables in the form of zipped JSON files. Using these files, my first step was data cleaning and transformation. For this process, I leveraged advanced Python libraries and packages. Subsequently, I performed data validation and exploratory analysis to find inconsistencies, missing values and anomalies in the dataset.

In order to resolve these issues, I considered each user to have many receipts, and each receipt to have many receipt items as the basic business statement. Accordingly, I eliminated duplicates in tables (users), broke down columns (receipts table), and replaced null values.

<u>While doing data quality checks, I gathered the following questions:</u>

- Examining the PointsEarned column from Receipt table, we see that data for points earned is NULL for 15% of the records. We are not sure if data is not populated at the backend or reward points are not given to the users. This confusion is because we also have users who received 0 bonus points. Hence, about 15% of data is misinterpreted. This can be a crucial points to improve the business process. So, there is a need to eliminate the ambiguity.
- I noticed that the 'brand Codes' are not numerical - to avoid any mistakes while joining the tables, it is advisable to have that column in alphanumeric or numeric so as to ensure precise and exact joins by avoiding problems of case sensitivity etc.
- The Signup source for users is only ‘Email’ and ‘Google’ from the data. There can be many other options such as Phone number, SSO, Two-factor authentication etc. and we will be missing out on those opportunities.
- I have also noticed many duplicate User ID’s in the user table. I wanted to understand the reason for such duplicate values since there is already a column for active users as well.

To optimize the data assets that I created, information about specific business requirements or outcomes will be more helpful. Once we understand the end goals, it will help us in tailoring the data objects and designing models to meet those objectives effectively.

- While scaling it to production, we can also add some default and check constraints during insertion of data into the tables to eliminate the issue of NULL values.
- Since it is a dynamic transactional dataset, we can use Amazon S3 for data handling to handle the complexity of rapidly increasing data volume.
- To improve query performance, we can use query indexing, and construct views according to the requirements.
- We can also use integration and automation techniques using dbt and Airflow.

These are my initial thoughts on the data. Thank you for your attention and continued support. We look forward to your feedback and progressing with next steps.

Best regards,
Likhita.