

# BuyGO：需求功能說明書

組員：508062334 陳彥志

510171292 吳家驊

511172176 李則霖

## 主旨

「BuyGO」是一個線上的訂購系統，目標是提供一個簡單且全面的訂購體驗給顧客。該系統將採用 JavaScript 作為前端開發語言，並選用 MySQL 作為後端資料庫。透過 JavaScript 讓我們開發出具有動態效果和互動性的網頁、並使用 MySQL 製作高效能的關聯式資料庫管理系統，讓我們儲存和處理大量的訂單和商品資料。透過學習網頁開發的流程，我們期望能夠掌握如何利用 JavaScript 和資料庫技術來實作一個完整的線上訂購系統。該系統將包含商品購買頁面、後台管理以及會員系統等相關功能。透過清晰且友善的介面設計，讓顧客可以輕鬆地瀏覽、選擇和支付商品。店家也可以使用後台管理功能，讓管理者方便地新增、修改和刪除商品資訊。會員系統部分則使顧客可以註冊、登入和查看自己的訂單紀錄。

## 使用者需求情境

在「BuyGO」的使用者需求情境中，我們可以看到顧客和店家共同面對的核心需求。顧客，如小明，尋求一個簡便且高效的線上購物體驗，他希望能夠輕鬆註冊並管理自己的帳戶，無縫地瀏覽和選擇商品，並能在購物車中靈活調整選擇，最後透過安全的支付系統完成購買。同時，對於訂單的實時追蹤與管理對他來說至關重要。

另一方面，店家如王女士則需要能夠有效管理商品庫存，輕鬆更新和管理商品信息，並且實時查看和處理顧客訂單。她還期望能通過系統獲取銷售數據，以便更好地理解市場趨勢和客戶需求。

綜合這些需求，「BuyGO」系統設計了一個全面的解決方案，旨在提供給顧客和店家一個直觀、高效且互動性強的線上購物平台。這不僅涵蓋了商品的瀏覽和購買過程，還包括了會員系統和後台管理功能，確保雙方都能在同一個平台上滿足其所有的需求。

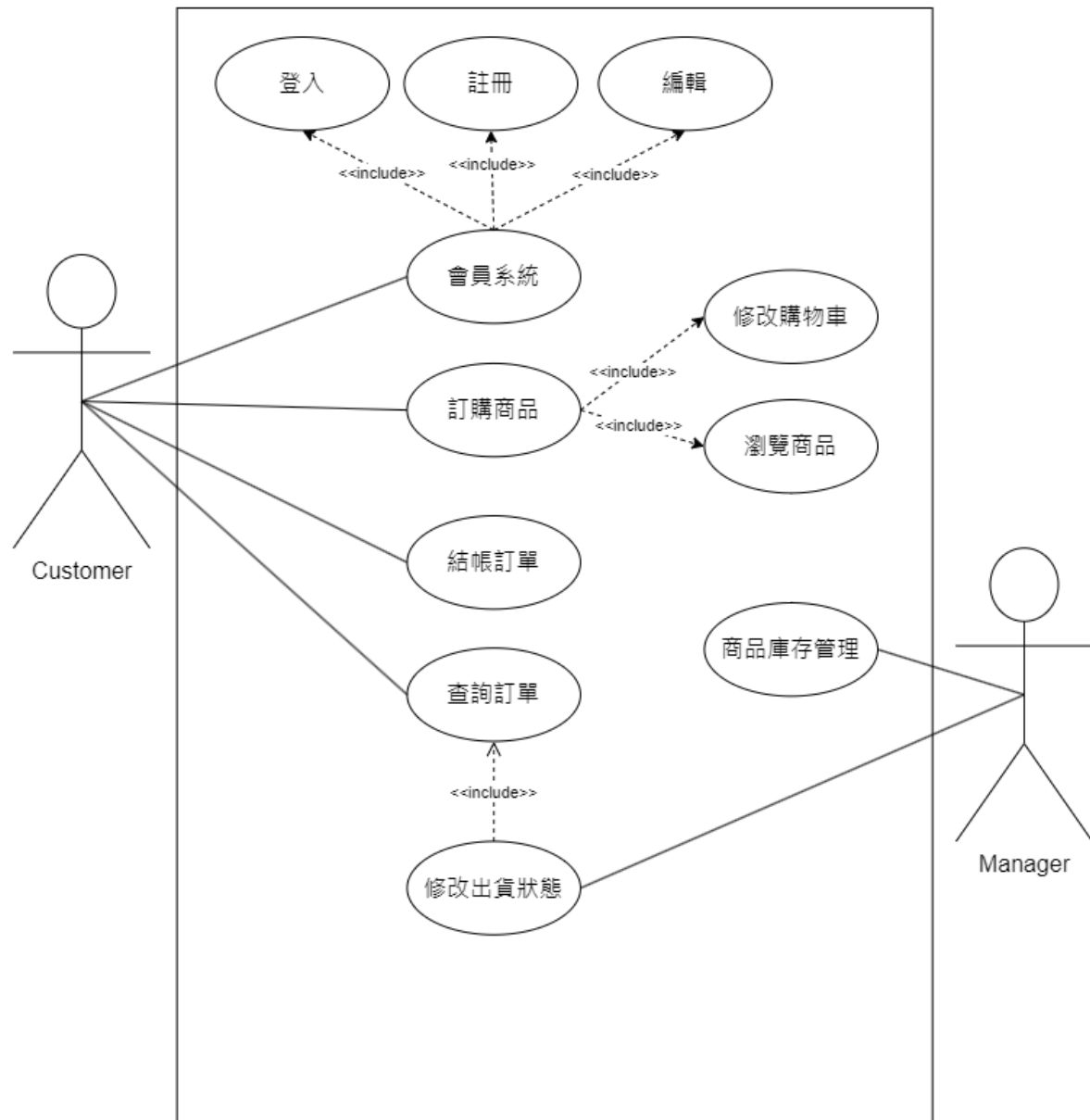
## 應用說明

將提供顧客註冊會員帳號、登入會員帳號、訂購商品、增修訂單功能。以直觀且易於使用的用戶介面使顧客能夠方便地瀏覽商品並下單，並透過資料庫的建構存儲顧客資訊及設定訂單內表格的關聯性。透過商品、購物車及結帳頁面的流程設計，顧客可隨意新增商品至購物車，並選擇自身喜好的付款與運送方式後完成訂單。並將提供完善的會員系統，以註冊、登入為基本需求，添加會員資料編輯頁面，使顧客能即時修改會員資訊。顧客亦可隨時登入自身帳號查詢/追蹤過往訂單紀錄或修改尚未出貨訂單。店家部分則可以透過管理者權限帳號登入後台，並查看顧客訂單更新商品出貨狀態，或可至商品資訊頁面上架或刪除商品並更新商品可售數量。

## 功能需求

1. **登入（帳號密碼）**：允許會員使用已註冊的帳號密碼進行登入，驗證成功後導向主頁面。
2. **註冊（帳號、密碼、會員名稱、電話）**：新用戶可以創建新帳戶，需要提供帳號、密碼、會員名稱及聯絡電話。
3. **編輯會員資料（編輯名稱、電話、修改密碼）**：允許會員編輯或更新其個人資訊。
4. **顯示所有商品**：主頁面展示商店中的所有可購商品。
5. **訂購商品（選擇數量加到購物車）**：用戶選擇商品並指定數量後，可以將其加入購物車。
6. **顯示已訂購商品**：列出用戶在購物車中的所有商品及其數量。
7. **修改訂購數量**：允許用戶調整購物車中商品的購買數量。
8. **刪除訂購商品**：用戶可以從購物車中移除不再需要的商品。
9. **顯示購物車總金額與總數量**：顯示購物車中所有商品的總計金額和總數量。
10. **選擇交易方式（信用卡、貨到付款）**：用戶在結帳時可以選擇其喜好的付款方式。
11. **選擇運送方式（宅配、店到店）**：用戶可以根據需要選擇運送方式。
12. **送出訂單（顯示完成訂購訊息）**：用戶完成結帳流程後，系統將確認訂單並給予用戶完成的訊息提示。
13. **查看已完成訂購商品**：會員可以查看其過去已完成的訂購商品清單。
14. **後台系統**：商家能夠使用管理者權限進入後台自行上架或刪除商品及修改商品可售數量，並能夠更新顧客訂單的出貨狀態。

## 功能設計



**(1) 顧客：**

- 1. 註冊會員：**可輸入會員資料並申請帳號密碼。

對應：註冊（帳號、密碼、會員名稱、電話）

- 2. 登入會員：**使顧客能夠登入網站購物。

對應：登入（帳號密碼）

- 3. 編輯會員資料：**可修改會員資料(姓名、電話、地址)或密碼。

對應：編輯會員資料（編輯名稱、電話、修改密碼）

- 4. 訂購商品：**讓顧客能夠自由選購商品及修改購物車內已選商品。

對應：顯示所有商品、顯示已訂購商品、修改訂購數量、刪除訂購商品

- 5. 結帳訂單：**確認總金額、選擇付款方式、選擇送貨方式，並送出訂單。

對應：顯示購物車總金額與總數量、選擇交易方式（信用卡、貨到付款）、選擇運送方式（宅配、店到店）、送出訂單（顯示完成訂購訊息）

- 6. 查詢訂單：**提供顧客查詢已訂購商品與目前訂單運送狀態。

對應：查看已完成訂購商品

**(2) 管理者：**

- 1. 登入管理者帳號：**使用管理者權向帳號與密碼登入作登入。

對應：使用管理者權限

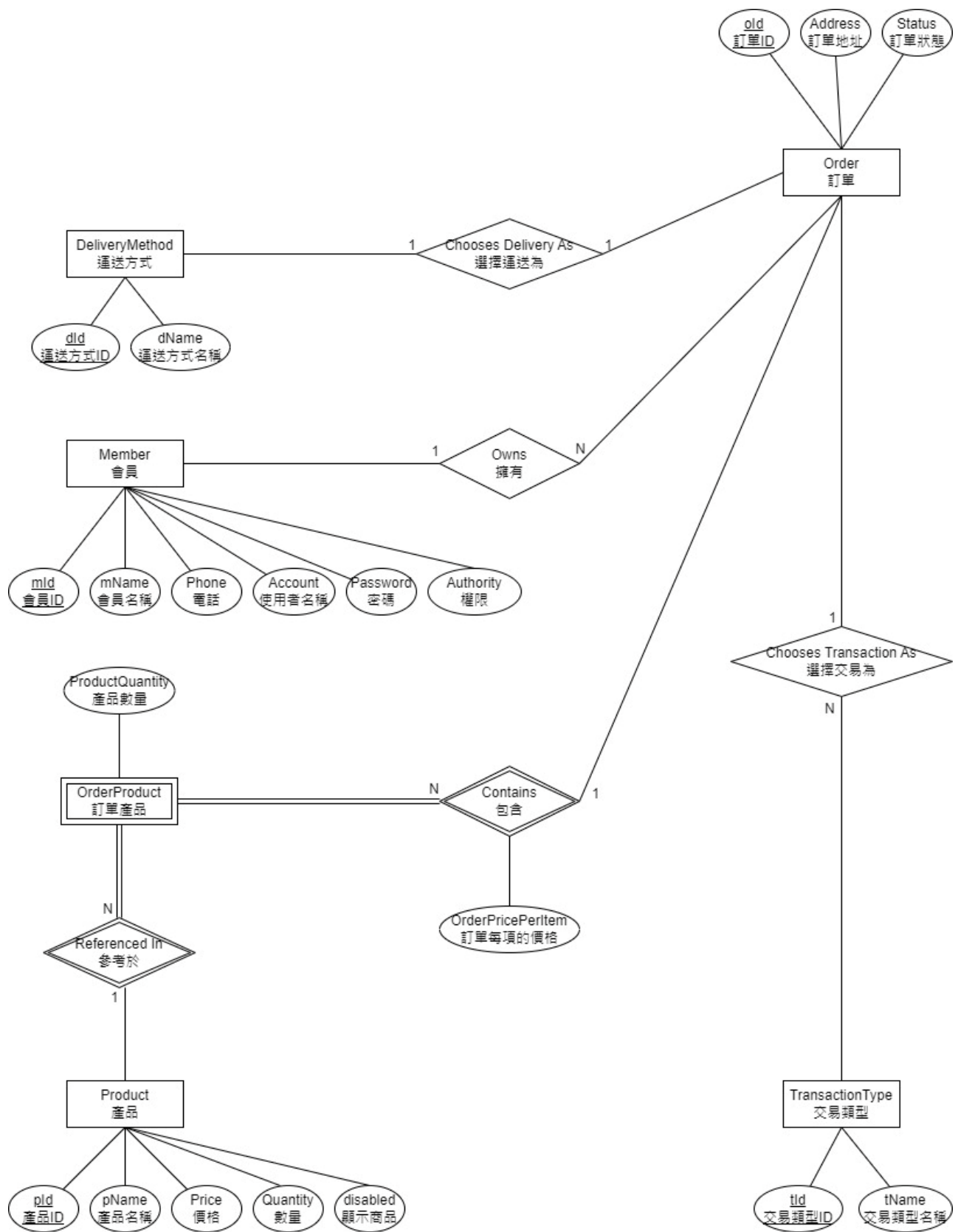
- 2. 商品庫存管理：**可自由上架或下架商品，並可修改商品可購數量。

對應：上架或刪除商品、修改可購數量

- 3. 修改出貨狀態：**可於顧客訂單修改出貨狀態。

對應：更新顧客訂單的出貨狀態

## ER Model 設計



## ER Model 關係說明

- **配送狀態**：一個狀態可以對應多個訂單。
- **運送進度**：一個運送進度可以對應多個訂單。
- **運送方式**：一個運送方式可以對應多個訂單，一個訂單也對應一種運送方式。
- **會員**：一個會員可以擁有多個訂單。
- **訂單商品**：一個訂單商品會參考到一種商品。
- **訂單**：一個訂單可能包含多個訂單商品。
- **交易類型**：一個訂單可以選擇多種交易類型，且多種交易類型也可能對應到同一個訂單。

# BuyGO：系統設計規格書

組員：508062334 陳彥志

510171292 吳家驊

511172176 李則霖

102/12/17 第二版修訂

## 主旨

依照需求規格書，設計應用系統的軟體架構、關聯式資料庫表格、以及資料查詢功能。



## 系統架構設計

- Model (資料模型層)

### MODEL

#### DeliveryMethodModel

+ constructor()  
+ getAll(): Promise<DeliveryMethodDto[]>

#### MemberModel

+ constructor()  
+ getAuth(member: MemberDto): Promise<MemberDto[] | null>  
+ getAll(): Promise<MemberDto[]>  
+ get(id: number): Promise<MemberDto[]>  
+ insert(member: MemberDto): Promise<number>  
+ update(member: MemberDto): Promise<boolean>

#### OrderModel

+ constructor()  
+ getAll(): Promise<OrderDto[]>  
+ get(memberId: number): Promise<OrderDto[]>  
+ insert(order: OrderDto): Promise<number>  
+ update(order: OrderDto): Promise<boolean>

#### TransactionTypeModel

+ constructor()  
+ getAll(): Promise<TransactionTypeDto[]>

#### ProductModel

+ constructor()  
+ getAll(): Promise<ProductDto[]>  
+ insert(product: ProductDto): Promise<number>  
+ update(product: ProductDto): Promise<boolean>  
+ delete(id: number): Promise<boolean>

#### OrderProductModel

+ constructor()  
+ getAll(): Promise<OrderProductDto[]>  
+ get(memberId: number): Promise<OrderProductDto[]>  
+ insert(data: OrderProductDto): Promise<number>

#### DeliveryMethodDto

+ id: number  
+ name: string  
  
+ constructor(payload: unknown)

#### MemberDto

+ id: number  
+ name: string  
+ phone: string  
+ account: string  
+ password: string  
+ authority: number  
  
+ constructor(payload: unknown)

#### OrderDto

+ id: number  
+ memberId: number  
+ address: string  
+ status: number  
+ deliveryId: number  
+ transactionId: number  
  
+ constructor(payload: unknown)

#### ProductDto

+ id: number  
+ name: string  
+ price: number  
+ quantity: number  
  
+ constructor(payload: unknown)

#### OrderProductDto

+ orderId: number  
+ productId: number  
+ quantity: number  
+ eachProductPrice: number  
+ total: number  
  
+ constructor(payload: unknown)

#### TransactionTypeDto

+ id: number  
+ name: string  
  
+ constructor(payload: unknown)

- View (視圖層)

## View

### LoginModule

```
+router.RouterService()
+member.MemberService()
+storage.StorageService()

+constructor(container)
+submitHandler(container)
+registerHandler(container)
+isLoginSuccess(account, pxwd)
```

### MemberInfoModule

```
+member.MemberService()

+constructor(container)
+renderUserInfo(container)
+submitHandler(container)
```

### MainModule

```
+products:array
+carts:array
+product.ProductService()
+order.OrderService()
+deliveryMethod.DeliveryMethodService()
+transactionTypes.TransactionTypeService()
+orderInfo:object

+constructor(container)
+submitHandler(container)
+renderProducts(container)
+renderProductItem(product)
+renderCarts(container)
+renderCartItem(cartItem)
+renderTransaction(container)
+renderDelivery(container)
+getProduct(pld)
```

### OrderModule

```
+oService.OrderService()
+opService.OrderProductService()
+pService.ProductService()
+tService.TransactionTypeService()
+dService.DeliveryMethodService()
+products:array
+orders:array
+orderProducts:array
+transactions:array
+deliverys:array

+constructor(container)
+render(container)
+renderOrderItem(order, filterOrderProducts)
+renderProducts(orderProduct)
+dialogEvent(orderId, status)
+getStatus(status)
+getStatusBgStyle(status)
+getProduct(productId)
+getTransaction(tid)
+getDelivery(dId)
```

### ProductModule

```
+product.ProductService()
+products:array

+constructor(container)
+renderProducts(body)
+renderProduct(product)
+addEvent(container)
+updateEvent(container)
+deleteEvent(container)
+cancelEvent(container)
+getProductDOM(container, productId)
```

### RegisterModule

```
+router.RouterService()
+member.MemberService()

+constructor(container)
+returnHandler(container)
+submitHandler(container)
+checkValue(account, pxwd, name, phone)
```

| DeliveryMethodService                  |
|--|
| +constructor()<br>+getDeliveryMethod() |

| MemberService   |
|---|
| +constructor()<br>+getMembers()<br>+getMember(memberId)<br>+updateMember(payload)<br>+addMember(payload)<br>+checkAuth(payload) |

| OrderService   |
|--|
| +constructor()<br>+getOrders()<br>+getOrder(memberId)<br>+updateOrder(payload)<br>+addOrder(payload) |

| TransactionTypeService                  |
|---|
| +constructor()<br>+getTransactionType() |

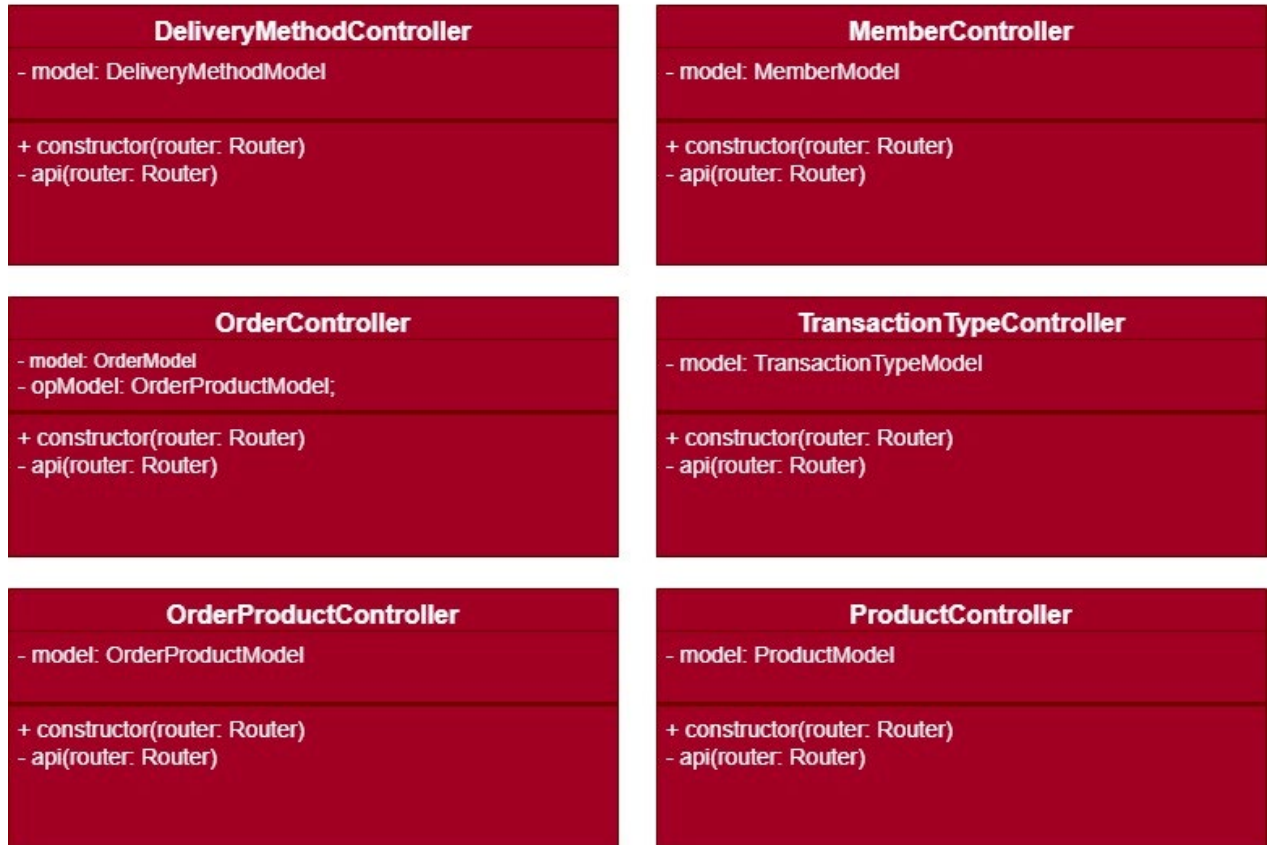
| OrderProductService  |
|--|
| +constructor()<br>+getOrderProducts()<br>+getOrderProduct(orderId) |

| ProductService   |
|--|
| +constructor()<br>+getProducts()<br>+updateProduct(payload)<br>+deleteProduct(productId)<br>+addProduct(payload) |

| BaseModule  |
|---|
| +header:headerElement<br>+router:RouterService()<br>+storage:StorageService()<br>+memberId:storage.getMemberId()<br>+auth:storage.getAuth() |
| +constructor(headerElement)<br>+checkIdentity()<br>+renderManagerHeader()<br>+renderUserHeader()<br>+headerRouterEvent()                    |

- Controller (控制層) & SQL (資料庫)

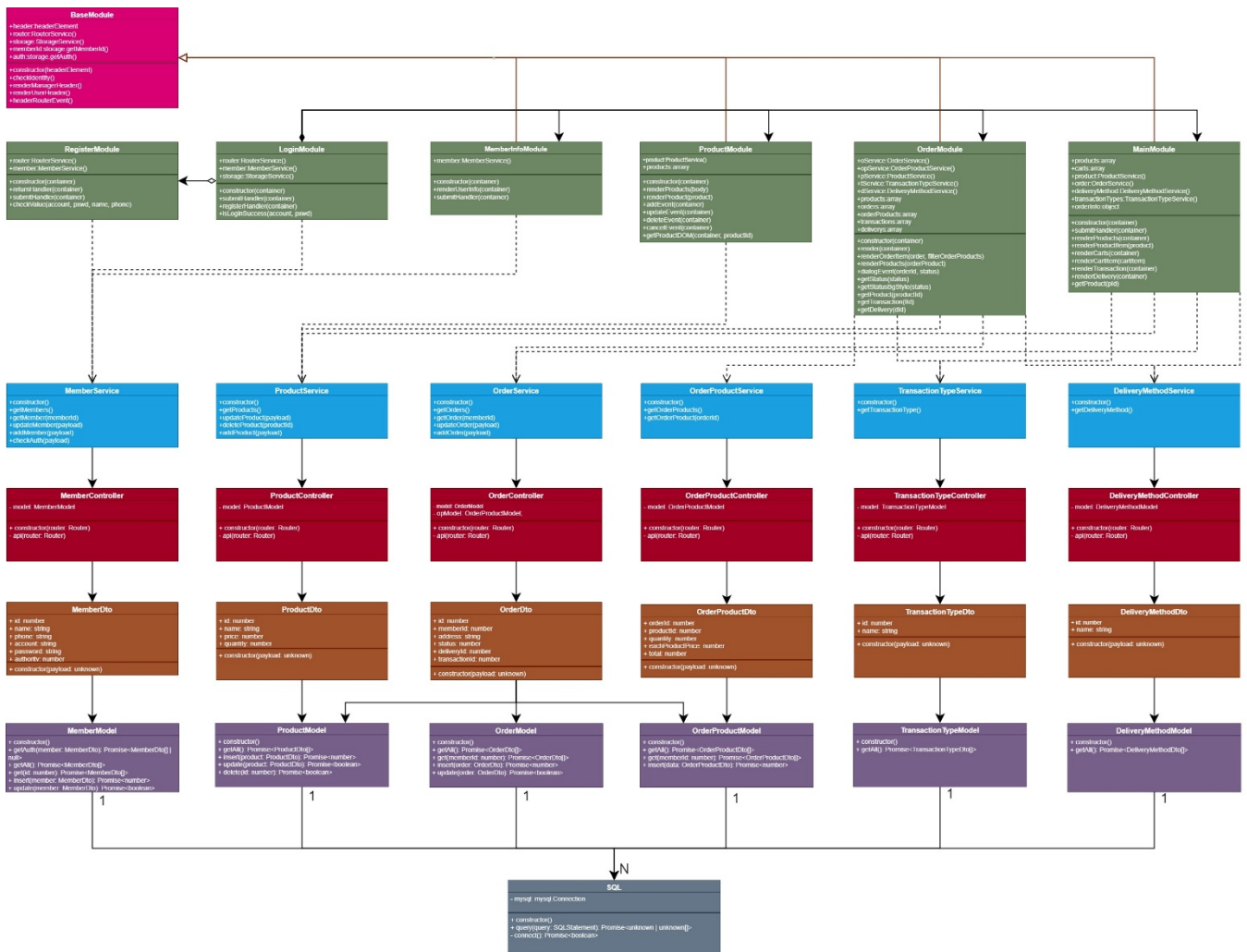
## CONTROLLER



## SQL

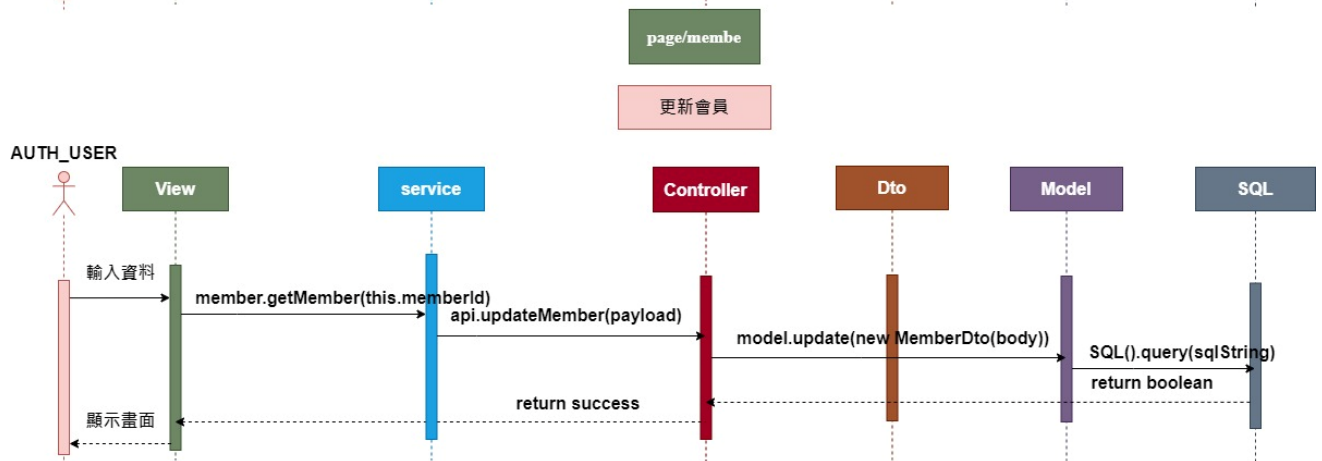
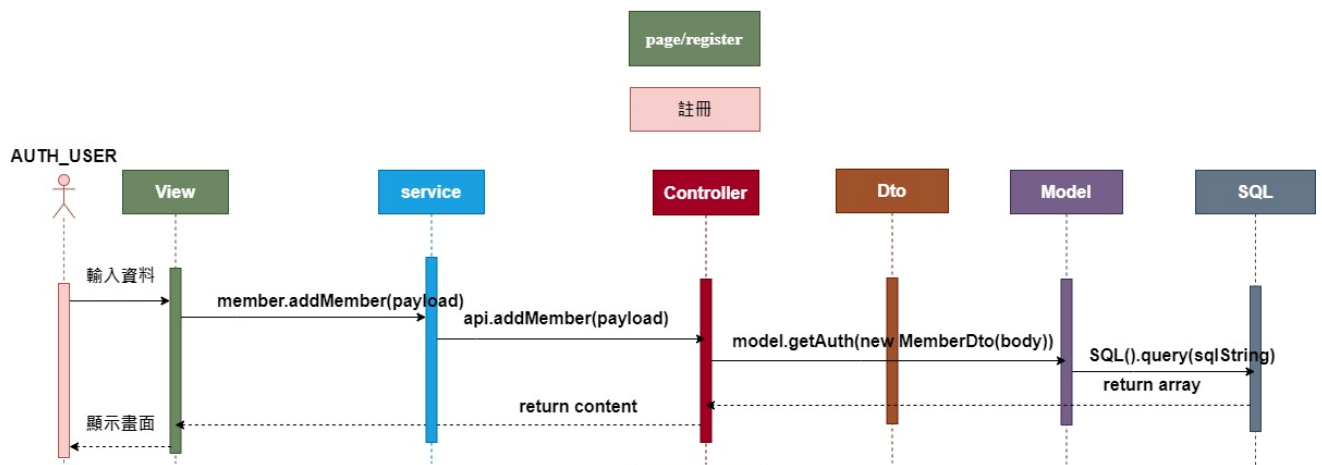
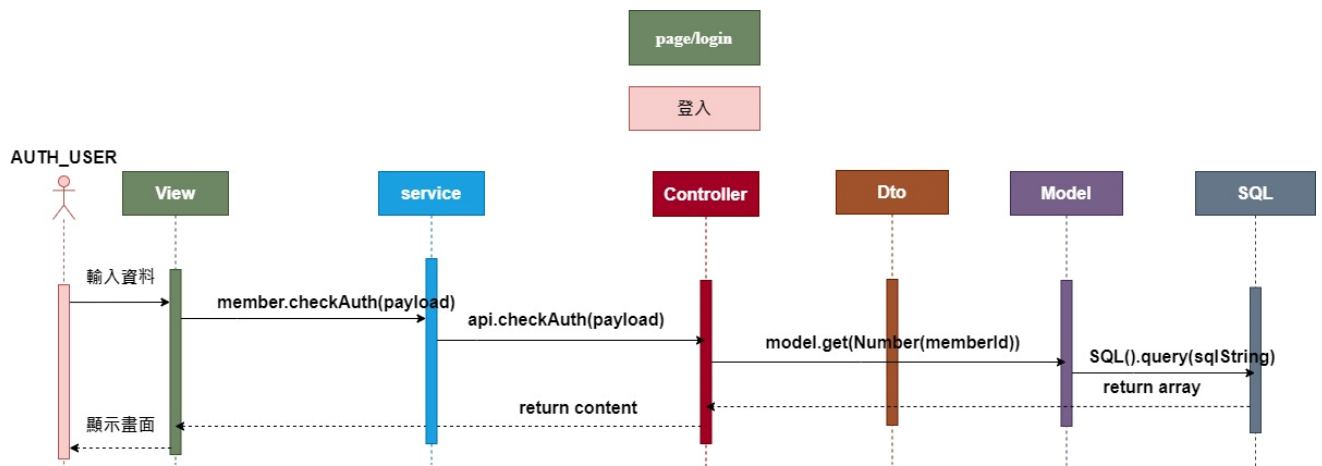


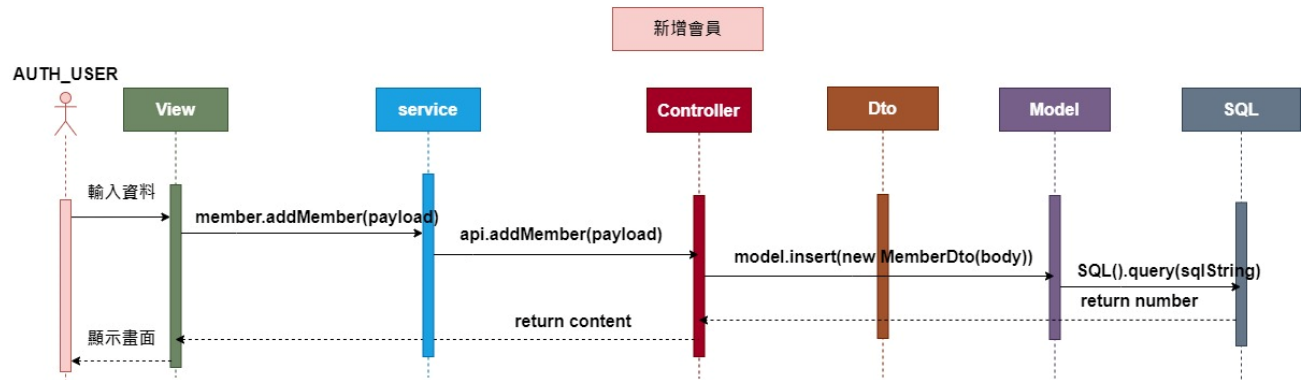
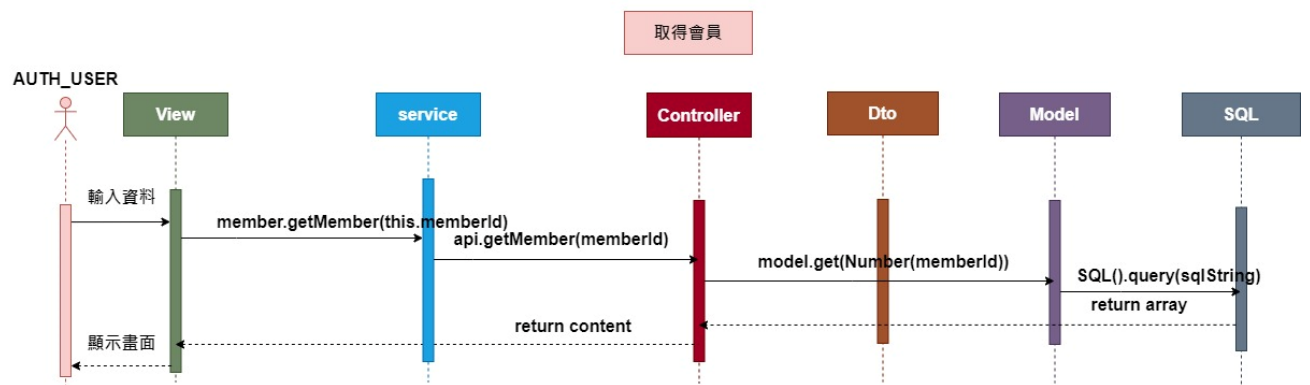
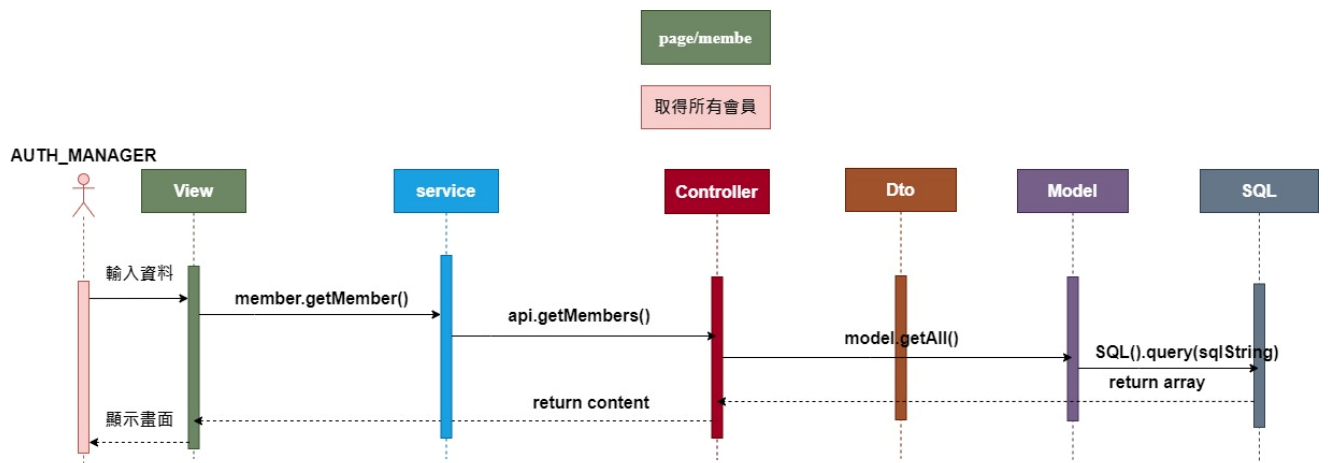
- 類別關係圖

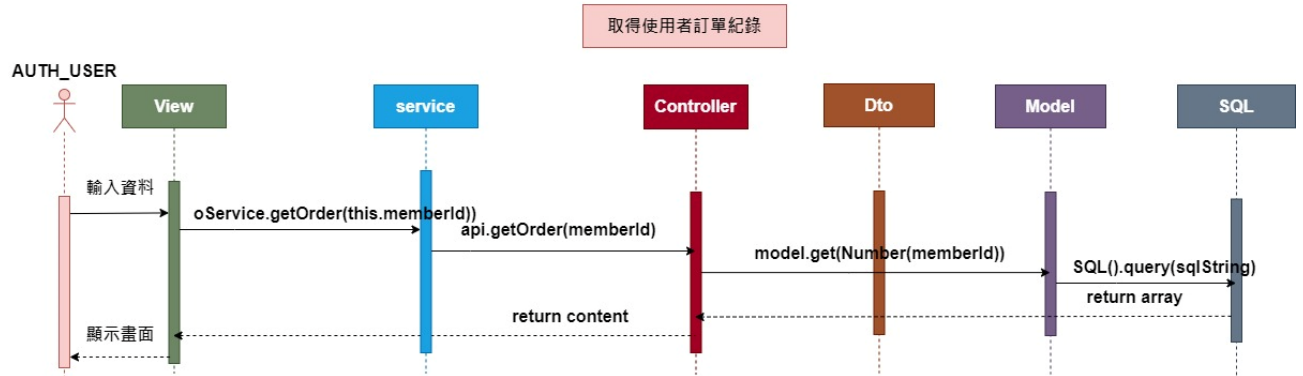
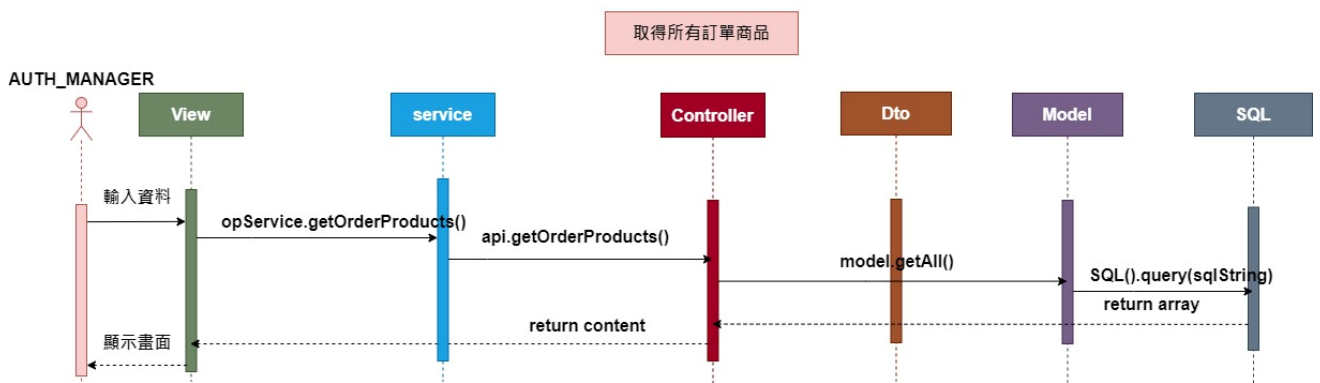
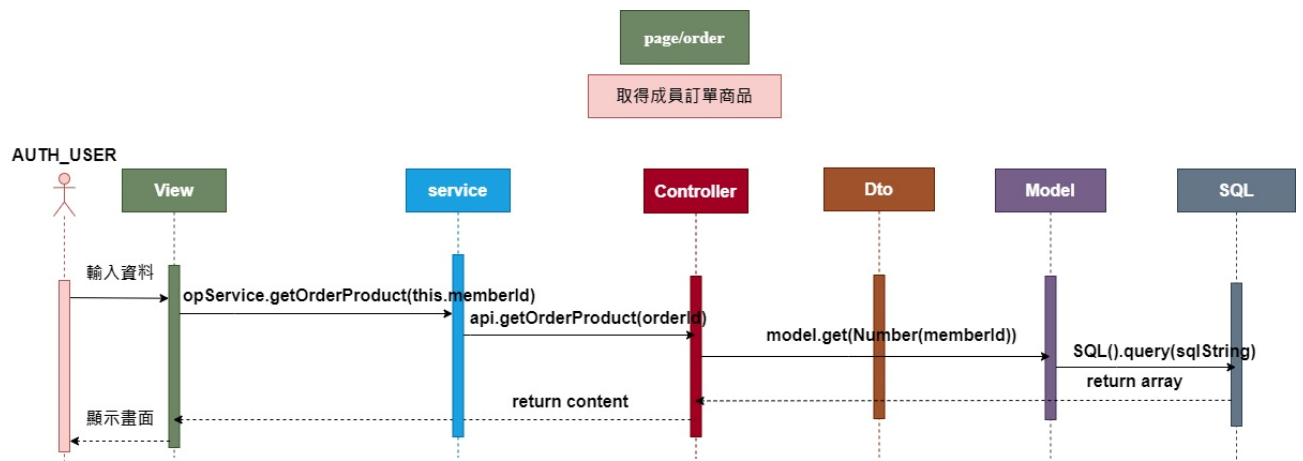




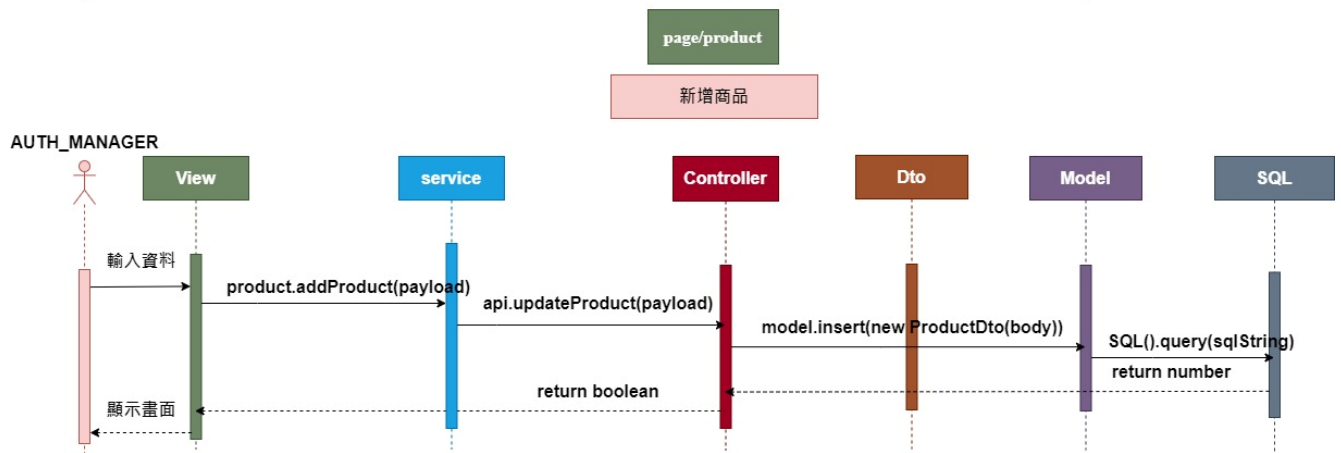
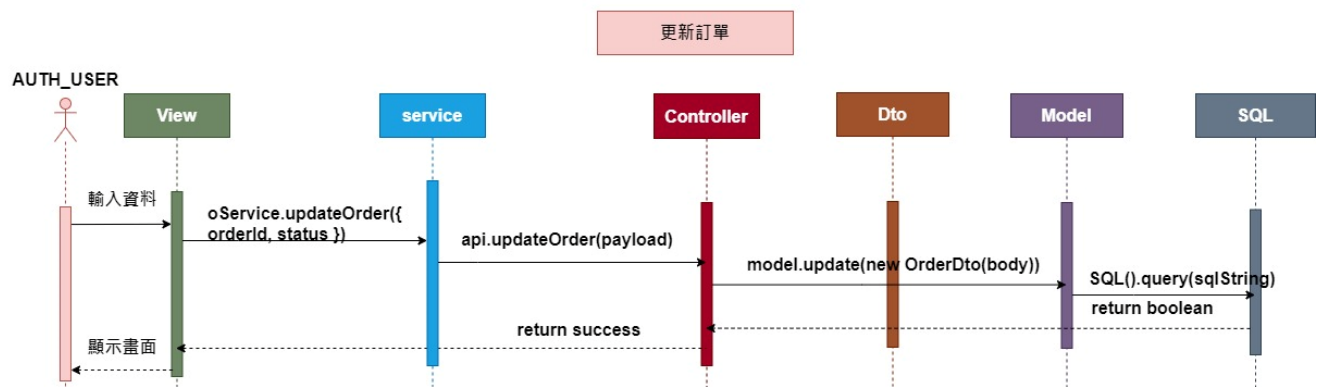
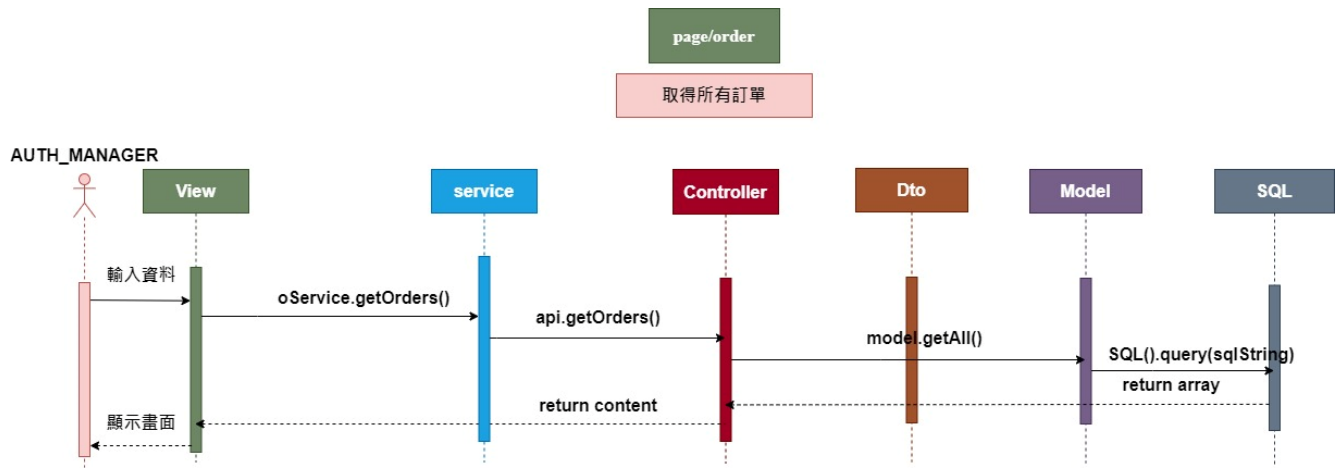
## 功能運作流程設計

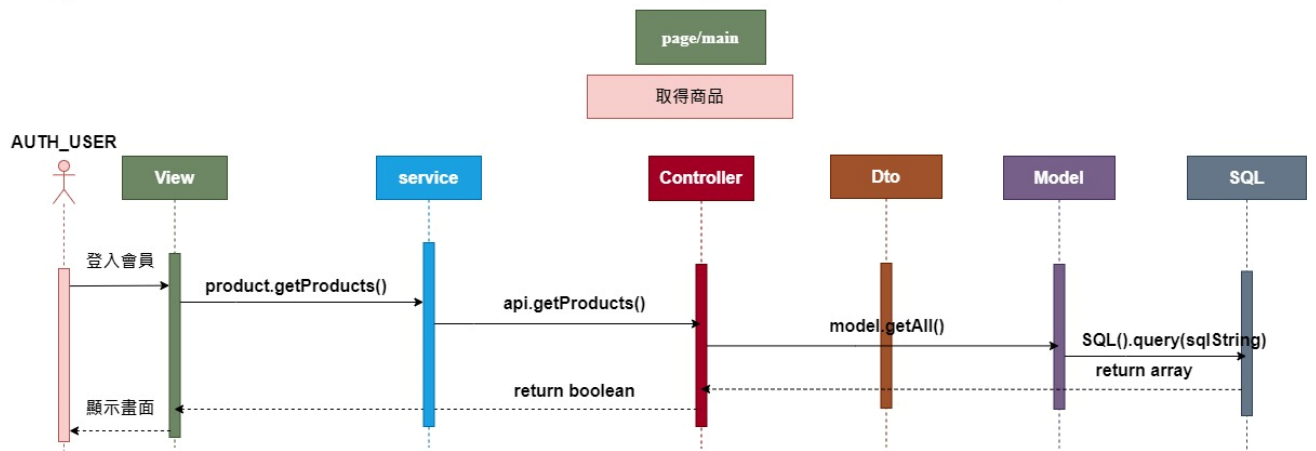
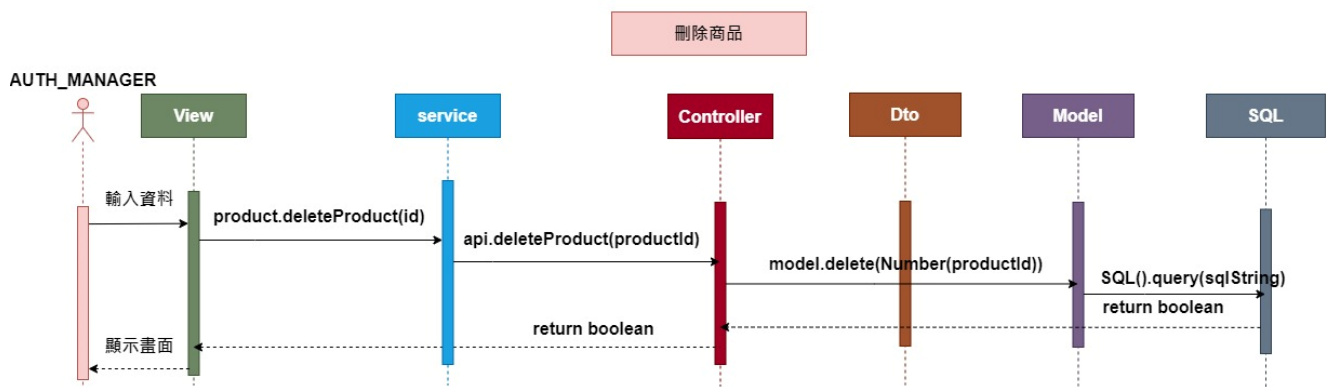
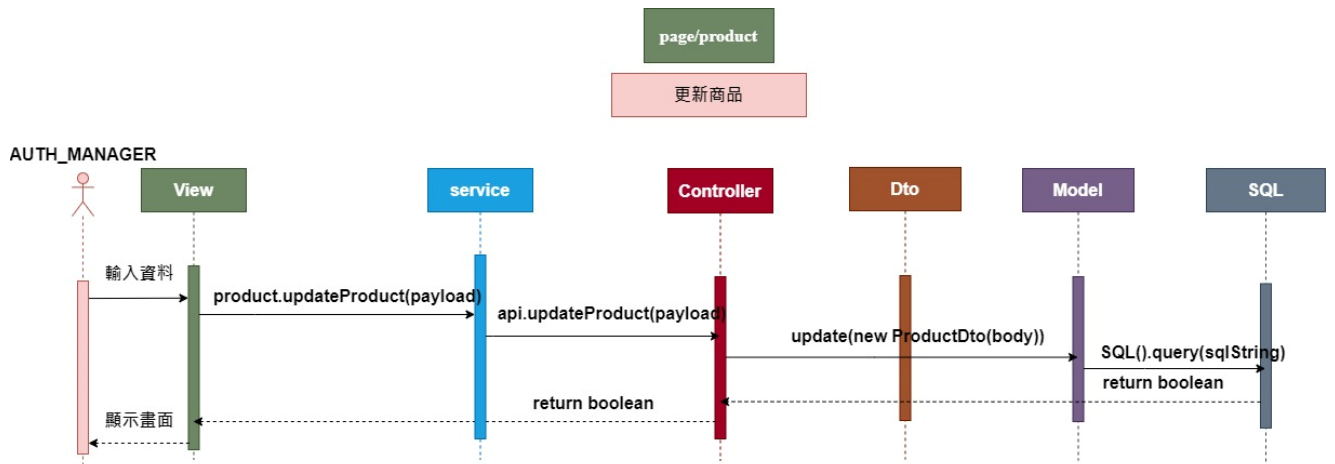


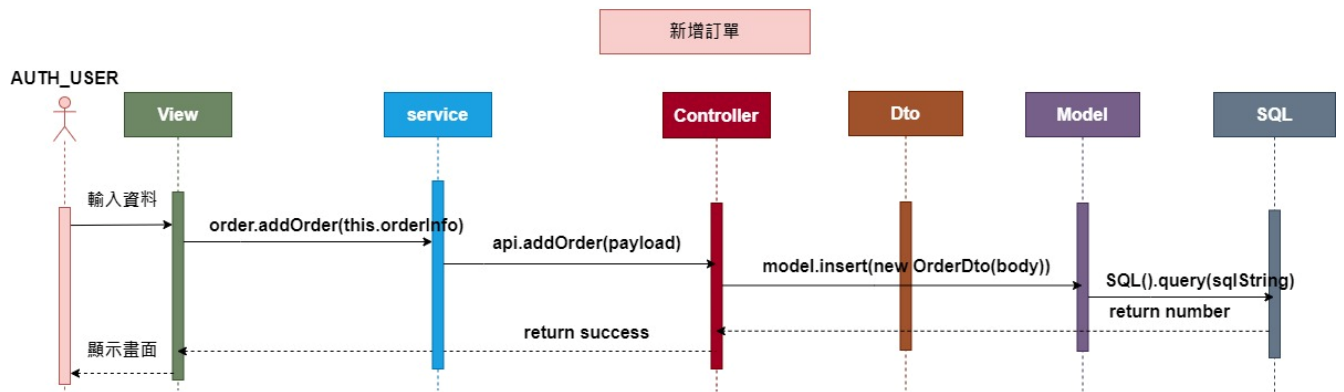
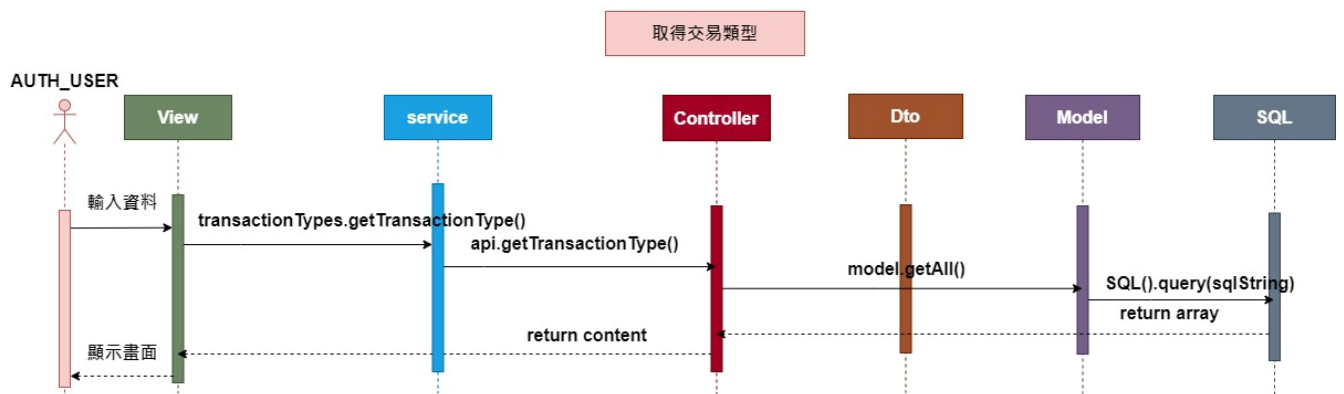
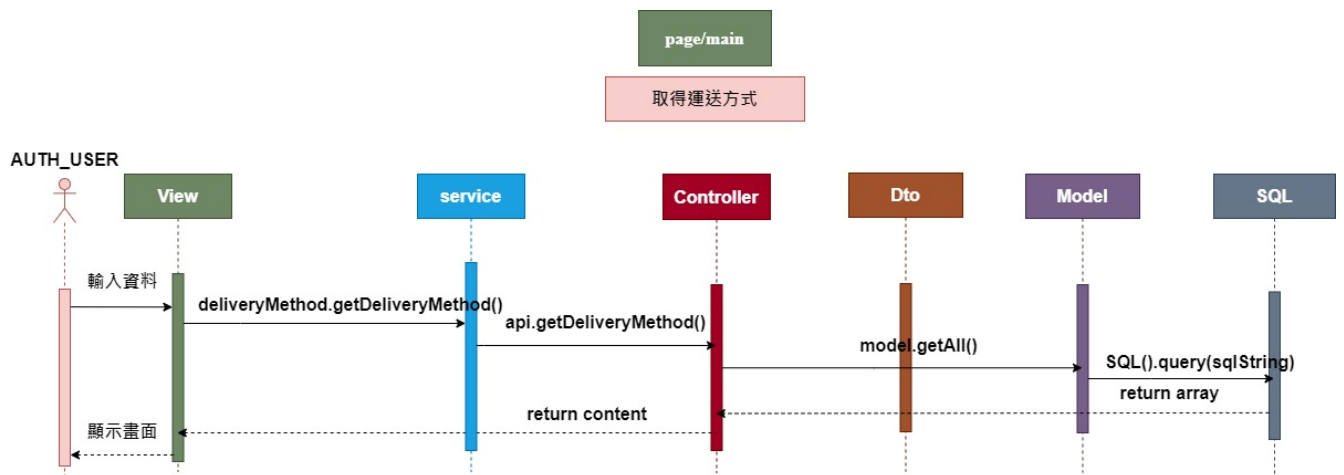






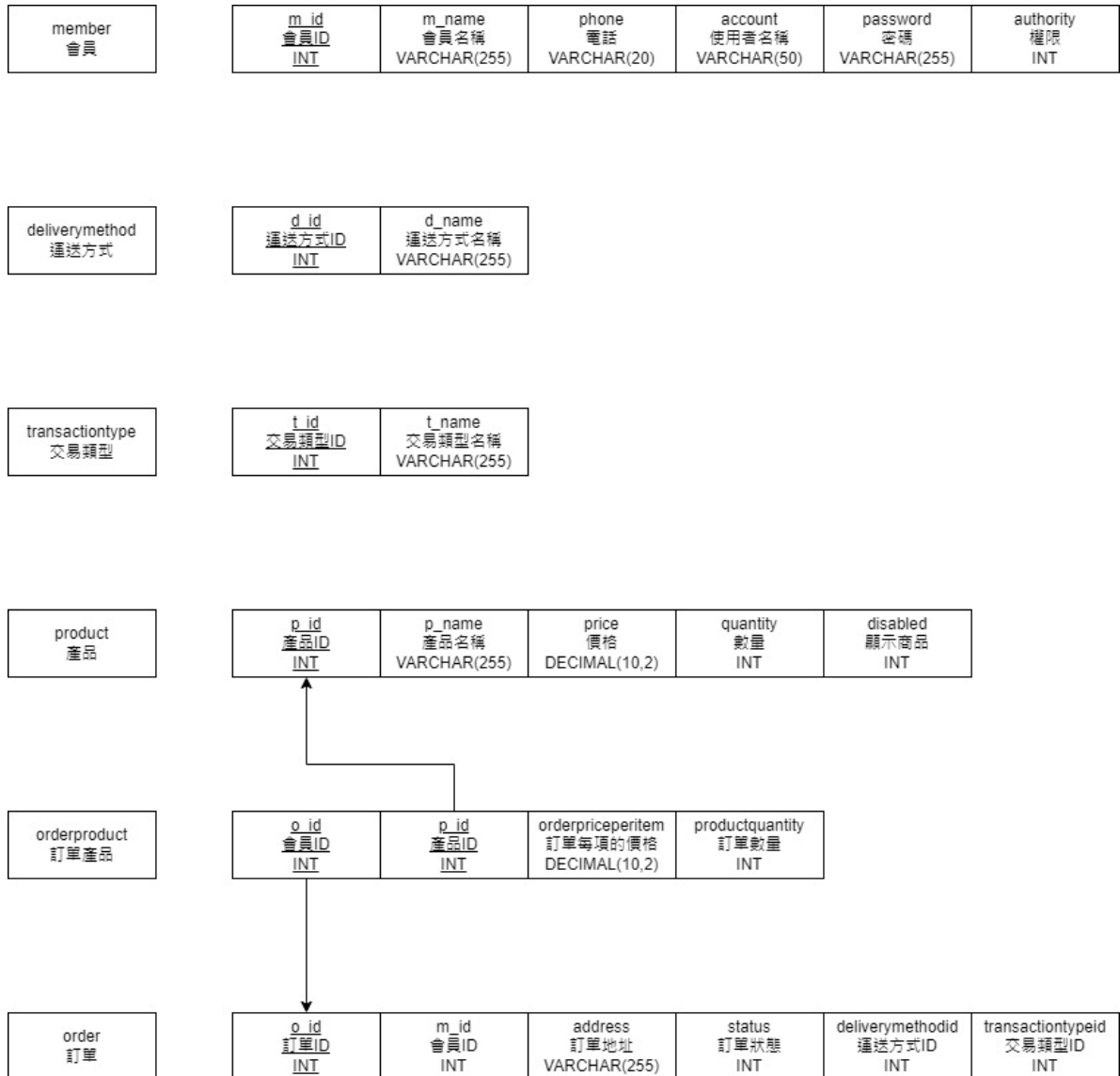






## 資料表設計

### • 綱目圖



● 欄位表

系統名稱:BUYGO

英文名稱:deliverymethod

保留期限:永久

中文名稱:運送方式

說明:記錄運送方式

| 序號 | 英文欄位   | 中文欄位    | 資料型態         | 額外描述 |
|----|--------|---------|--------------|------|
| 1  | d_id   | 運送方式 ID | int          |      |
| 2  | d_name | 運送方式名稱  | varchar(255) |      |

系統名稱:BUYGO

英文名稱:member

保留期限:永久

中文名稱:會員

說明:記錄會員資料

| 序號 | 英文欄位      | 中文欄位   | 資料型態         | 額外描述           |
|----|-----------|--------|--------------|----------------|
| 1  | m_id      | 會員 ID  | int          |                |
| 2  | m_name    | 會員名稱   | varchar(255) |                |
| 3  | phone     | 會員手機   | varchar(20)  |                |
| 4  | account   | 會員帳號   | varchar(50)  |                |
| 5  | password  | 會員密碼   | varchar(255) |                |
| 6  | authority | 會員帳號權限 | int          | 1:使用者<br>2:管理員 |

系統名稱:BUYGO

英文名稱:TransactionType

保留期限:永久

中文名稱:交易類型

說明:記錄交易類型

| 序號 | 英文欄位   | 中文欄位    | 資料型態         | 額外描述 |
|----|--------|---------|--------------|------|
| 1  | t_id   | 交易方式 ID | int          |      |
| 2  | t_name | 交易方式名稱  | varchar(255) |      |

系統名稱:BUYGO

英文名稱:Product

保留期限:永久

中文名稱:產品

說明:產品資訊

| 序號 | 英文欄位     | 中文欄位  | 資料型態          | 額外描述         |
|----|----------|-------|---------------|--------------|
| 1  | p_id     | 商品 ID | int           |              |
| 2  | p_name   | 商品名稱  | varchar(255)  |              |
| 3  | price    | 商品價格  | decimal(10,2) |              |
| 4  | quantity | 商品數量  | int           |              |
| 5  | disabled | 顯示商品  | int           | 0:開啟<br>1:關閉 |

系統名稱:BUYGO

英文名稱:order

保留期限:永久

中文名稱:訂單

說明:記錄訂單

| 序號 | 英文欄位              | 中文欄位    | 資料型態         | 額外描述   |
|----|-------------------|---------|--------------|--|
| 1  | o_id              | 訂單 ID   | int          |  |
| 2  | m_id              | 會員 ID   | int          |  |
| 3  | address           | 訂單地址    | varchar(255) |  |
| 4  | status            | 訂單狀態    | int          | 0:訂單處理中<br>1:運送中<br>2:已送達<br>3:已取貨(完成<br>訂單) |
| 5  | deliverymethodid  | 運送方式 ID | int          |  |
| 6  | transactiontypeid | 交易方式 ID | int          |  |

系統名稱:BUYGO

英文名稱:orderproduct

保留期限:永久

中文名稱:訂單產品

說明:記錄訂單產品

| 序號 | 英文欄位              | 中文欄位   | 資料型態          | 額外描述 |
|----|-------------------|--------|---------------|------|
| 1  | o_id              | 訂單 ID  | int           |      |
| 2  | p_id              | 商品 id  | int           |      |
| 3  | productquantity   | 訂單產品數量 | int           |      |
| 4  | orderpriceperitem | 訂單商品單價 | decimal(10,2) |      |

-- 1. 取得運送方式

```
SELECT * FROM new_schema.deliverymethod;
```

-- 2. 取得交易類型

```
SELECT * FROM new_schema.transactiontype;
```

-- 3. 身份驗證

```
SELECT * FROM new_schema.member WHERE account = ${member.account}
```

-- 4. 取得所有會員

```
SELECT * FROM new_schema.member;
```

-- 5. 取得會員

```
SELECT * FROM new_schema.member WHERE m_id = ${id}
```

-- 6. 新增會員

```
INSERT INTO new_schema.member (m_name, phone, account, password,  
    authority) VALUES (${member.name}, ${member.phone}, ${member.account},  
    ${member.password}, 1);
```

-- 7. 更新會員

```
UPDATE new_schema.member SET phone = ${member.phone}, password =  
    ${member.password}, m_name = ${member.name} WHERE account =  
    ${member.account};
```

-- 8. 取得所有訂單

```
SELECT * FROM new_schema.order;
```

-- 9. 取得會員訂單

```
SELECT * FROM new_schema.order WHERE m_id = ${memberId}
```

-- 10. 新增訂單

```
INSERT INTO new_schema.order (m_id, address, status, deliverymethodid,  
    transactiontypeid) VALUES (${order.memberId}, ${order.address}, 1 ,  
    ${order.deliveryId}, ${order.transactionId});
```

-- 11. 更新商品庫存

```
UPDATE new_schema.product SET quantity = ${product.quantity} WHERE p_id =  
    ${product.id};
```

-- 12. 更新訂單

```
UPDATE new_schema.order SET status = ${order.status} WHERE o_id =  
    ${order.id};
```



-- 13. 取得所有訂單商品

```
SELECT * FROM new_schema.orderproduct
```

-- 14. 取得成員訂單商品

```
SELECT op.* FROM new_schema.orderproduct as op JOIN new_schema.order as o
  WHERE op.o_id = o.o_id AND o.m_id = ${memberId}
```

-- 15. 取得商品

```
SELECT * FROM product WHERE disabled <> 1
```

-- 16. 新增商品

```
INSERT INTO product (p_name, price, quantity, disabled) VALUES
  (${product.name}, ${product.price}, ${product.quantity} , 0);
```

-- 17. 更新商品

```
UPDATE new_schema.product SET p_name = ${product.name}, quantity =
  ${product.quantity}, price = ${product.price} WHERE p_id =
  ${product.id};
```

-- 18. 刪除商品

```
UPDATE new_schema.product SET disabled = 1 WHERE p_id = ${id};
```