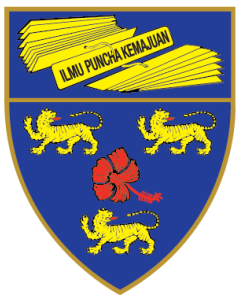

UNIVERSITI MALAYA

WIE3007 DATA MINING AND WAREHOUSING

SESSION 2022/2023
SEMESTER 1

Individual Assignment

Topic: Cashierless Retail Stores

PREPARED FOR:

Dr. Riyaz Ahamed

PREPARED BY:

Name	Matric No.
GAN JIA SOON	17206343/1

1.0 Introduction

The cashierless retail stores can be defined as stores that abolish traditional physical cashier desks and apply the newest technology in their check-out process. Currently, the traditional check-out process still plays a central role in retail, and is mostly done through a combination of human contact with the cashier and machine-assisted services such as self-checkout stations or through electronic service solutions like pick-up stations (Fitzsimmons, 2019).

One of the instances for cashierless retail stores is the Amazon Go store, which provides IoT-based shopping that mainly aims to reduce queuing times at the cashier desks by using an app, cameras and sensors (Dougall, 2018; NZZ, 2018). It is useful to helpful to adopt the cashierless retail concept as the automation in retail can enable companies to save up to 81 % of the time currently needed for cashier activities (Begley, Hancock, Kilroy, & Kohli, 2019).

2.0 Requirement analysis

2.1 Business Problems

1. **Time-consuming and limited analysis from the raw data**

   The analytics team will need to perform very complex joins in order to do analysis for some specific business cases. Besides, they require a lot of time to do the pre-processing of the raw data such as data cleaning before the analysis could be made.

2. **Low profit and conversion rate**

   Customers are not interested to buy things from the stores although the number of customers coming in the stores is high. The company is also facing difficulty in identifying popular product and payment method to suit customer preferences.

3. **Inconsistent answer from the analysis**

   The result of the analysis from the data is not consistent as the data directly from the database is updated from time to time.

4. **Slow query performance**

   The normal database is not fully optimized for analytical processing.

5. **Poor inventory management**

   Currently the staffs have to manually check and monitor the available stock amount and restock accordingly if necessary. This is a time-consuming process and a waste of manpower to monitor regularly.

2.2 Project Milestone & Duration

Project Title: Data warehousing of Cashierless Retail Stores

Estimated Project Duration: 316 days

There are 8 phases in this project with reference to the data warehouse development life cycle model. Besides, there will be 5 important milestones within the whole project timeline. The overall project timeline and important milestones are showed in the following gantt chart:

TASK	START	ETA	DAYS
Phase 1 Requirements Gathering			
Determining the project scope in relation to business objectives	01-11-2022	08-11-2022	7
Business requirement specification documents - Preparation & Alignment	08-11-2022	23-11-2022	15
Business requirement specification documents - Sign-off	23-11-2022	28-11-2022	5
Phase 2 Setting Up Physical Environments			
Development environment	28-11-2022	08-12-2022	10
Testing environment	08-12-2022	13-12-2022	5
Production environment	13-12-2022	23-12-2022	10
Phase 3 Data Modeling			
Data Model Design - Draf	23-12-2022	30-12-2022	7
Data Model Design - Final Version	30-12-2022	09-01-2023	10
Documentation	09-01-2023	14-01-2023	5
Phase 4 ELT Design and Development			
ETL Implementation plan & Selection of Tools	14-01-2023	03-02-2023	20
Technical specification documents - Alignment	03-02-2023	13-02-2023	10
Technical specification documents - Sign-off	13-02-2023	23-02-2023	10
Technical implementation	23-02-2023	24-04-2023	60
Phase 5 Online Analytic Processing (OLAP) Cube			
OLAP Server Design & Alignment	24-04-2023	14-05-2023	20
OLAP Server setup	14-05-2023	03-06-2023	20
Phase 6 UI Development			
UI Design	03-06-2023	23-06-2023	20
User testing & Feedback	23-06-2023	03-07-2023	10
Sign-off	03-07-2023	08-07-2023	5
Phase 7 Maintenance			
Update of schema & application domain	08-07-2023	18-07-2023	10
Phase 8 Test and Deployment			
User Acceptance Testing (UAT)	18-07-2023	07-08-2023	20
Deployment	07-08-2023	06-09-2023	30
Project Sign-off	06-09-2023	13-09-2023	7

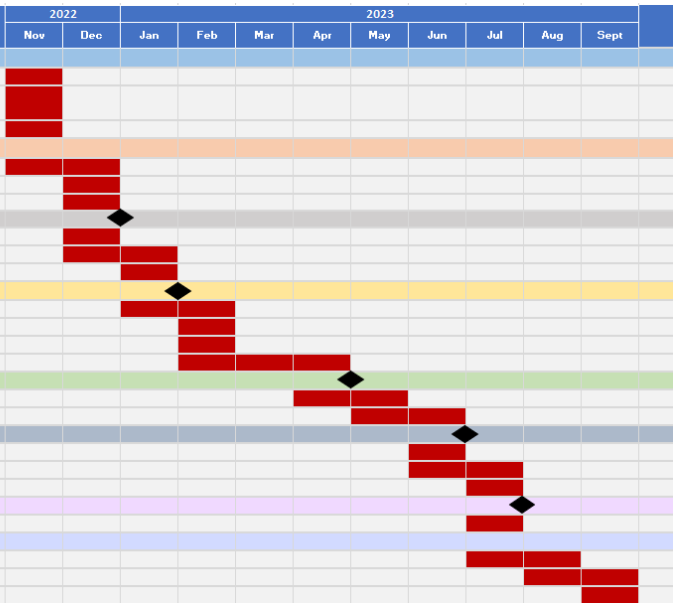

Figure 2: Project Gantt Chart

2.3 Major Project Outcome

1. **Better efficiency for the data analysis**
   The structure of both data warehouses and data marts enables end users to report in a flexible manner and to quickly perform interactive analysis based on various predefined angles (dimensions). Faster for the analytics team to process the raw data. Less time is needed to clean and transform data to perform analysis.

2. **Higher profit and conversion rate**
   Learning from data is the central method to improve their product. By analyzing the customers' behaviour and historical purchase data collected from the IOT devices, we will be able to improve the selection of the products after identifying the popular products and payment methods.

3. **Consolidating data to a single, reliable repository**
   Standardizing data from different sources also reduces the risk of error in interpretation and improves overall accuracy. It also helps in creating a single source of the truth.

4. **Faster query performance**
   The time required to query from the data warehouse is faster than the traditional database. As it is separated from transactional data schema, queries will not affect system performance, and not affected by rapid changes in the data.

5. **Better inventory management**
   More efficient inventory management based on real-time data, notification will also be sent by the Azure Logic App if the item is needed to be restocked.

2.4 Reporting Authorities

The table below displays the list of reporting authorities with their respective roles related to this project:

Reporting Level	Role	Reporting PIC
C-Level	Project Sponsor	Dr. Riyaz Ahamed
Project Management	Project Manager	Jia Soon
	Deputy Project Manager	Alex
	Technical Lead	Carmen
Data warehouse management and maintenance	Data warehouse administrator	Wade
	Data warehouse organizational change manager	Dave
	Database administrator	Seth
	Data warehouse maintenance developers	Ivan
	Metadata Manager	Riley
Analysis and design	Business requirements analysts	Gilbert
	User groups	Jorge
	Data warehouse architect	Dan
	Data quality analyst	Brian
	Data acquisition developer	Roberto
	Data access developer	Ramon
		Miles
Data procurement	Data quality analyst	Nathaniel
	Data acquisition developer	Ethan
	Data access developer	Lewis
IS executive sponsor	IS executive sponsor	Milton
Support roles	Iteration sponsors	Claude
	Subject matter experts	Joshua
	User support technician	Glen

3.0 Multidimensional model - Start Schema

A star schema consists of a fact table in the middle connected to a set of dimension tables. The Figures 3.1 shows the star schema of this project:

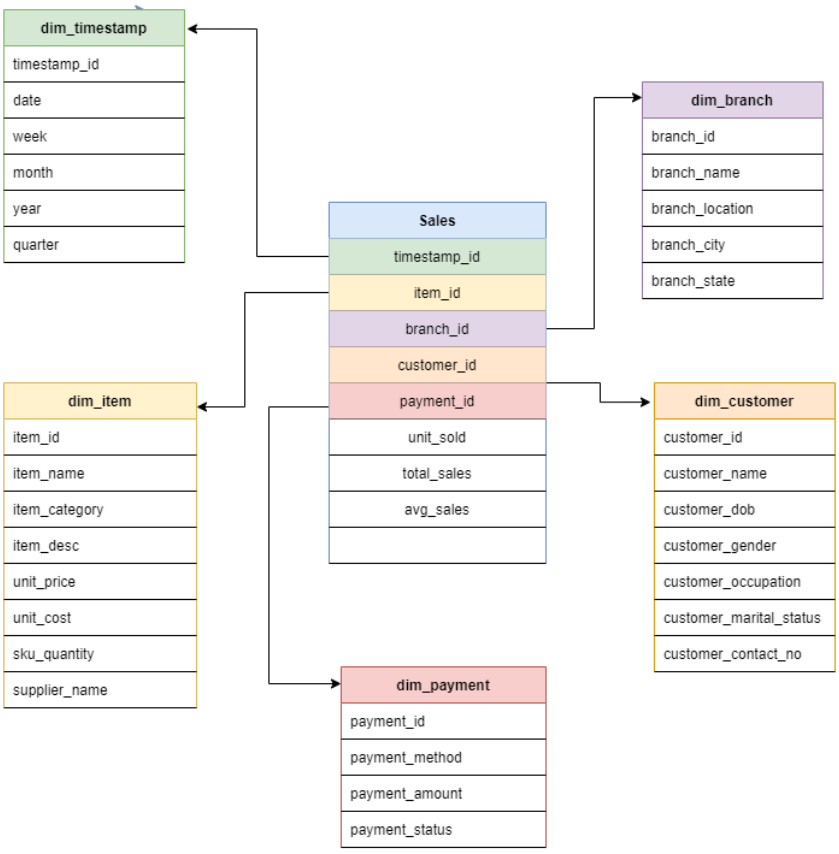

Figure 3.1: Star Schema

The star schema above consist of 1 fact table(Sales) and 5 dimension tables(dim_timestamp, dim_item, dim_payment, dim_branch and dim_customer). The following tables explain and describe about the each of the attributes.

Sales

Attributes	Data Type	Description/Remarks
timestamp_id	Big Int	The unique ID of the timestamp when the sale is created.
Item_id	Int	The unique ID of the item sold
branch_id	Int	The unique ID of the branch where the sale transaction is created
customer_id	Int	The unique ID of the customer
payment_id	Int	The unique ID of the payment
unit_sold	Int	The amount of the item sold

total_sales	Int	The total amount of sales in a single sale transaction
avg_sales	Int	The average amount of the sales

Timestamp Dimension

Attributes	Data Type	Description/Remarks
timestamp_id	Big Int	The unique ID of the timestamp when the sale is created.
date	Date	The date when the sale is created.
week	Int	The week of the year (1 year ≈52 weeks)
month	Int	The month as of the sale transaction is created
year	Int	The year as of the sale transaction is created.
quarter	Int	The quarter as of the sale transaction is created.

Item Dimension

Attributes	Data Type	Description/Remarks
item_id	Int	The unique ID of item.
item_name	String	The name if the item.
Item_category	String	The category of the item.
Item_desc	String	The description of the item.
unit_price	Int	The unit selling price of the item.
unit_cost	Int	The unit purchase cost of the item.
sku_quantity	Int	The quantity of the item left.
supplier_name	String	The supplier name for the item.

Payment Dimension

Attributes	Data Type	Description/Remarks
payment_id	Int	The unique ID for the payment of the sales.
payment_method	String	The method name of the payment.
payment_amount	Int	The payment amount received from customers.
payment_status	String	The status of the payment. (Eg:Success/Pending/Failed)

Branch Dimension

Attributes	Data Type	Description/Remarks
branch_id	Int	The unique ID of the branch.
branch_name	String	The name if the branch.
branch_location	String	The location of the branch.
branch_city	String	The city of the branch.
branch_state	String	The state of the branch.

Customer Dimension

Attributes	Data Type	Description/Remarks
customer_id	Int	The unique ID of customer.
customer_name	String	The name if the customer.
customer_dob	Date	The birthday of the customer.
customer_gender	String	The gender of the customer.
customer_occupation	String	The occupation of the customer.
customer_marital_status	String	The marital status of the customer.
customer_contact_no	Int	The contact number of the customer.

4.0 ETL Architecture Design

Extract, Transform and Load (ETL) is a process in which data is gathered from the source system, configured, and stored in a data warehouse or database. It is typically used for extracting data from multiple source systems to a single data mart or data warehouse.

For this project, we will be mainly using the Microsoft Azure services to build our ETL pipeline. As the company is also using other Microsoft products, the integrated environment for Azure makes it incredibly user friendly once we have made the move. Besides, if compared to other service providers such as AWS, Azure has more functionality in general than AWS, and it is simpler to use. AWS can be complex and is known for lots of documentation, whereas Azure uses technologies that you and your users are already accustomed to using, like Windows, Active Directory and Linux, so the transition to the cloud is less obvious.

Below are some features for the Azure services we will be using:

- **Azure Event Hub**: Fully managed, real-time data ingestion service that's simple, trusted, and scalable.
- **Azure IoT Hub**: Managed service to enable bi-directional communication between IoT devices and Azure.
- **Azure Data Factory**: Hybrid data integration service that simplifies ETL at scale.
- **Azure Data Explorer**: Fast, fully managed and highly scalable data analytics service for real-time analysis on large volumes of data streaming from applications, websites, IoT devices, and more.
- **Apache Kafka in Azure HDInsight**: Easy, cost-effective, enterprise-grade service for open-source analytics. Microsoft provides a 99.9% Service Level Agreement (SLA) on Kafka uptime.
- **Azure Data Lake Storage Gen2**: It offers low-cost storage capacity and transactions for both relational and non-relational data.

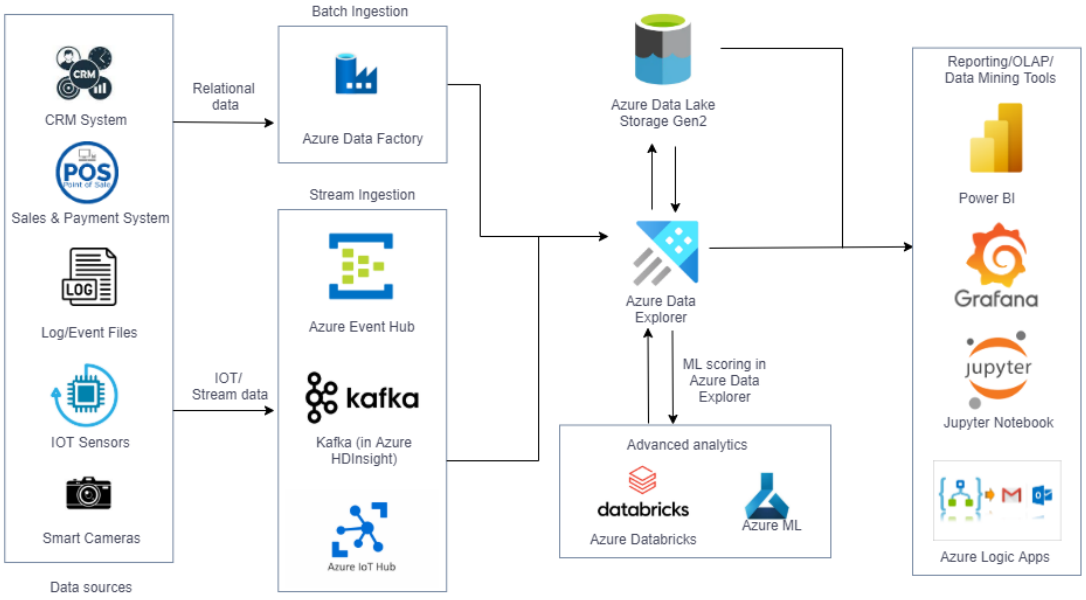

Figure 4.1: ETL Architecture Design

Figure 4.1 shows the ETL architecture design of the data warehouse. As we can see from the diagram, we have multiple sources of data from enterprise systems like CRM System and IOT devices such as sensors and cameras. These raw structured, semi-structured, and unstructured data can be extracted and ingested into Azure Data Explorer in streaming or batch mode using different methods. For example, the batch ingestion mode can be used to extract the relational data into Azure Data Explorer for analytics service via Azure Data Factory.

For relational data, at the Data Factory stage, it is where the data transformation part started. In order to perform the transformation actions, we can create a pipeline with a Data Flow activity in the data factory. After that, we can build a mapping data flow and customize the configuration to perform various transformations according to the requirement. Besides, we are also able to ingest the stream data from IOT devices into Azure Data Explorer and Azure Data Lake Storage with low-latency and high-throughput using its connectors for Azure Event Hub, Azure IoT Hub, Kafka and so on. Azure Event Hubs, Azure IoT Hub, or Kafka are able to transform and integrate a wide variety of fast-flowing streaming data such as logs, business events, and user activities.

After that, with the help from Azure Data Explorer, the transformed data can then be stored in the Azure Data Lake Storage Gen2 and ready for high-performance analytics workloads. The data lake allows us to store both relational and non-relational data, while the

Azure Data Explorer toolbox is powerful enough to gives us the end-to-end solution for data ingestion, query, visualization, and management.

For the reporting and analytics purpose, we can build near real-time analytics dashboards using Azure Data Explorer dashboards, Power BI, or Grafana. If we want to connect to your Azure Data Explorer cluster directly, we can use Jupyter notebooks, Spark connector, any TDS-compliant SQL client, and JDBC and ODBC connections.

The Azure Kusto Logic App connector enables you to run Kusto queries and commands automatically as part of a scheduled or triggered task, using the Microsoft Logic App connector. The Logic App is also helpful for the retail store to set the alert to notify the staff when the stock level for certain item is too low and should be restocked.

Lastly but not least, Azure Data Explorer can also be integrated with Azure Databricks and Azure Machine Learning to provide machine learning (ML) services. We can build ML models to learn about customer behavior and preferences, and export them to Azure Data Explorer for scoring data and further insights to drive better growth.

5.0 References

Aversa, J., Hernandez, T., & Doherty, S. (2021). Incorporating big data within retail organizations: case study approach. Journal of Retailing and Consumer Services, 60, 102447. https://www.sciencedirect.com/science/article/abs/pii/S0969698921000138

Begley, S., Hancock, B., Kilroy, T., & Kohli, S. (2020, February 18). Automation in retail: An executive overview for getting ready. McKinsey & Company. Retrieved from https://www.mckinsey.com/industries/retail/our-insights/automation-in-retail-an-executive-overview-for-getting-ready

Bunu, A. S., & Shehu, A. (2020). DATA WAREHOUSE IMPLEMENTATION FRAMEWORK FOR RETAIL BUSINESSES. GSJ, 8(2). https://www.academia.edu/44154161/DATA_WAREHOUSE_IMPLEMENTATION_FRAMEWORK_FOR_RETAIL_BUSINESSES?auto=citations&from=cover_page

Gazzola, P., Grechi, D., Martinelli, I., & Pezzetti, R. (2022). The Innovation of the Cashierless Store: A Preliminary Analysis in Italy. Sustainability, 14(4), 2034. MDPI AG. Retrieved from http://dx.doi.org/10.3390/su14042034

Güratan, I. (2005). The Design and Development of a Data Warehouse Using Sales Database and Requirements of a Retail Group (Order No. 28475592). Available from ProQuest Dissertations & Theses Global. (2567984424). https://www.proquest.com/dissertations-theses/design-development-data-warehouse-using-sales/docview/2567984424/se-2

Schögel, M., & Lienhard, S. D. (2020). Cashierless stores–the new way to the customer?. Marketing Review St. Gallen. Retrieved from https://www.alexandria.unisg.ch/259170/1/MRSG_0120_04_SPT_Lienhard_Schoegel_191107_is.pdf

Şimşek, H. (2022, March 13). Checkout Free Systems in 2022: What is it & How does it work? AIMultiple. https://research.aimultiple.com/checkout-free/

Ying, S., Sindakis, S., Aggarwal, S., Chen, C., & Su, J. (2021). Managing big data in the retail industry of Singapore: Examining the impact on customer satisfaction and organizational performance. European Management Journal, 39(3), 390–400. https://www.sciencedirect.com/science/article/abs/pii/S0263237320300530