**作業說明：**  
 製作購物網站的架設是一個非常具有實際應用價值的專案，這不僅能讓我們學習技術上的挑戰，還能深入了解網站開發的各個層面，包括前端設計、後端開發、資料庫管理、安全性考量等等。以下是如何透過這個案例來達成學習目標和預期效益的一些說明：

### **學習目標**

1. **技術技能的提升**：
   * **前端開發**：學習使用HTML、CSS和JavaScript來設計和實現用戶界面，包括頁面的布局、樣式設計和互動效果。
   * **後端開發**：學習使用適當的後端語言（如Python、Node.js等）來處理用戶請求、業務邏輯和數據庫操作。
   * **數據庫管理**：學習如何設計和使用關聯型數據庫（如MySQL、PostgreSQL）來存儲和管理商品信息、用戶數據等。
   * **安全性考量**：理解常見的網站安全漏洞（如跨站腳本攻擊、SQL注入），並學習如何預防和處理這些安全問題。
2. **團隊協作技能的培養**：
   * 學習如何與他人合作，共同設計和開發一個大型項目。這包括溝通、任務分配、代碼版本控制（如Git）、問題解決等技能的提升。
3. **實戰經驗的積累**：
   * 透過實際的開發項目，學習如何應對真實世界中的技術挑戰和問題，如性能優化、用戶體驗改進等。

### **預期效益**

1. **技能全面提升**：
   * 通過實際項目的實作，可以徹底理解網站開發的全貌，從前端到後端再到數據庫的整合和應用，提升技術深度和廣度。
2. **團隊協作和溝通能力**：
   * 學習如何有效地與團隊成員合作，理解協作中的角色分工和互補，提高溝通效率和團隊合作能力。
3. **實戰經驗的增加**：
   * 購物網站的開發專案，將為你的履歷增添有價值的實際工作經驗，這對於未來求職或升學都會有很大幫助。
4. **問題解決能力的培養**：
   * 在開發過程中遇到各種挑戰和問題，這些都是寶貴的學習機會，能夠培養解決問題的能力和創新思維。

透過這樣一個實際且有挑戰性的項目，你和你的同學們將能夠全面提升自己的技術能力和團隊合作能力，同時也能夠在學術學習之外獲得更多寶貴的實戰經驗和生活技能。

**背景資料：**

### **Python 的應用範疇：**

1. **後端開發**：
   * 使用 Python 的 Web 框架，Flask，來處理用戶請求、業務邏輯和數據庫操作。
   * Flask 則是一個輕量級的 Web 框架，更靈活和容易擴展，適合小型到中型的應用開發。
2. **數據庫操作**：
   * Python 的 SQLAlchemy 庫可以幫助我們進行交互，從而管理和存取商品信息、用戶數據等。
3. **前端開發的支援**：
   * Python 的後端可以與前端技術（如HTML、CSS、JavaScript）無縫集成，通過 RESTful API 或模板引擎（如Django的模板系統）來提供動態內容和數據。

### **問題的介紹及相關知識：**

在開發購物網站的過程中，我們遇到了以下一些的挑戰和問題：

* **性能優化**：如何設計高效的數據庫結構、緩存常用數據以減少查詢時間，以及使用適當的技術來提高網站的響應速度。
* **用戶體驗（UX）改進**：如何設計直觀友好的用戶界面，並確保購物流程順暢和用戶能夠輕鬆找到他們想要的商品。
* **安全性**：如何預防和處理常見的網站安全漏洞，如跨站腳本攻擊（XSS）、SQL 注入攻擊以及應用適當的安全性措施來保護用戶數據和支付信息。
* **支付和訂單處理**：如何整合第三方支付服務（如PayPal、Stripe）並確保訂單處理的準確性和安全性。

在解決這些問題的過程中，我們三位將會學到許多實用的技能和知識，包括但不限於 Python 的高級功能、Web 開發最佳實踐、安全性措施和數據庫管理技術。這些技能不僅適用於購物網站的開發，還可以應用於各種其他類型的 Web 開發項目中。

**環境介紹：**

**作業系統 :** Windows 10、macOS

**程式碼編輯器 :** Visual Studio Code

**瀏覽器 :** Google Chrome

**實作方法：**

**環境檢查**：

* 確保所有團隊成員的開發環境都是一致的，包括操作系統（如Windows、macOS、Linux）、Python 版本（建議使用最新的穩定版本）、相應的開發工具和依賴庫。

**安裝與設定**：

* 安裝 Python 和相應的開發工具（如pip 包管理器、虛擬環境工具如virtualenv或者conda）。
* 選擇合適的集成開發環境（IDE），如PyCharm、VS Code等，或者使用文本編輯器如Sublime Text或Atom。

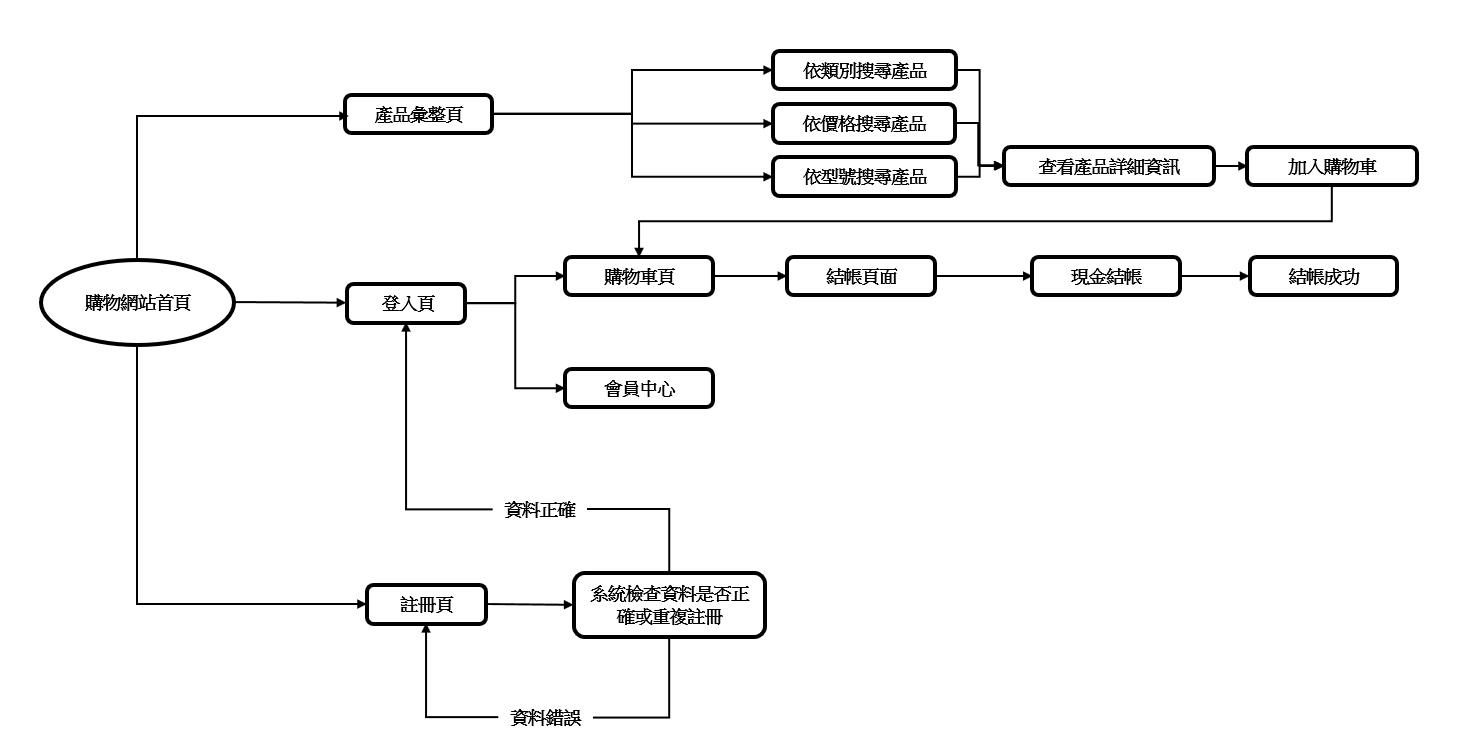
### **解決方案步驟**：

1. **設計數據庫結構**：
   * 使用適當的工具（如MySQL Workbench、pgAdmin等）設計購物網站所需的數據庫結構，包括商品信息、用戶信息、訂單信息等表格和關係。
2. **後端開發**：
   * 選擇合適的框架（如Django或Flask），並開始實作後端邏輯，包括路由設置、視圖函數、模型定義和數據庫操作。
   * 實現用戶註冊、登錄、商品列表展示、購物車管理、訂單生成等核心功能。
3. **前端開發**：
   * 設計並實作用戶界面，包括主頁面、商品列表頁、商品詳細頁、購物車頁面、訂單頁面等。
   * 使用HTML、CSS和JavaScript實現頁面結構、樣式和互動效果。
4. **集成與測試**：
   * 將前端與後端集成，通過RESTful API或模板引擎實現數據的動態顯示和交互。
   * 進行單元測試和集成測試，確保各個模塊和功能的正常運行。

**解決方案中所運用之技巧**：

* **RESTful API 設計**：用於前後端的數據交換和通信。
* **前端框架**：如React、Vue.js等，用於更高效的前端開發和組件化管理。
* **安全性實踐**：包括用戶身份驗證、輸入數據驗證、防止跨站腳本攻擊等。
* **性能優化**：使用緩存、異步處理等技術來提升網站的性能和響應速度。

**流程圖**：



**實作心得：**

(期末作業的想法、心得、困難、分組的感想…等。)

**林和泰:**

**在這次期末作業中，我負責用Python來編寫後端，並連接前端HTML以及SQL資料庫。這是一個非常有趣且具有挑戰性的任務。我的主要目標是創建一個功能完善的應用程序，使用戶能夠通過前端與後端進行互動，並能夠在資料庫中儲存和檢索數據。**

### **心得**

**通過這次作業，我學到了許多關於全棧開發的知識。使用Flask框架來編寫後端代碼，並與前端的HTML頁面進行互動，使我對於HTTP請求和響應有了更深入的理解。此外，與SQL資料庫的整合也讓我學會了如何有效地進行數據的存儲和查詢。**

### **困難**

**在這個過程中，我遇到了一些困難。首先是Flask框架的學習曲線，儘管它比其他一些框架更為輕量級和易於使用，但仍然需要一些時間來掌握。其次，處理數據庫的連接和查詢也遇到了一些挑戰，特別是在處理複雜查詢時需要仔細考慮效能問題。最後，確保前後端之間的數據傳輸安全和有效也是一個重要的挑戰。**

### **分組的感想**

**在這次作業中，我們進行了分組合作。這次的分組經驗對我來說非常寶貴。通過與組員的合作，我學到了如何更好地進行溝通和協作。我們分工明確，各自負責不同的部分，並通過定期的會議來交流進展和解決問題。這樣的分工合作不僅提高了工作的效率，還增強了團隊的凝聚力。**

**陳柏穎:  
這次期末作業中，我們需要利用SQLite來建立一個簡單的資料庫系統，包含會員（members）、商品（products）和訂單（orders）三個資料表。透過這個作業，我們可以學習和掌握SQL語法以及資料庫設計的基本概念，這對於日後的系統開發和資料管理將非常有幫助。**

### **心得**

**通過這次的期末作業，我學到了如何使用DB Browser for SQLite來建立和管理資料庫。以下是我的一些心得：**

1. **實作過程中學到的技能：**
   * **如何建立SQLite資料庫和資料表。**
   * **使用SQL語法進行資料的插入、查詢、更新和刪除。**
   * **資料庫設計的基本原則，如何設計資料表結構以滿足系統需求。**
2. **對資料庫設計的重要性有了更深的理解：**
   * **資料表之間的關聯如何設計，確保數據的一致性和完整性。**
   * **使用外鍵（foreign key）來建立資料表之間的聯繫。**

### **困難**

**在作業的實作過程中，我遇到了一些挑戰和困難：**

1. **SQL語法的熟悉度：**
   * **起初對於SQL語法不夠熟悉，特別是複雜的查詢和聯合查詢（JOIN）的使用。透過多次的練習和查閱資料，逐漸掌握了這些語法。**
2. **資料表設計：**
   * **設計資料表結構時，需要仔細考慮各個欄位的數據類型及其關聯性。這需要一定的經驗和設計思維，剛開始會有點困難，但在不斷調整和優化後，最終設計出了較為合理的資料表結構。**

### **分組的感想**

**在這次作業中，我們以小組的形式進行討論和實作。以下是我的分組感想：**

1. **分工合作：**
   * **我們小組每個成員分工明確，各自負責不同的部分。例如，有人負責資料庫的建立，有人負責SQL語法的編寫，有人負責測試和調試。這種分工合作的方式提高了效率，也讓我們學會了如何在團隊中協作。**
2. **討論和交流：**
   * **在討論和交流中，大家互相分享自己的想法和解決問題的方法，這讓我們學到了更多不同的解決方案和思路。同時，通過彼此的指導和幫助，也增進了小組成員之間的默契和友誼。**

**總結來說，這次的期末作業讓我受益匪淺，不僅提升了自己的技術水平，還加深了對資料庫設計和管理的理解。更重要的是，通過分組合作，我學到了如何有效地進行團隊合作，這對於未來的工作和學習都是非常寶貴的經驗。**

**林政漢:**

1. **學習收穫：**
   * **透過這個作業，我不僅學會了如何設計和實現不同類型的HTML頁面，還了解了前端頁面如何與後端服務集成以實現動態內容的展示。**
   * **在實作過程中，可能學到了許多CSS樣式設計技巧、JavaScript互動效果的實現，以及如何使用模板引擎或JavaScript框架來管理和顯示動態數據。**
2. **團隊合作：**
   * **分組開發讓我體驗了團隊協作的重要性。在與同學合作的過程中，可能遇到了溝通、分工、協調進度等方面的挑戰，這些都是現實工作中必須克服的問題。**
3. **技術挑戰和解決困難：**
   * **可能在開發過程中遇到了各種技術挑戰，比如兼容性問題、頁面設計的美觀性和易用性、跨瀏覽器支持等。**
   * **解決這些問題需要不斷學習和查找解決方案的能力，並且與團隊成員共同協作找出最佳的實現方式。**

### **分組的感想**

* **分組合作讓我體驗了團隊協作的樂趣和挑戰。通過分工合作，每個人可以專注於自己擅長的部分，同時又能共同解決整個項目面臨的難題。**
* **分組還有助於相互學習和知識交流，我的組員分別有不同的技術背景和經驗，這樣可以彼此補充和提升。**

### **建議和未來展望**

* **當前我已經完成了大部分HTML頁面的設計和實現，接下來可以專注於優化和完善這些頁面，確保它們在不同設備和瀏覽器上的表現良好。**
* **如果還有時間，可以深入學習和掌握一些JavaScript框架（如React、Vue.js），這將有助於提升我的前端開發效率和能力。**

**附註：**

**工作分配**

**9B117028林政漢（HTML)**

index.html - 首頁 (完成)

exhibit.html - 首頁隨機商品 (完成)

login.html - 登入 (完成)

register.html - 註冊 (完成)

member.html - 會員中心 (完成)

member\_info.html - 顯示會員資料 (完成)

edit.html - 變更會員資料 (完成)

member\_info.html -會員資料-訂單查看 (完成)

cart.html - 購物車 (完成)

product - 商品頁 (完成)

product\_details - 各商品介紹頁(動態網頁) (完成)

administrator - 管理員介面(查看訂單、新增刪除商品)

all\_member.html所有會員資料

all\_order.html商品管理

all\_product.html訂單查詢

實作方法

設計前端各項 HTML

尋找適合的模板加以更改

**9B117033林和泰(Python)**

**整合html以及連接sql資料庫**

**以及主程式的撰寫**

**實作方法**

設置 Flask 後端

**撰寫後端邏輯，**根據不同的路由來處理請求，從資料庫中獲取數據並返回給前端 (HTML)。

### **添加一些測試資料**

在 app.py 文件中，添加一些測試數據

最後運行應用程式 如沒達到原本想要的畫面再逐步進行修改

**9B117031陳柏穎**

資料庫檔mydb.db

再建立sqlite3資料庫

會員(members)跟 商品(products)訂單 (orders)資料表欄位

INSERT INTO member

INSERT INTO products

INSERT INTO orders

作業說明

資料庫實作方法

1.安裝DB Browser for SQLite

2.尋找商品

3.手動建立資料庫、資料表並新增記錄

3.1以【DB Browser for SQLite】在專案目錄下建立【computer store.db】

3.2新增資料到商品.會員資料表

負責尋找商品以及建立資料庫跟資料表

怎麼建立資料庫跟資料表是參考老師第三次小考跟期中作業

看需要什麼樣的資料表以及需要哪些欄位

SQL語法也是參考老師的 慢慢把商品逐項轉成SQL語法

**參考文獻**

**商品圖片以及資料是出自順發賣場**

全文完