

專案名稱(Project Name)

Emomo 購物網

系統需求規格書

Software Requirements Specification (SRS)

Version: 1.0

姓名	學號	E-mail
林星主	110590056	t110590056@ntut.org.tw
劉承軒	110590011	t110590011@ntut.org.tw
陳昱軒	110590012	t110590012@ntut.org.tw
許宸瑋	110590024	t110590024@ntut.org.tw
歐陽銳	110590027	t110590027@ntut.org.tw
陳思群	110590029	t110590029@ntut.org.tw

Department of Computer Science & Information Engineering
National Taipei University of Technology

10/05/2023

目錄 (Table of Contents)

Section 1	簡介 (Introduction)	1
1.1	目的 (Purpose)	1
1.2	系統名稱 (Identification)	1
1.3	概觀 (Overview)	2
1.4	符號描述 (Notation Description)	3
Section 2	系統(System)	4
2.1	系統描述 (System Description)	4
1.	2.1.1 系統架構圖 (System Architecture Diagram)	4
2.2	操作概念 (Operational Concepts or User Stories)	4
2.3	功能性需求 (Functional Requirements)	4
2.4	資料需求 (Data Requirements)	5
2.5	非功能性需求 (Non-Functional Requirements)	5
2.5.1	效能需求 (Performance Requirements)	5
2.5.2	資安需求 (Security Requirements) (if any)	5
2.6	介面需求 (Interface Requirements)	5
2.6.1	使用者介面需求 (User Interfaces Requirements)	5
2.7	其他需求 (Other Requirements)	5
2.7.1	環境需求 (Environmental Requirement)	5
2.7.2	安裝需求 (Installation Requirement)	5
2.8	商業規則與限制 (Business Rules and Integrity Constrains)	5
Section3	資料庫概念設計 (Conceptual Design of the Database)	6
Section4	邏輯資料庫綱要 (Logic Database Schema)	7
4.1	Schema of the Database	7
4.2	Possible DB operations frequencies and data volumes	9
4.3	SQL Statements Used to Construct the Schema	10
4.4	SQL Statements Used to Insert the data - data population	11
4.5	The implementation of tables in target DBMS	12
Section 5	功能性依賴(Functional Dependencies and Database Normalization)	13
Section 6	資料庫概念設計(Additional Queries and Views)	14
Glossary		16
References		17
Appendix		18

Section 1 簡介 (Introduction)

1.1 目的 (Purpose)

這學期的資料庫系統我們要做一个購物網站，**Emomo 購物網**將為顧客提供便捷的購物體驗和多樣的商品選擇，商品內容類似於家樂福及大潤發這類的大賣場。

我們的網購平台融合了傳統零售和現代電子商務的優勢，讓民眾可以待在家裡使用我們的網購平台買齊所有生活所需的商品，**Emomo 購物網**也具備購物車功能，用戶可以將所需商品添加到購物車，等所有商品皆選擇完畢且丟入購物車之後，再由購物車的頁面點選結帳，即可一次購買所有您選擇的商品。

當然我們也會使買家和賣家之間能夠即時訊息通信，以便處理訂單狀態、查詢產品信息和解決問題。

- 管理員(Administrator):
 - 維護資料庫
 - 保障交易公平性
- 會員(Member):
 - 瀏覽、搜尋已上架的票券
 - 購買並管理票券
 - 額外優惠
- 訪客(Guest):
 - 瀏覽、搜尋已上架的票券
 - 購票以 Email 寄送

1.2 系統名稱 (Identification)

主系統名稱：線上商場系統(Web-based Mall Business System, WMBS)

子系統名稱：

使用者管理系統(User manage system UMS)：

1. 使用者註冊和登錄
2. 個人資料管理
3. 訂單歷史記錄

商品管理系統(commodity manage system CMS)：

1. 商品列表和搜尋
2. 商品詳細資訊
3. 商品評論和評分

購物車和結帳系統(shoppingcart and checkout system SCS)：

1. 新增和管理購物車
2. 計算訂單總價
3. 選擇支付方式和配送選項

訂單管理系統(Order manage system OMS)：

1. 創建、編輯和取消訂單
2. 訂單狀態追蹤
3. 生成訂單確認和發貨通知

庫存管理系統(Inventory manage system IMS)：

1. 管理商品庫存
2. 預測庫存需求
3. 自動化補貨

支付和結帳系統(payment and checkout system PCS)：

1. 支付處理
2. 優惠券和促銷管理
3. 發票和收據生成

客戶支援系統(User support system USS)：

1. 客戶查詢和反饋
2. 線上聊天支援
3. 常見問題解答

數據分析和報告系統(?)：

1. 分析銷售數據
2. 定制報告生成
3. 使用者行為分析

安全性和隱私系統(safety and privacy system SPS)：

1. 使用者數據保護

後台管理系統(backstage manage system BMS)：

1. 管理商品、使用者和訂單
2. 監控系統性能和健康狀況
3. 數據備份和恢復

1.3 概觀 (Overview)

為了方便開發、提高效率以及未來的可維護性，我們採用前後端分離的方式，將前後的程式視為兩個獨立的專案，前端僅負責 View 和 Control，後端負責 Model，最後再進行整合。

前端部分我們使用常見的 bootstrap 作為前端開發工具，它提供了完整的 css grid system 可以輕鬆打造響應式 Layout，並且有良好的瀏覽器支援度；後端程式語言我們使用 PHP，而資料庫考量到資訊安全的部分，我們選用 MySQL。

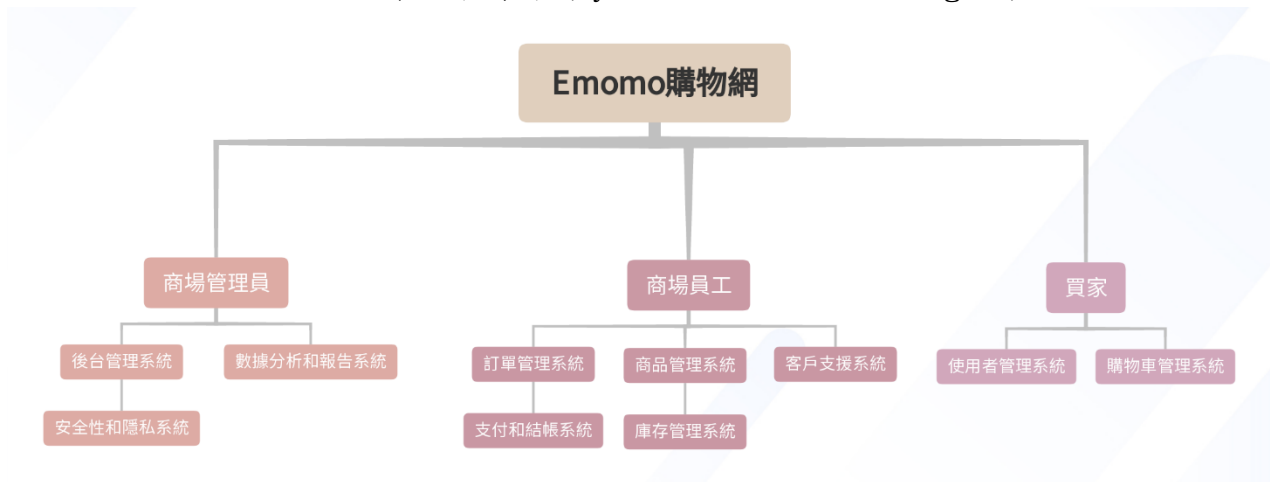
1.4 符號描述 (Notation Description)

編號代號	詳細名字
WMBS-F-nnn	線上商場系統-功能性需求-編號
WMBS-N-nnn	線上商場系統-非功能性需求-編號
UMS	使用者管理系統
CMS	商品管理系統
SCS	購物車和結帳系統
OMS	訂單管理系統
IMS	庫存管理系統
PCS	支付和結帳系統
USS	客戶支援系統
SPS	安全性和隱私系統
BMS	後台管理系統

Section 2 系統(System)

2.1 系統描述 (System Description)

1. 2.1.1 系統架構圖 (System Architecture Diagram)



2.2 操作概念 (Operational Concepts or User Stories)

客戶端：

當客戶完成註冊後，他們將能夠進入使用者管理系統（UMS）並登入。一旦登入，他們可以瀏覽商品管理系統（CMS），以檢視可能感興趣的商品。

如果他們決定購買商品，他們可以使用購物車和結帳系統（SCS）來提交訂單，然後使用支付和結帳系統（PCS）來完成付款流程。

如果客戶有任何問題或需要支援，他們可以隨時使用客戶支援系統（USS）來提出詢問或諮詢。

員工端：

當客戶完成訂單後，他們可以在訂單管理系統（OMS）中確認訂單的狀態。同時，庫存管理系統（IMS）可供使用以確認商品庫存狀況。

如果客戶有任何疑問，他們也可以隨時透過客戶支援系統（USS）提出詢問或反饋。

管理員：

在使用者管理系統（UMS）中，可以有效地管理客戶名單。同時，後台管理系統（BMS）可用於維護整個系統的運作。

2.3 功能性需求 (Functional Requirements)

需求編號	需求描述
WMBS-F-001	可註冊會員
WMBS-F-002	可搜尋與推薦喜好商品
WMBS-F-003	可新增或刪除商品至購物車
WMBS-F-004	可一鍵清除購物車
WMBS-F-005	可選擇付款方式
WMBS-F-006	可選擇取物管道
WMBS-F-007	可追蹤訂單進度
WMBS-F-008	可選擇優惠方案
WMBS-F-009	可統計銷售資訊
WMBS-F-0010	可編輯商品資訊

2.4 資料需求 (Data Requirements)

1. 使用者資料需求：姓名、用戶名稱、密碼（加密存儲）、電子郵件地址、連絡資訊、使用者類型（客戶、員工、管理員）。
2. 商品資料需求：商品名稱、描述、價格、圖片或影像、商品分類、庫存數量、上架和下架狀態。
3. 訂單資料需求：訂單號碼、用戶關聯（哪個客戶下的單）、商品購買詳情、訂單狀態（已支付、處理中、已完成等）、付款方式、發貨和運送資訊、訂單日期和時間。
4. 客戶支援資料需求：客戶詢問和反饋記錄、支援申請的類型、支援處理狀態、支援人員的分派和回應。
5. 安全性和隱私資料需求：使用者帳戶安全性資料（例如：密碼）、用戶隱私偏好和同意。
6. 數據分析和報告資料需求：使用者活動記錄、銷售數據、用戶反饋和評論、系統性能監測數據。

2.5 非功能性需求 (Non-Functional Requirements)

2.5.1 效能需求 (Performance Requirements)

需求編號	需求描述
WBBS-N-101	使用者瀏覽時，頁面讀取不大於 5 秒
WBBS-N-102	使用者搜尋時，搜尋時間不大於 3 秒
WBBS-N-103	搜尋時，關鍵字出現時間不大於 2 秒
WBBS-N-104	沒現貨時商品會顯示
WBBS-N-105	商品資訊更新時間不大於 5 秒
WBBS-N-106	顯示商品庫存量
WBBS-N-107	顯示商品運送時間

2.5.2 資安需求 (Security Requirements) (if any)

保護使用者數據、商品資料。

2.6 介面需求 (Interface Requirements)

2.6.1 使用者介面需求 (User Interfaces Requirements)

需求編號	需求描述
WMBS-N-201	商品瀏覽介面
WMBS-N-202	購物車介面
WMBS-N-203	搜尋介面
WMBS-N-204	會員登入及註冊介面
WMBS-N-205	訂單查詢介面
WMBS-N-206	商品管理介面
WMBS-N-207	銷售報表介面
WMBS-N-208	商品結帳介面

2.7 其他需求 (Other Requirements)

2.7.1 環境需求 (Environmental Requirement)

Windows、Mac、Android、ios

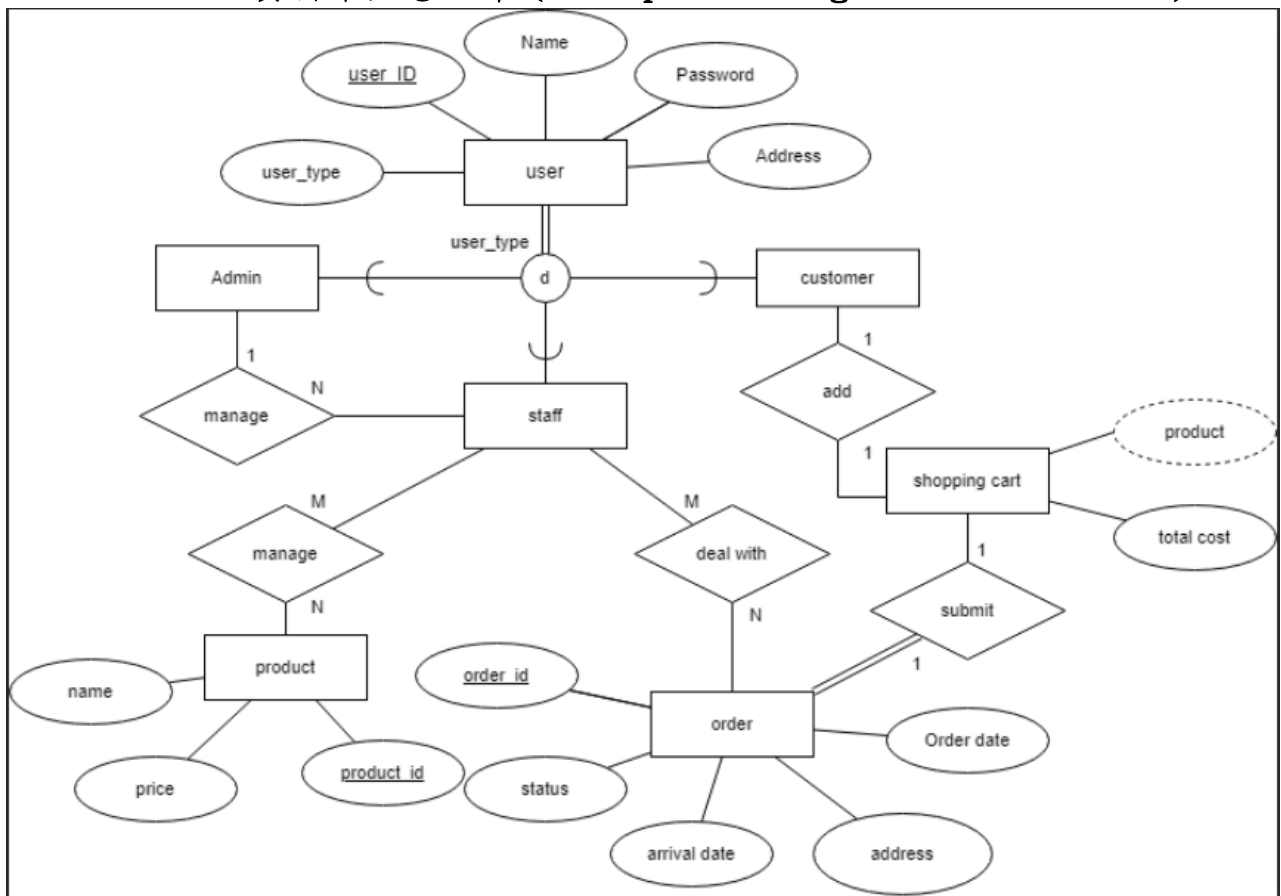
2.7.2 安裝需求 (Installation Requirement)

容量達到 1GB 以上。

2.8 商業規則與限制 (Business Rules and Integrity Constrains)

購買國外商品需要做資料認證。

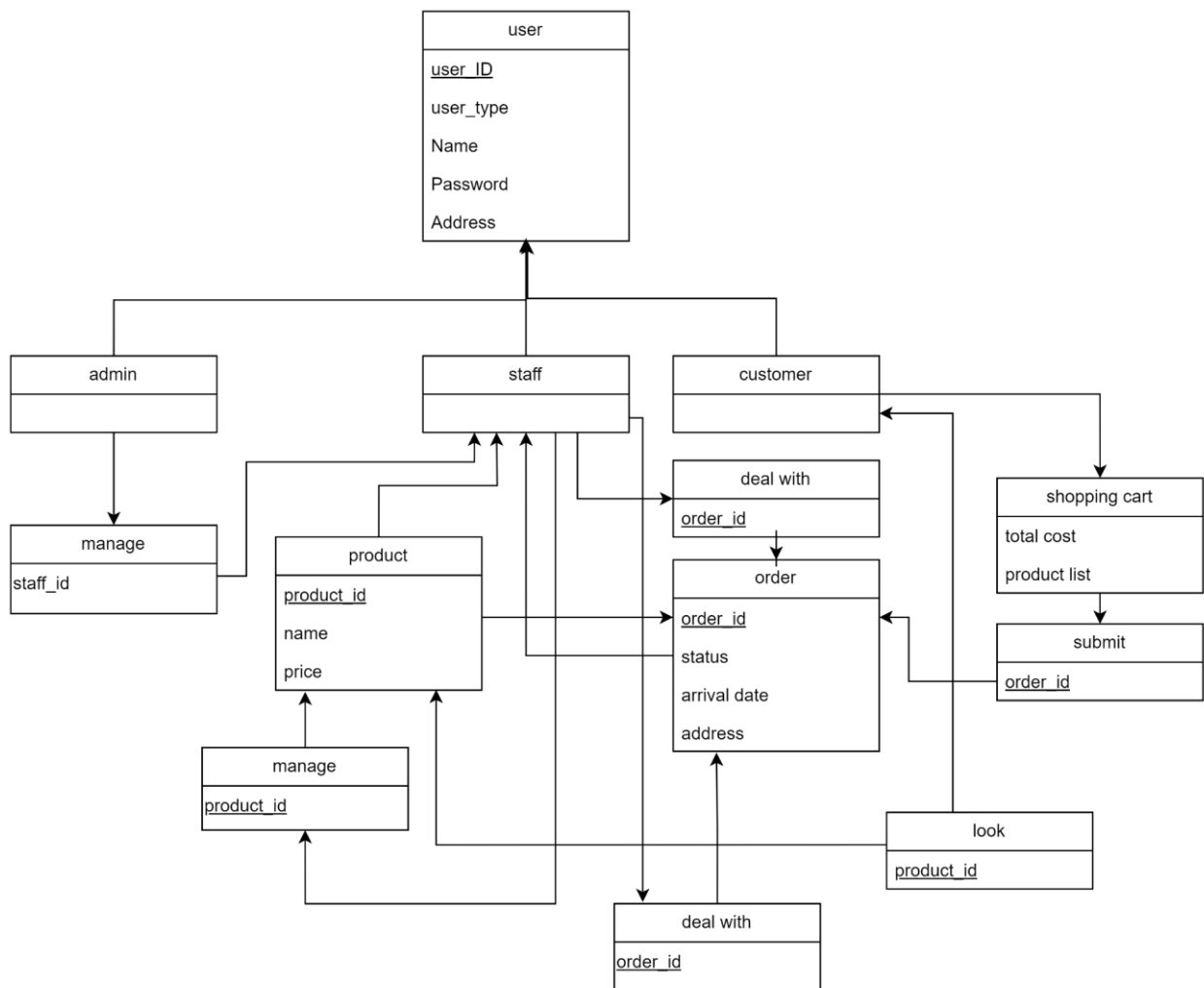
Section3 資料庫概念設計 (Conceptual Design of the Database)



Section4 邏輯資料庫綱要 (Logic Database Schema)

4.1 Schema of the Database

+ field: type



User				
Description：所有使用者				
Attribute	Domain Type	Key	Nullable	Description
user_ID	Integer	Primary	NO NULL	使用者 ID
user_type	string		NO NULL	使用者型態
Name	string		NO NULL	使用者名
Password	string		NO NULL	密碼
address	string		NO NULL	地址

product				
Description：商品				
Attribute	Domain Type	Key	Nullable	Description
product_ID	Integer	Primary	NO NULL	商品 ID
name	string		NO NULL	商品名
price	Integer		NO NULL	價格

order				
Description：訂單				
Attribute	Domain Type	Key	Nullable	Description
order_id	Integer	Primary	NO NULL	訂單 ID
status	string		NO NULL	訂單狀態
arrival date	string		NO NULL	抵達日期
address	string		NO NULL	送達地址

shopping cart				
Description：購物車				
Attribute	Domain Type	Key	Nullable	Description
total price	Integer		NO NULL	總價格
product	list		NO NULL	商品列表

4.2 Possible DB operations frequencies and data volumes

Table	可能操作	預估使用頻率	表格資料量	系統負擔
User	登入身分驗證	200	5000	1000000Query/day
Admin	管理 Staff 成員	2	20	40 次 Insert/Day
Admin	更新 Staff 資料	2	20	40 次 Update/Day
Staff	讀取商店資料	2000	50	100000 次 Query/day
Staff	修改商店資料	1	50	50 次 Update/day
Customer	瀏覽商品	500	5000	2500000 次 Query/day
Customer	下單商品	50	5000	250000 次 Insert/day
Manage	紀錄商品和負責人	10	500	5000 次 Query/day
Shopping_Cart	新增購物車內容	2	don't care	2 次 Insert/Day
Shopping_Cart	更新購物車內商品數量	3	30	60 次 Query/Day 3 次 Upday/Day
Shopping_Cart	刪除購物車內容	1	30	30 次 Query/Day 1 次 Upday/Day
Product	查詢商品資料	300	300	90000 次 Query/Day
Product	新增商品	10	don't care	10Inster/Day
Product	更新商品資料	600	300	180,000 次 Query/Day 600 次 Update/Day
Order	新增訂單	100	20	2000 次 Insert/day
Order	修改訂單	100	20	2000 次 Update/day
Deal_with	新增訂單的負責人資料	20	don't care	20 次 Insert/day
Coupon	新增優惠券資料	3	100	3000 次 Insert/day
Coupon	更新優惠券資料	3	100	3000 次 Update/day
Coupon	新增優惠券號碼	100	1000	100000 次 Query/day

4.3 SQL Statements Used to Construct the Schema

```
CREATE TABLE USER (
  ID INT PRIMARY KEY,
  Type VARCHAR(255),
  Name VARCHAR(255),
  Password VARCHAR(255),
  Address VARCHAR(255),
  Email VARCHAR(255)
);

CREATE TABLE ADMIN (
  User_ID INT PRIMARY KEY,
  FOREIGN KEY (User_ID) REFERENCES USER(ID) ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE STAFF (
  User_ID INT PRIMARY KEY,
  FOREIGN KEY (User_ID) REFERENCES USER(ID) ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE CUSTOMER (
  User_ID INT PRIMARY KEY,
  FOREIGN KEY (User_ID) REFERENCES USER(ID) ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE SHOPPING_CART (
  Shopping_ID INT PRIMARY KEY,
  Product VARCHAR(255),
  Total_Cost DECIMAL(10, 2),
  Customer_User_ID INT,
  FOREIGN KEY (Customer_User_ID) REFERENCES USER(ID) ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE COUPON (
  Coupon_ID INT PRIMARY KEY,
  Name VARCHAR(255),
  Gift_Code VARCHAR(255),
  Rule VARCHAR(255),
  Date DATE, -- 'YYYY-MM-DD'
  Discount DECIMAL(5, 2),
  Staff_ID INT,
  FOREIGN KEY (Staff_ID) REFERENCES USER(ID) ON UPDATE CASCADE ON DELETE SET NULL
);

CREATE TABLE `ORDER` ( -- 這裡因為oder是保留字(排序指令)
  Order_ID INT PRIMARY KEY,
  Status VARCHAR(255),
  Arrival_Date DATE,
  Order_Date DATE,
  Address VARCHAR(255),
  Shopping_Cart_ID INT,
  Coupon_ID INT,
  FOREIGN KEY (Shopping_Cart_ID) REFERENCES SHOPPING_CART(Shopping_ID) ON UPDATE CASCADE ON DELETE
SET NULL,
  FOREIGN KEY (Coupon_ID) REFERENCES COUPON(Coupon_ID) ON UPDATE CASCADE ON DELETE SET NULL
);

CREATE TABLE PRODUCT (
  Product_ID INT PRIMARY KEY,
  Name VARCHAR(255),
  Price DECIMAL(10, 2), -- 假设 Price 是一个带小数的数字
  Category VARCHAR(255)
);

CREATE TABLE MANAGE (
  Staff_ID INT,
  Product_ID INT,
  PRIMARY KEY (Staff_ID, Product_ID),
  FOREIGN KEY (Staff_ID) REFERENCES STAFF(User_ID) ON UPDATE CASCADE ON DELETE CASCADE,
  FOREIGN KEY (Product_ID) REFERENCES PRODUCT(Product_ID) ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE DEAL_WITH (
  Order_ID INT,
  Staff_ID INT,
  PRIMARY KEY (Order_ID, Staff_ID),
  FOREIGN KEY (Order_ID) REFERENCES `ORDER`(Order_ID) ON UPDATE CASCADE ON DELETE CASCADE,
  FOREIGN KEY (Staff_ID) REFERENCES STAFF(User_ID) ON UPDATE CASCADE ON DELETE CASCADE
);
```

4.4 SQL Statements Used to Insert the data – data population

```
-- 示例 USER 数据
INSERT INTO USER (ID, Type, Name, Password, Address, Email) VALUES
(1, 'Admin', 'Admin1', 'adminpass', 'Admin Address', 'admin@example.com'),
(2, 'Staff', 'Staff1', 'staffpass', 'Staff Address 1', 'staff1@example.com'),
(3, 'Staff', 'Staff2', 'staffpass', 'Staff Address 2', 'staff2@example.com'),
(4, 'Customer', 'Customer1', 'customerpass', 'Customer Address 1', 'customer1@example.com'),
(5, 'Customer', 'Customer2', 'customerpass', 'Customer Address 2', 'customer2@example.com');

-- 示例 ADMIN 数据
INSERT INTO ADMIN (User_ID) VALUES (1);

-- 示例 STAFF 数据
INSERT INTO STAFF (User_ID) VALUES (2), (3);

-- 示例 CUSTOMER 数据
INSERT INTO CUSTOMER (User_ID) VALUES (4), (5);

-- 示例 SHOPPING_CART 数据
INSERT INTO SHOPPING_CART (Shopping_ID, Product, Total_Cost, Customer_User_ID) VALUES
(1, 'Product A', 50.00, 4),
(2, 'Product B', 30.00, 5);

-- 示例 COUPON 数据
INSERT INTO COUPON (Coupon_ID, Name, Gift_Code, Rule, Date, Discount, Staff_ID) VALUES
(1, 'Coupon 1', 'GIFT123', 'Rule 1', '2023-01-01', 10.00, 2),
(2, 'Coupon 2', 'GIFT456', 'Rule 2', '2023-02-01', 5.00, 3);

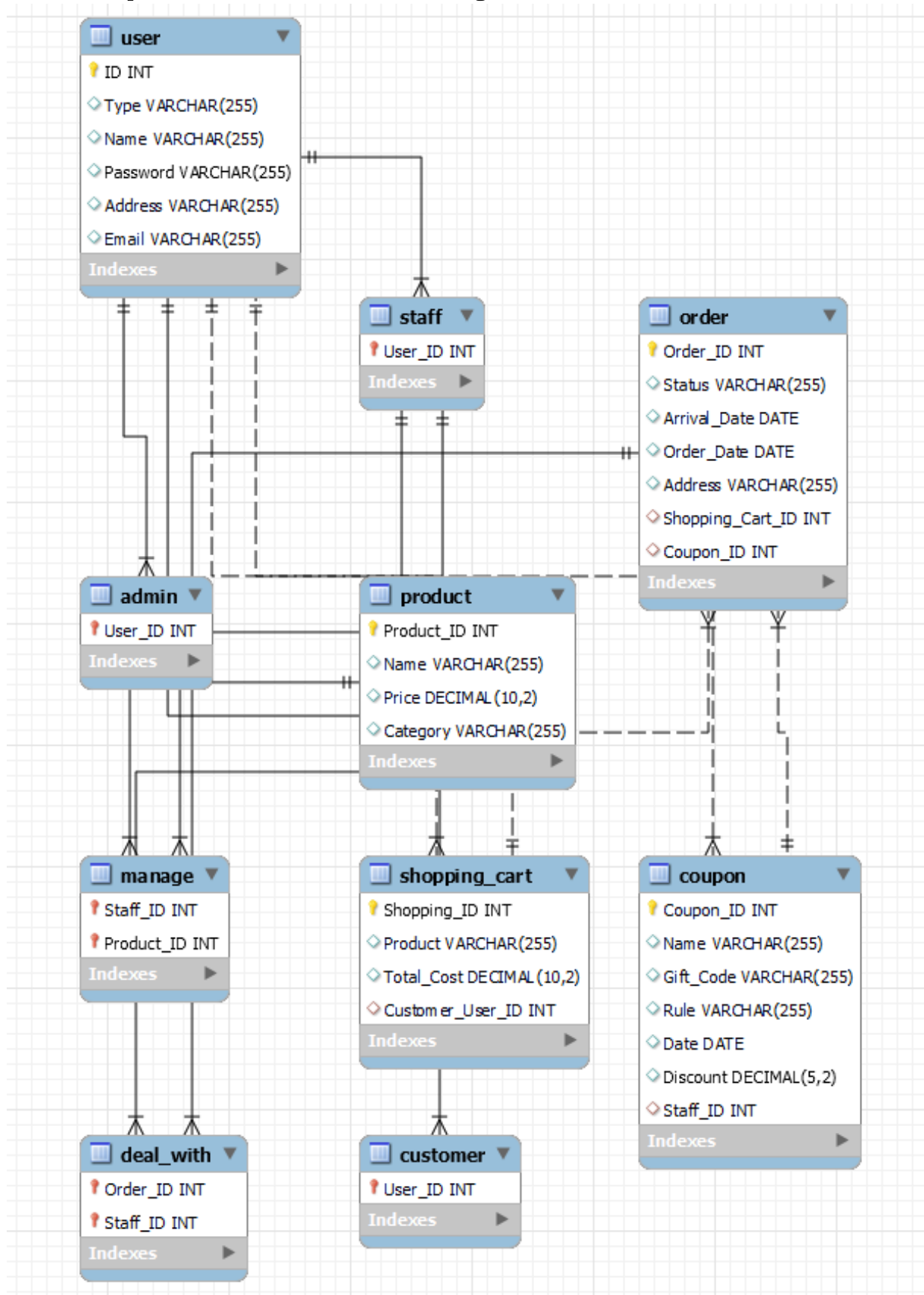
-- 示例 ORDER 数据
INSERT INTO `ORDER` (Order_ID, Status, Arrival_Date, Order_Date, Address, Shopping_Cart_ID, Coupon_ID)
VALUES
(1, 'Shipped', '2023-03-01', '2023-02-15', 'Shipping Address 1', 1, 1),
(2, 'Processing', NULL, '2023-03-01', 'Shipping Address 2', 2, 2);

-- 示例 PRODUCT 数据
INSERT INTO PRODUCT (Product_ID, Name, Price, Category) VALUES
(1, 'Product A', 50.00, 'Category 1'),
(2, 'Product B', 30.00, 'Category 2'),
(3, 'Product C', 20.00, 'Category 1');

-- 示例 MANAGE 数据
INSERT INTO MANAGE (Staff_ID, Product_ID) VALUES
(2, 1),
(2, 2),
(3, 3);

-- 示例 DEAL_WITH 数据
INSERT INTO DEAL_WITH (Order_ID, Staff_ID) VALUES
(1, 2),
(2, 3);
```

4.5 The implementation of tables in target DBMS



Section 5 功能性依賴(Functional Dependencies and Database Normalization)

Section 6 資料庫概念設計(Additional Queries and Views)

1. 首頁(Home Page)

EMOMO

Cart Search EN

SHOP PAGES BLOG ELEMENTS

類別

智慧型手機

APPLE

SAMSUNG

SONY

展開

零食

Apple iPhone 15 Pro 256G

\$39900

SAMSUNG Galaxy S23 Ultra 256G

\$31380

SONY Xperia IV 256G

\$30812

Panasonic 10吋遙控空氣循環扇

\$3999

CHIMEI 奇美 14吋變頻壁扇電風扇

\$2540

TATUNG 復古小電扇

USB

TATUNG 復古紀念小電扇

\$790

樂事薯片混合包

\$116

湖池屋 卡樂媽久洋芋片

\$58

卡迪那95℃北海道風味薯條

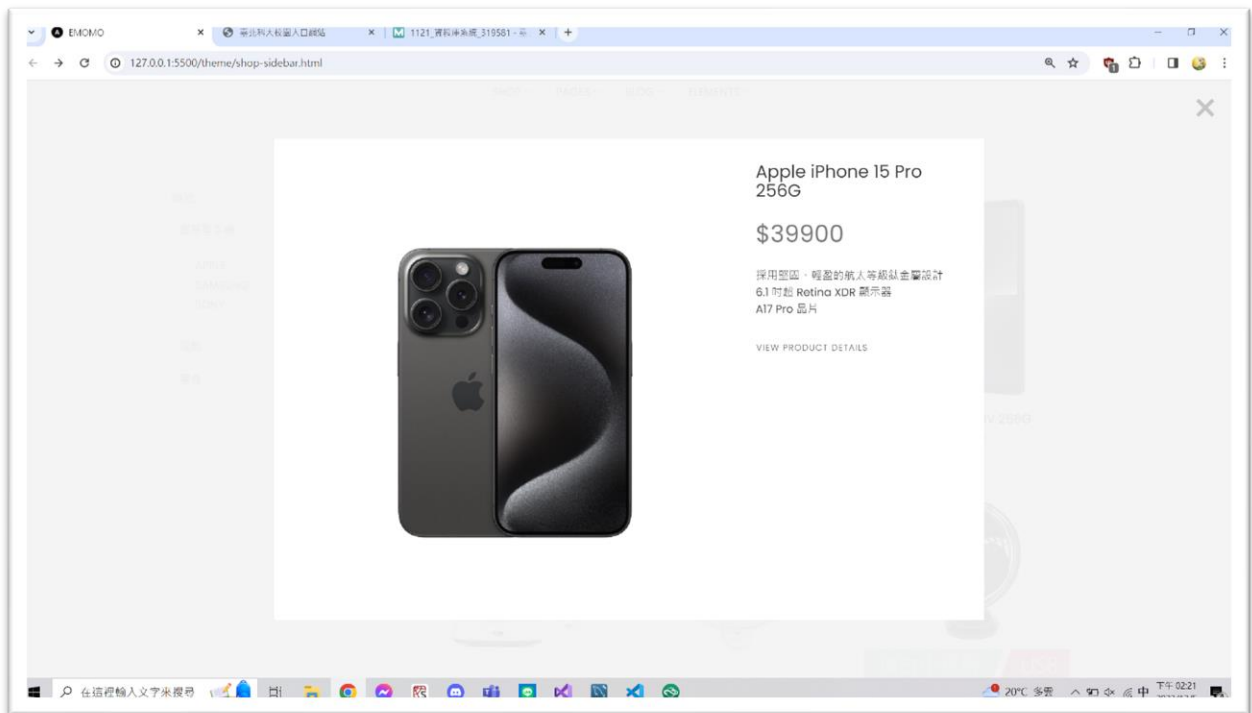
\$79

github 連結

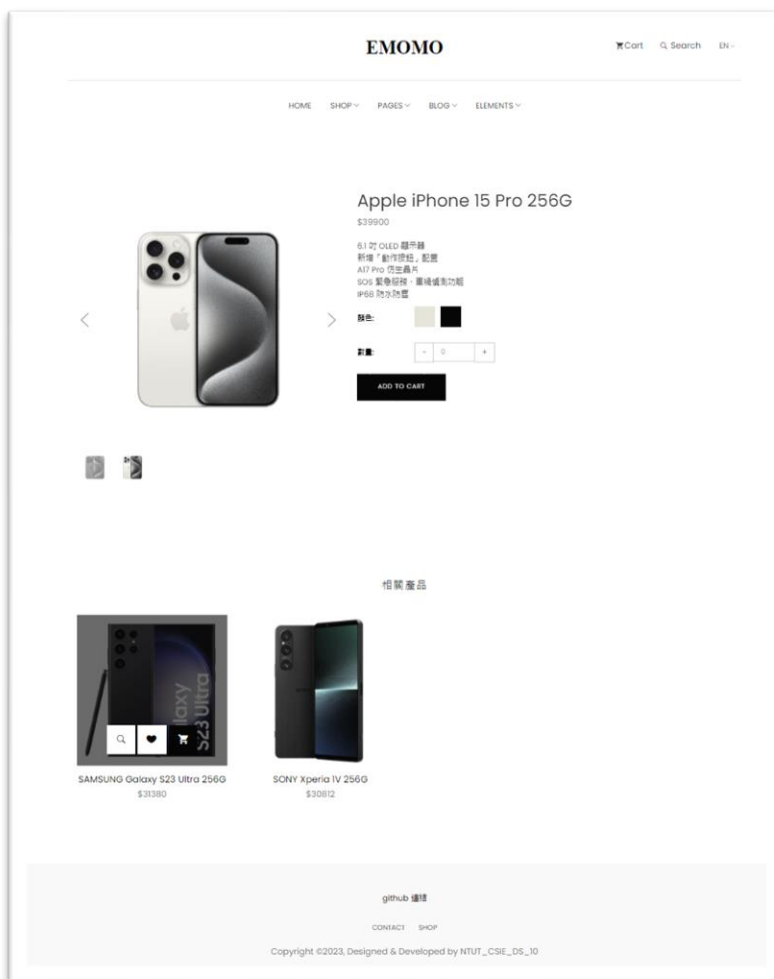
CONTACT SHOP

Copyright ©2023, Designed & Developed by NTUT_CSIE_DS_10

2. 商品預覽浮動視窗(Product preview pop-up windows)



3. 商品詳細介紹頁面 (Product introduction Page)



Glossary

References

Appendix