MGS613
Project Report

Jin Chen

Business Case:

In the 21st century, the database is the marketplace. -Stan Rapp-.

A small company such as the Asian Market in Buffalo, has some issues with the data management system, which in this case they are lack data strategy. For instance, I went to the Asian Market regularly and realized that I can't find my favorite snack since last year.

Data strategy is essentially a roadmap that identifies how a company will collect, handle, manage and store content. Based on observation, it was revealed that the company still collects data through a third-party site or there is a potential that they don't collect any data from customers. Thus, they are lack collecting individuals' information, e.g satisfaction, favorite brand, financial information (if needed), frequency of shopping at the food market.

Problem and need for database application:

1. Centralized systems
    a. adequate - sufficient to properly fulfill your stated purpose
    b. relevant - has a rational link to that purpose
    c. limited to what is necessary - you do not hold more than you need for that purpose
2. Better management of human resources (HR) matters
    a. manage staff records
    b. staff hours
    c. leave
    d. benefits
    e. payroll
3. Managing customer data and relationships
    a. how their business is performing
    b. how profitable their product lines are
    c. if customers are making repeat purchases
    d. historical data can show you business trends
    e. sales records can identify valuable customers
    f. track product's expiration date can better manage sales promotions and minimize the expense of expired products.
4. Efficient inventory tracking (minimize lost sales while maximizing your opportunities for growth)
    a. too much inventory- sitting on a shelf risking wastage
    b. too little inventory- disappointing your customers
5. Planning for growth
    a. anticipating future trends and customers' needs

Introduction:

Asian Market management & online delivery database for locals in the Buffalo area. The system of this online business is where the customer can order their groceries online and the employees will gather the information or data from the market website and delivered the groceries to each of the customer doorsteps. The market allows the customer to choose a minimum of one product and a maximum of twenty products for each customer. The system will increase the profits to the market due to online orders. Also the potential increase of customers (more likely the University at Buffalo students), which have difficulty going to the market. On the other hand, the system will have all the information and details about employees which helps administrators (manager) to better manage the business.

Goal:

My goal is to deliver a database with a user interface such as the application for administration purpose, the market website and phone Apps where the customer can see all different ingredients or products that the market sell. The order will be sent to the employees who work in the “online” department. And it will be sent by a courier. The focus is to create user-friendly Apps and websites for customers. Also, the market will be guaranteed 100% fresh products.

Users:

There are three types of users that can interface with the system. There are customers, employees, and managers. The customers are the type of user that creates orders. They can simply add orders or delete orders or just visiting the website or the Apps and create a new account for their personal information. The employee is the type of user that processes the orders and preparing the orders to get delivered to each customer. As the manager, they can add new products or simply delete any out of stock products from the system. Also, adding new accounts, deleting former employees in the system, or update details for a specific employee.

Business Rules:

- A customer can place zero to many orders; an order can be placed by exactly one customer.
- One order can have three to many products; one product can be in zero to many orders.
- One employee can work in exactly one department; One department can have one to many employees
- A shipment can have one to many orders; one order can only belong to exactly one shipment.
- One shipment can only be delivered by only one transportation; one transportation can deliver with one to many shipments.
- One shipment can only be managed by only one employee; one employee can manage zero to many shipments.
- An employee can only have exactly one job; one job can be assigned to one to many employees.

Assumptions (important!!!):

1. Records table → Product line
2. Once an order is assigned to a shipment, it has to be delivered because we are doing local delivery, so we **won't** use any third-party logistics like USPS
3. Transportation table → only have types of transportation like (car, truck...)

ER Diagram:

Entity: Customers: Customer_id, name, email, address, phone_num

Order: Order_id, order_date, ***customer_id, shipment_id,*** Total_Price

Employees: Employee_id, name, email, bod, address, manager_id, ***job_id,*** ***department_id***

Record: Product_id, Order_ID, quantity, total_price

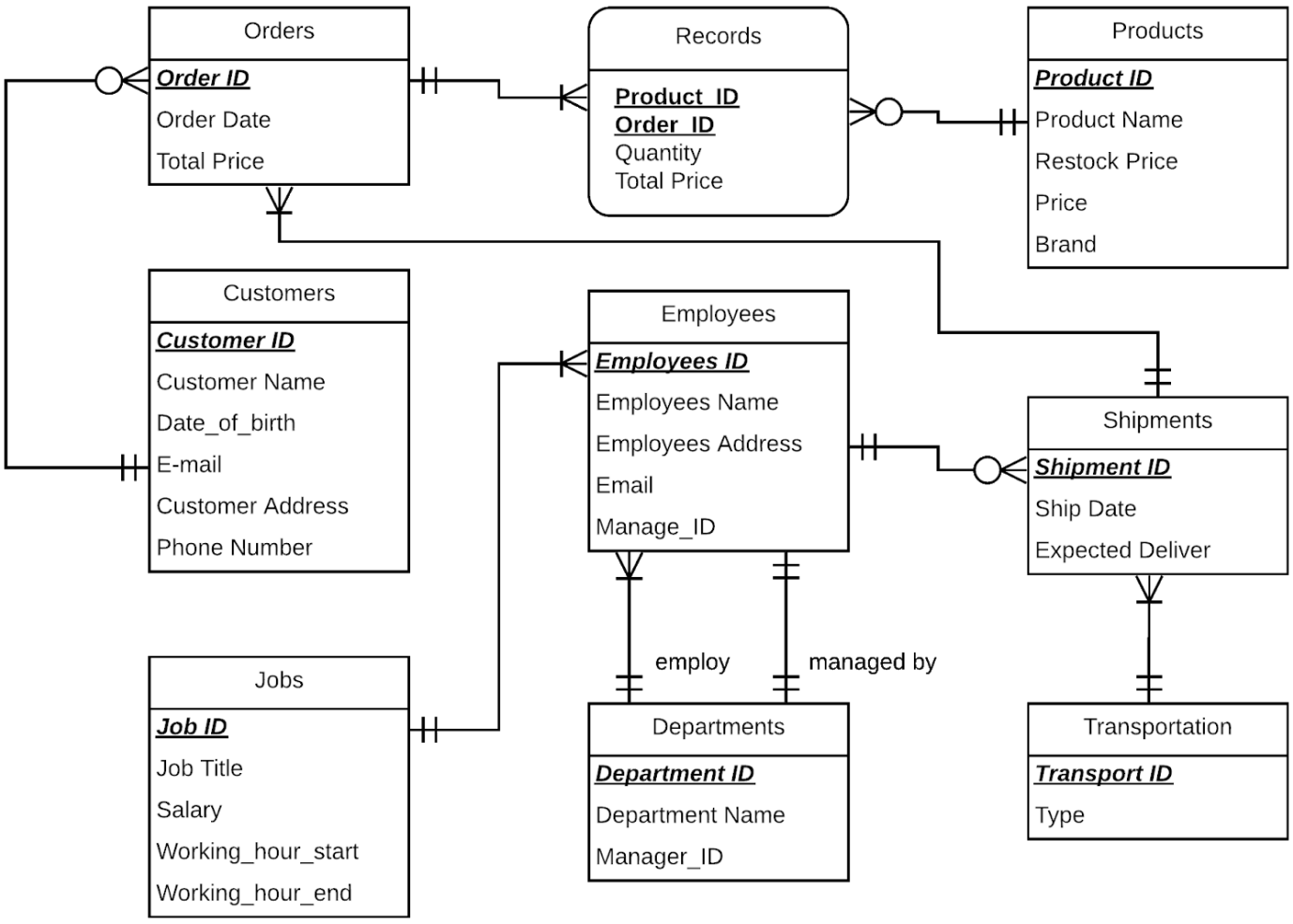
Products: Product_id, product_name, product_price, brand, restock_price

Jobs: Job_id, working_hour, job_title, salary

Departments: Department_id, department_name, manager_id

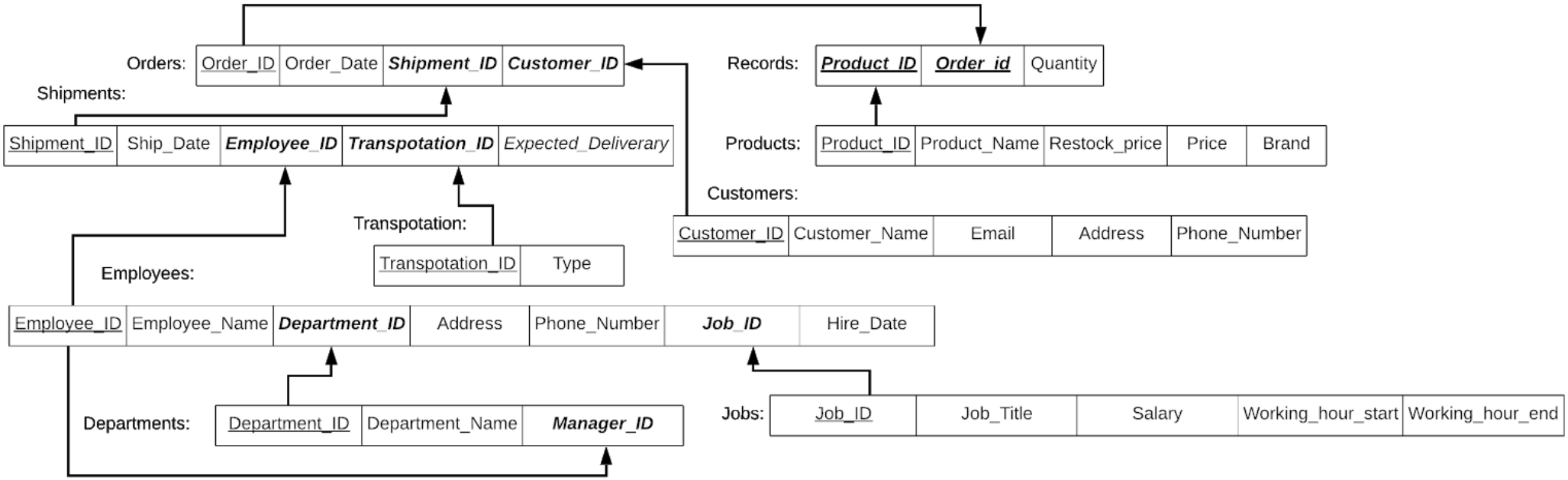
Shipment: Shipment_id, ***Transport_id***, ship_date, expected_deliver_date,***employee_id***

Transportation: Transport_id, type



Logical Models / Normalized relational schema:

Orders: Order_ID, Order_Date, ***Shipment_id***, ***Customer_id***
Records/sales: ***Product_ID***, ***Order_ID***, Quantity
Products: Product_ID, Product_Name,restock_price, Price, Brand
Shipments: Shipment_ID, Expected_Deliver_Date, Ship_Date, ***Employee_ID***, ***Transpotation_ID***
Transportation: Transpotation_ID, Type (car, truck, on_foot, bike)
Customers: Customer_ID, Customer_Name, Email, Address, Phone_Number
Employees: Employee_ID, Employee_Name, Email, Address, Hire_Date, Phone_Number, ***Department_ID***, ***Job_ID***
Jobs: Job_ID, Job_Title, Salary, Working_hour_start (‘7:00 AM’), Working_hour_end (‘8:00 PM’)
Departments: Department_ID, Department_Name, ***Manager_ID***



SQL Code:

Create Table statements:

Please see file: create_tables.sql

Insert statements:

Please see file: records_table.sql (this code only have example statements for ‘Record Table’. Other table can be inserted using Oracle SQL Developer)