

Software Design: Homework [Template]

Homework Number: #1

Homework Start Date: 20240405

Homework Due Date: 20240407

Homework Submitted Date: 20240407

Student: 軟體系大三陳昶亨

Student ID: 411077013

Answer:

1. 客人選擇商品，甜度冰塊及客製化選項
2. 系統顯示商品價格以及詢問客人飲料甜度冰塊
3. 系統顯示其他客製化選項
4. 系統計算客人的商品總額
5. 系統顯示付款選項
6. 客人選擇付款方式並付款
7. 系統列出客人的商品並印出廚吧單
8. 系統以商品數量計算成本及利潤製成日報表和月報表

Hw1---使用案例圖

Methods	My Outcomes	
draw.io	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1hMirzIE9AQT2kAaABqyCXsgcxGSgSju1?usp=sharing
	Type	使用案例圖
	Diagram	<pre> graph LR Customer((客人)) --> UC1(選擇商品及數量) UC1 --> UC2(選擇甜度冰塊) UC2 --> UC3(填寫客製化選項) UC3 --> UC4(付款) System((系統)) --> UC5(顯示價格和飲料選項) UC5 --> UC6(顯示其他客製化選項) UC6 --> UC7(計算價錢並收款) System --> UC8(列印廚吧單) UC8 --> UC9(計算成本和利潤製作報表) </pre>
	MyLearning	1. 依照修改過後的 plantuml 畫一次就很快了，而且可以依照美觀進行調整

PlantUML	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1hMirzIE9AQT2kAaABqyCXGcxGSgSju1?usp=sharing
	Type	使用案例圖
	Code	<pre> @startuml left to right direction actor 客人 as Customer actor 系統 as System rectangle "購物流程" { Customer --> (選擇商品及數量) (選擇商品及數量) --> (選擇甜度冰塊) (選擇甜度冰塊) --> (填寫客製化選項) (填寫客製化選項) --> (付款) System --> (顯示價格和飲料選項) (顯示價格和飲料選項) --> (顯示其他客製化選項) (顯示其他客製化選項) --> (計算商品總額並收款) (計算商品總額並收款) --> (列印出廚吧單) (列印出廚吧單) --> (計算成本和利潤製作報表) } @enduml </pre>

	Diagram	
	MyLearning 1. 先用 chatgpt 生成一次，在自己針對錯誤和缺少的地方修改會比較快且較正確	
ChatGPT	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1hMirzIE9AQT2kAaABqyCXSgcxGSgSju1?usp=sharing
	Type	使用案例圖
	ChatGPT Version	V3.5
	ChatGPT Prompt	請以以下需求，用 plantuml 的格式，輸出使用案例圖
	ChatGPT Code	<pre>@startuml left to right direction actor 客人 as Customer actor 系統 as System rectangle "購物流程" { Customer->>選擇商品及數量 選擇商品及數量->>選擇甜度冰塊 選擇甜度冰塊->>填寫客製化選項 填寫客製化選項->>付款 System->>顯示價格和飲料選項 顯示價格和飲料選項->>顯示其他客製化選項 顯示其他客製化選項->>計算商品總額並收款 計算商品總額並收款->>列印出廚吧單 列印出廚吧單->>計算成本和利潤製作報表 }</pre>

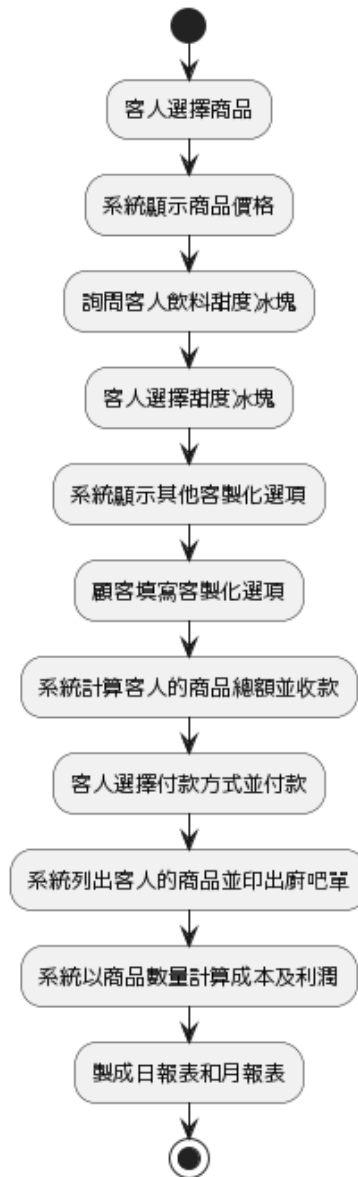
		<p>Customer --> (選擇商品)</p> <p>(選擇商品) --> (顯示價格和詢問飲料選項)</p> <p>(顯示價格和詢問飲料選項) --> (顯示其他客製化選項)</p> <p>(顯示其他客製化選項) --> (計算商品總額並收款)</p> <p>(計算商品總額並收款) --> (列出商品並印出廚吧單)</p> <p>(列出商品並印出廚吧單) --> (計算成本和利潤製作報表)</p> <p>}</p> <p>@enduml</p>
	Diagram	<pre> sequenceDiagram actor 客人 participant 系統 客人->>選擇商品 選擇商品->>顯示價格和詢問飲料選項 顯示價格和詢問飲料選項->>顯示其他客製化選項 顯示其他客製化選項->>計算商品總額並收款 計算商品總額並收款->>列出商品並印出廚吧單 </pre>
	MyLearning	<ol style="list-style-type: none"> 1. 善用 chatgpt 可以快速整理出圖表架構 2. Chatgpt 雖然好用，但是正確性不一定，需要再手動檢查修改

Hw2---活動圖

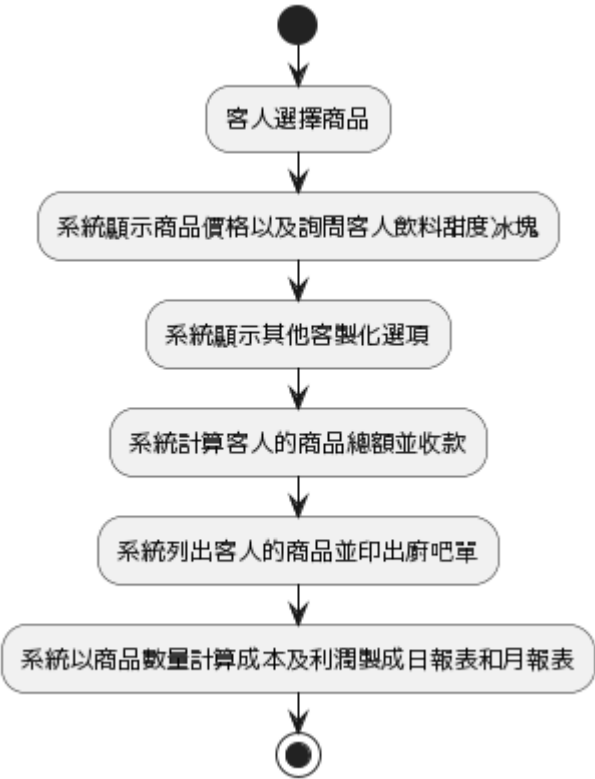
Methods	My Outcomes	
draw.io	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1jMVRO1w0d6tFrPjJ7vIUxChZ3ZKD79Nd?usp=sharing
	Type	活動圖
	Diagram	<pre> graph LR subgraph Customer [顧客] Start(()) --> Input[輸入選擇商品] Input --> SelectIce[選擇甜度冰塊及數量] SelectIce --> FillOptions[填寫客製化選項] FillOptions --> CalculateTotal[計算商品總額並收款] CalculateTotal --> ListMenu[列印廚吧單] ListMenu --> CalculateProfit[以商品數量計算成本及利潤] CalculateProfit --> MakeReport[製作日及月報表] MakeReport --> End((())) end subgraph System [飲料店電子商務] DisplayList[顯示商品選擇] SelectIceSys[選擇甜度冰塊及數量] FillOptionsSys[填寫客製化選項] CalculateTotalSys[計算商品總額並收款] ListMenuSys[列印廚吧單] CalculateProfitSys[以商品數量計算成本及利潤] MakeReportSys[製作日及月報表] end Input --> DisplayList DisplayList --> SelectIceSys SelectIceSys --> SelectIce SelectIce --> FillOptionsSys FillOptionsSys --> FillOptions FillOptions --> CalculateTotalSys CalculateTotalSys --> CalculateTotal CalculateTotal --> ListMenuSys ListMenuSys --> ListMenu ListMenu --> CalculateProfitSys CalculateProfitSys --> CalculateProfit CalculateProfit --> MakeReportSys MakeReportSys --> MakeReport </pre>
	MyLearning	<ol style="list-style-type: none"> 1. 依照修改過後的 plantuml 畫一次就很快了，而且可以依照美觀進行調整 2. Drawio 有泳道圖的功能，可以將顧客和系統的行為區分
PlantUML	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1jMVRO1w0d6tFrPjJ7vIUxChZ3ZKD79Nd?usp=sharing
	Type	活動圖
	Code	@startuml

		<p>start</p> <p>:客人選擇商品;</p> <p>:系統顯示商品價格;</p> <p>:詢問客人飲料甜度冰塊;</p> <p>:客人選擇甜度冰塊;</p> <p>:系統顯示其他客製化選項;</p> <p>:顧客填寫客製化選項;</p> <p>:系統計算客人的商品總額並收款;</p> <p>:客人選擇付款方式並付款;</p> <p>:系統列出客人的商品並印出廚吧單;</p> <p>:系統以商品數量計算成本及利潤;</p> <p>:製成日報表和月報表;</p> <p>stop</p> <p>@enduml</p>
--	--	--

Diagram



	MyLearning	<ol style="list-style-type: none"> 1. 先用 chatgpt 生成一次，在自己針對錯誤和缺少的地方修改會比較快且較正確 2. 在 plantuml 沒有找到泳道圖的相關語法，需要用 drawio 自行修改 3. 詳細的需求要自己新增
ChatGPT	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1jMVRO1w0d6tFrPjJ7vIUxChZ3ZKD79Nd?usp=sharing
	Type	活動圖
	ChatGPT Version	V3.5
	ChatGPT Prompt	請以以下需求，用 plantuml 的格式，輸出活動圖
	ChatGPT Code	<pre> @startuml start :客人選擇商品; :系統顯示商品價格以及詢問客人飲料甜度冰塊; :系統顯示其他客製化選項; :系統計算客人的商品總額並收款; :系統列出客人的商品並印出廚吧單; :系統以商品數量計算成本及利潤製成日報表和月報表; stop @enduml </pre>

	<p>Diagram</p>	 <pre> graph TD Start(()) --> A[客人選擇商品] A --> B[系統顯示商品價格以及詢問客人飲料甜度冰塊] B --> C[系統顯示其他客製化選項] C --> D[系統計算客人的商品總額並收款] D --> E[系統列出客人的商品並印出廚吧單] E --> F[系統以商品數量計算成本及利潤製成日報表和月報表] F --> End((())) </pre>	
	<p>MyLearning</p>	<ol style="list-style-type: none"> 1. 善用 chatgpt 可以快速整理出圖表架構 2. 活動圖的需求比較明確，圖輸出比較正確 	

Hw3---類別圖

Methods	My Outcomes	
draw.io	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1lfKbxnrnwb-f3tZXzJmkl5uEt5wxzE3Y?usp=sharing
	Type	類別圖
	Diagram	<pre> classDiagram class System { displayPrice(item: item): void displayCustomizations(item: item): void calculateTotal(customer: Customer): float } class Customer { name: string selectedItem: list<string> selectPayWay: string<PayWay> chooseItem(item: item): void chooseCustomization(chooseCustomization: chooseCustomization): void choosePayWay(PayWay: string): void } class Item { name: string price: float customization: list<customization> getPrice(): float } class Report { year, month, date: string sale: list<string, int> cost: list<string, int> profit: int generateDailyReport(): void generateMonthlyReport(): void } class Customization { name: string options: list<string> } System --> Customer : generate System --> Report : generate System --> Item : calculate System --> Customization : display Customer --> Item : select Customer --> Customization : select </pre> <p>The diagram illustrates the relationships between five classes: System, Customer, Item, Report, and Customization. The System class has methods for displaying prices, customizations, and calculating totals. It interacts with the Customer class (generate), the Report class (generate), the Item class (calculate), and the Customization class (display). The Customer class has attributes for name, selected items, and payment method, and methods for choosing items, customizations, and payment methods. It interacts with the Item class (select) and the Customization class (select). The Item class has attributes for name, price, and customizations, and a method to get the price. The Report class has attributes for date, sale, cost, and profit, and methods to generate daily and monthly reports. The Customization class has attributes for name and options.</p>

	MyLearning	1. 依照修改過後的 plantuml 畫一次就很快了，而且可以依照美觀進行調整 2. 格子很多不好拉
PlantUML	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1IfKbxnrnwb-f3tZXzJmkl5uEt5wxzE3Y?usp=sharing
	Type	類別圖
	Code	<pre> @startuml class Customer { - name: string - selectedItems: list<Item> - selectPayWay:string<PayWay> + chooseItem(item: Item): void + chooseCustomization(customization: Customization): void + choosePayWay(PayWay:string):void } class Item { - name: string - price: float - customizations: list<Customization> + getPrice(): float </pre>

```
}
```

```
class Customization {
```

```
    - name: string
```

```
    - options: list<string>
```

```
}
```

```
class System {
```

```
    + displayPrice(item: Item): void
```

```
    + displayCustomizations(item: Item): void
```

```
    + calculateTotal(customer: Customer): float
```

```
}
```

```
class Report {
```

```
    - year,month,date: string
```

```
    - sale: list<string,int>
```

```
    - cost:list<string,int>
```

```
    - profit:<int>
```

```
    + generateDailyReport(): void
```

```
    + generateMonthlyReport(): void
```

```
}
```

```
Customer -- Item : selects >
```

```
Customer -- Customization : selects >
```

		<p>System -- Report : generate ></p> <p>System -- Item : displays ></p> <p>System -- Customization : displays ></p> <p>System -- Customer : calculates, generates ></p> <p>@enduml</p>	
--	--	--	--

	<p>Diagram</p>	<pre> classDiagram class System { +displayPrice(item: Item): void +displayCustomizations(item: Item): void +calculateTotal(customer: Customer): float } class Customer { +name: string +selectedItems: list<Item> +selectPayWay: string<PayWay> +chooseItem(item: Item): void +chooseCustomization(customization: Customization): void +choosePayWay(PayWay: string): void } class Item { +name: string +price: float +customizations: list<Customization> +getPrice(): float } class Customization { +name: string +options: list<string> } class Report { +year,month,date: string +sale: list<string,int> +cost: list<string,int> +profit: <int> +generateDailyReport(): void +generateMonthlyReport(): void } System --> Customer : calculates, generates System --> Report : generate Customer --> Item : selects Customer --> Customization : selects Item --> System : displays Customization --> Report : displays </pre> <p>The diagram illustrates the relationships between five classes: System, Customer, Item, Customization, and Report. Each class has a set of attributes and methods. The relationships are as follows:</p> <ul style="list-style-type: none"> System (Methods: <code>displayPrice(item: Item): void</code>, <code>displayCustomizations(item: Item): void</code>, <code>calculateTotal(customer: Customer): float</code>) <ul style="list-style-type: none"> Calculates and generates data for Customer. Generates data for Report. Customer (Attributes: <code>name: string</code>, <code>selectedItems: list<Item></code>, <code>selectPayWay: string<PayWay></code>; Methods: <code>chooseItem(item: Item): void</code>, <code>chooseCustomization(customization: Customization): void</code>, <code>choosePayWay(PayWay: string): void</code>) <ul style="list-style-type: none"> Selects Item and Customization. Item (Attributes: <code>name: string</code>, <code>price: float</code>, <code>customizations: list<Customization></code>; Method: <code>getPrice(): float</code>) <ul style="list-style-type: none"> Displays data to System. Customization (Attributes: <code>name: string</code>, <code>options: list<string></code>) <ul style="list-style-type: none"> Displays data to Report. Report (Attributes: <code>year,month,date: string</code>, <code>sale: list<string,int></code>, <code>cost: list<string,int></code>, <code>profit: <int></code>; Methods: <code>generateDailyReport(): void</code>, <code>generateMonthlyReport(): void</code>)
	<p>MyLearning</p>	<ol style="list-style-type: none"> 1. 將報表另外列出來成一個類別 2. 按照年月日分別，內容有銷售、成本、利潤 3. 將 custom 內不太適合的東西刪除

ChatGPT	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1lfKbxnrnwb-f3tZXzJmkl5uEt5wxzE3Y?usp=sharing
	Type	類別圖
	ChatGPT Version	V3.5
	ChatGPT Prompt	請以以下需求，用 plantuml 的格式，輸出 Class Diagram
	ChatGPT Code	<pre> @startuml class Customer { - name: string - selectedItems: list<Item> + chooseItem(item: Item): void + chooseCustomization(customization: Customization): void } class Item { - name: string - price: float - customizations: list<Customization> + getPrice(): float } class Customization { </pre>

		<pre>- name: string - options: list<string> } class System { + displayPrice(item: Item): void + displayCustomizations(item: Item): void + calculateTotal(customer: Customer): float + generateReceipt(customer: Customer): void + generateDailyReport(): void + generateMonthlyReport(): void } Customer -- Item : selects > Customer -- Customization : selects > System -- Item : displays > System -- Customization : displays > System -- Customer : calculates, generates > @enduml</pre>
--	--	---

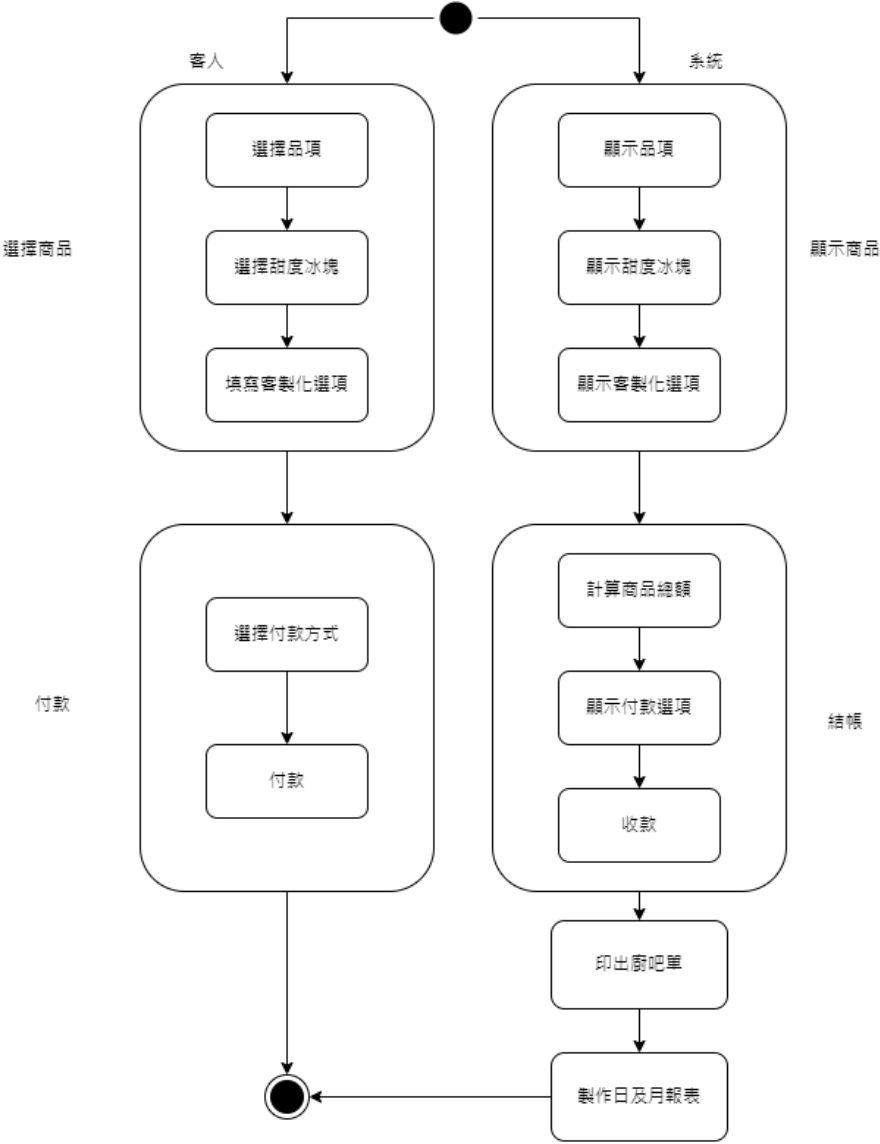
	<p>Diagram</p> <pre>classDiagram class System { <<methods>> displayPrice(item: Item): void displayCustomizations(item: Item): void calculateTotal(customer: Customer): float generateReceipt(customer: Customer): void generateDailyReport(): void generateMonthlyReport(): void } class Customer { <<attributes>> name: string selectedItems: list<Item> <<methods>> chooseItem(item: Item): void chooseCustomization(customization: Customization): void } class Item { <<attributes>> name: string price: float customizations: list<Customization> <<methods>> getPrice(): float } class Customization { <<attributes>> name: string options: list<string> } System --> Customer : calculates, generates Customer --> Item : selects Customer --> Customization : selects Item --> System : displays Customization --> System : displays</pre> <p>System</p> <ul style="list-style-type: none">• displayPrice(item: Item): void• displayCustomizations(item: Item): void• calculateTotal(customer: Customer): float• generateReceipt(customer: Customer): void• generateDailyReport(): void• generateMonthlyReport(): void <p>Customer</p> <ul style="list-style-type: none">□ name: string□ selectedItems: list<Item>• chooseItem(item: Item): void• chooseCustomization(customization: Customization): void <p>Item</p> <ul style="list-style-type: none">□ name: string□ price: float□ customizations: list<Customization>• getPrice(): float <p>Customization</p> <ul style="list-style-type: none">□ name: string□ options: list<string>
MyLearnin	1. 善用 chatgpt 可以快速整理出圖表架構

	g	<div>2. Chatgpt 在理解需求上我們不太一樣</div> <div>3. 要將報表另外生成一個類別會比較妥當</div>

Hw4---狀態圖

Methods	My Outcomes	
draw.io	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1slpSv7mOiWlcvtuJVuJlwVwl5TZSYngf?usp=sharing
	Type	狀態圖

Diagram

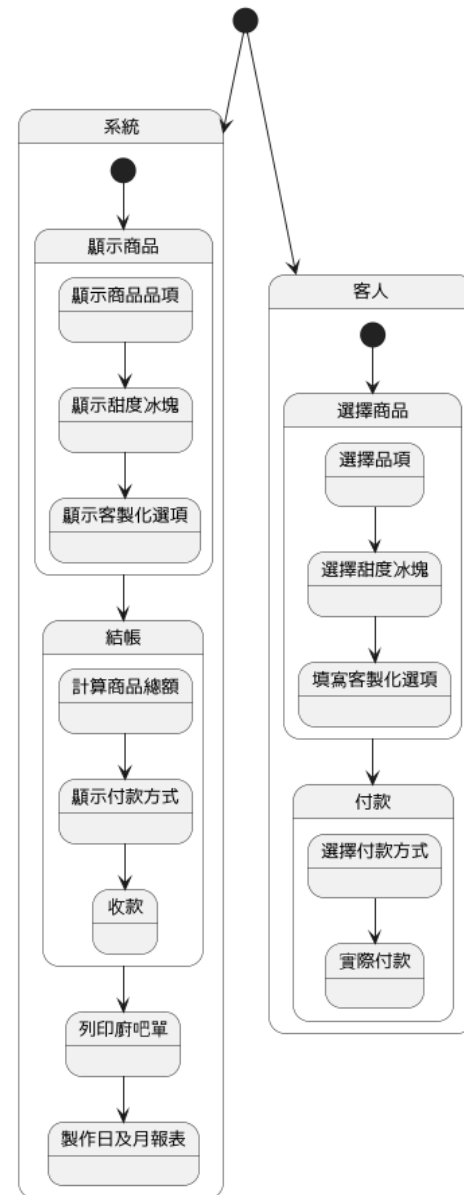


	MyLearning	1. 依照修改過後的 plantuml 畫一次就很快了，而且可以依照美觀進行調整 2. 格子很多不好拉 3. 沒有外框的選項，只能手動拉，在外框寫字
PlantUML	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1slpSv7mOiWlcvtuJVuJlwVwl5TZSYnqf?usp=sharing
	Type	狀態圖
	Code	<pre> @startuml scale 350 width [*] --> 系統 [*] --> 客人 state 系統 { [*] -->顯示商品 顯示商品-->結帳 結帳-->列印廚吧單 列印廚吧單-->製作日及月報表 state 顯示商品 { 顯示商品品項-->顯示甜度冰塊 顯示甜度冰塊-->顯示客製化選項 state 顯示商品品項 { } } state 顯示甜度冰塊 { </pre>

		<pre> } state 顯示客製化選項 { } } state 結帳 { 計算商品總額-->顯示付款方式 顯示付款方式-->收款 state 計算商品總額 { } state 顯示付款方式 { } state 收款 { } } state 列印廚吧單 { } state 製作日及月報表 { } } state 客人 { [*] -->選擇商品 選擇商品 -->付款 state 選擇商品 {</pre>
--	--	--

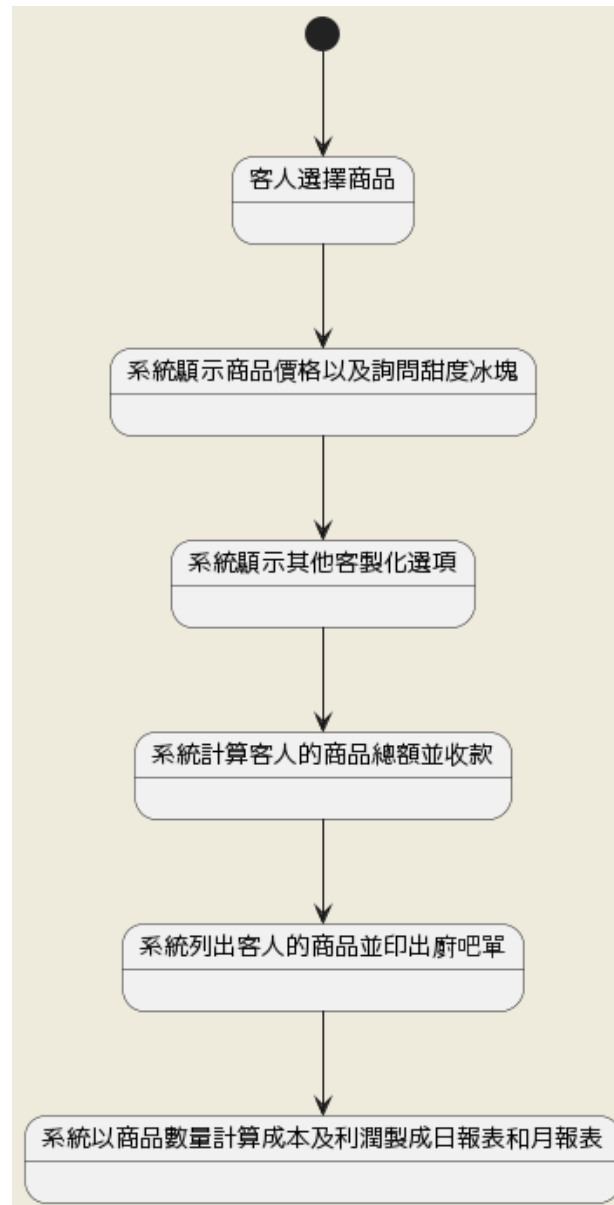
		<div>選擇品項-->選擇甜度冰塊</div> <div>選擇甜度冰塊-->填寫客製化選項</div> <div>state 選擇品項 {</div> <div> }</div> <div>state 選擇甜度冰塊 {</div> <div> }</div> <div>state 填寫客製化選項 {</div> <div> }</div> <div>}</div> <div>state 付款 {</div> <div> 選擇付款方式-->實際付款</div> <div> state 選擇付款方式 {</div> <div> }</div> <div> state 實際付款 {</div> <div> }</div> <div> }</div> <div>}</div> <div>@enduml</div>
--	--	--

Diagram



	MyLearning	1. 程式碼正確也是能畫出美觀的圖 2. 如果 state 名稱重複，他會不知道該連哪個，就會出現錯誤，或是連錯的情況 3. 狀態圖和活動圖有一點點相似
ChatGPT	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1sIpSv7mOiWlcvtuJVuJlwVwl5TZSYngf?usp=sharing
	Type	狀態圖
	ChatGPT Version	V3.5
	ChatGPT Prompt	請以以下需求，用 plantuml 的格式，輸出 Class Diagram
	ChatGPT Code	<pre> @startuml skinparam backgroundColor #EEEBDC [*] --> 客人選擇商品 客人選擇商品 --> 系統顯示商品價格以及詢問甜度冰塊 系統顯示商品價格以及詢問甜度冰塊 --> 系統顯示其他客製化選項 系統顯示其他客製化選項 --> 系統計算客人的商品總額並收款 系統計算客人的商品總額並收款 --> 系統列出客人的商品並印出廚吧單 系統列出客人的商品並印出廚吧單 --> 系統以商品數量計算成本及利潤製成日報表和月報表 @enduml </pre>

Diagram

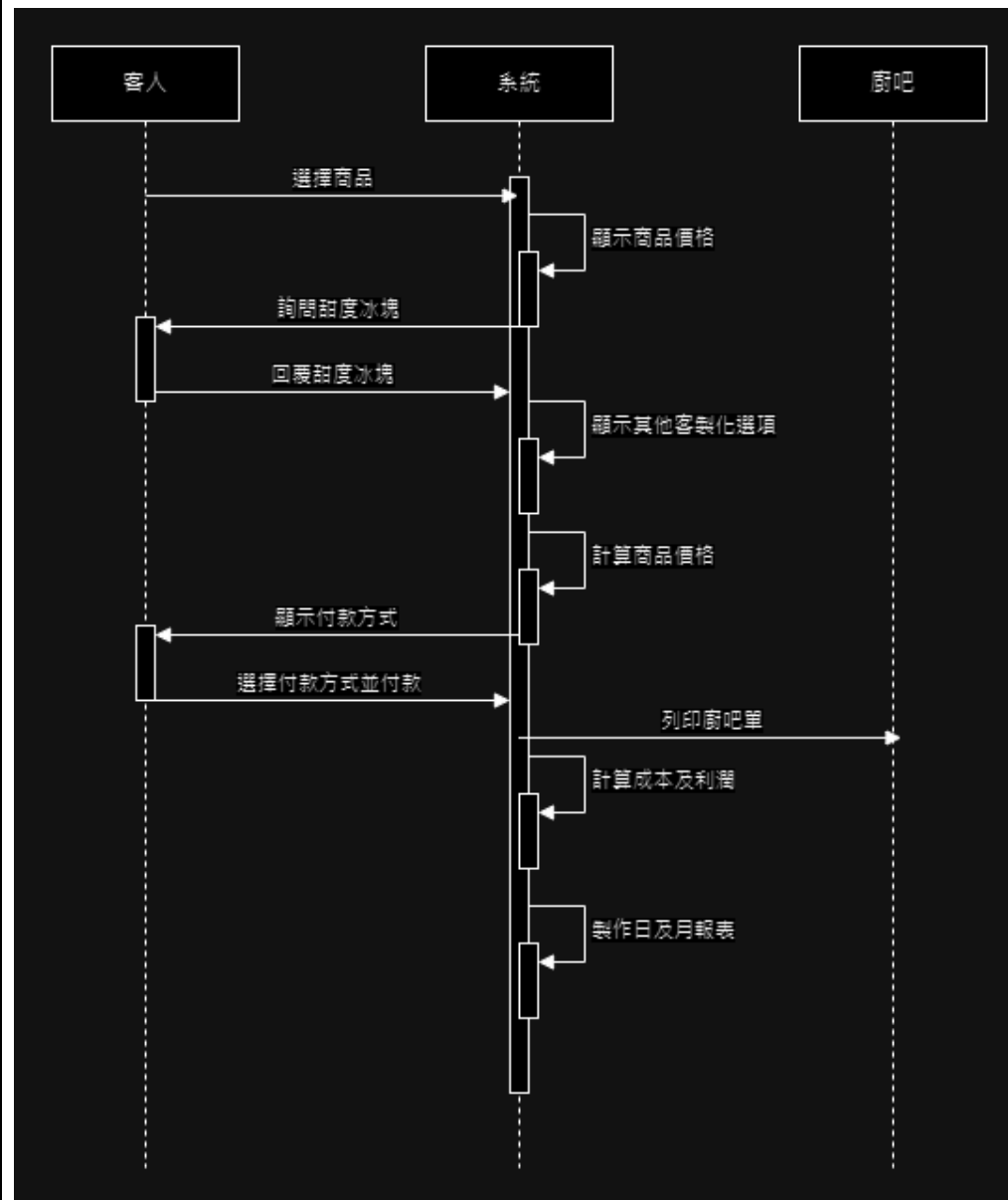


	MyLearning	<ol style="list-style-type: none">1. Chatgpt 的理解和人不一樣，可能是條件寫得過於簡單，他給的圖過於簡潔2. 狀態圖和活動圖有一點點相似

Hw5---時序圖

Methods	My Outcomes	
draw.io	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1klBiP7m7a5smAOg_qBD5ZA0dxuVIG0RQ?usp=sharing
	Type	時序圖

Diagram



	MyLearning	<ol style="list-style-type: none"> 1. 依照修改過後的 plantuml 畫一次就很快了，而且可以依照美觀進行調整 2. 牽線過程要精準隊到每個物件的錨點，不然很不好拉
PlantUML	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1klBiP7m7a5smAOg_qBD5ZA0dxuVIG0RQ?usp=sharing
	Type	時序圖
	Code	<pre> @startuml participant 客人 participant 系統 participant 廚吧 客人 -> 系統: 選擇商品 activate 系統 系統 -> 系統: 顯示商品價格 系統 -> 客人: 詢問甜度、冰塊 activate 客人 客人 -> 系統: 回覆甜度、冰塊 deactivate 客人 系統 -> 系統: 顯示其他客製化選項 系統 -> 系統: 計算商品總額 系統 -> 客人: 顯示付款選項 activate 客人 </pre>

		<p>客人 -> 系統: 選擇付款方式並付款</p> <p>deactivate 客人</p> <p>系統 -> 廚吧: 列出商品並印出廚吧單</p> <p>系統 -> 系統: 計算成本及利潤</p> <p>系統 -> 系統: 製成日報表和月報表</p> <p>deactivate 系統</p> <p>@enduml</p>
--	--	---

	Diagram	<pre> sequenceDiagram participant 客人 participant 系統 participant 廚吧 客人->>系統: 選擇商品 activate 系統 系統->>系統: 顯示商品價格 系統->>客人: deactivate 系統 客人->>系統: 詢問甜度、冰塊 activate 系統 系統->>系統: 回覆甜度、冰塊 系統->>客人: deactivate 系統 客人->>系統: 回覆甜度、冰塊 activate 系統 系統->>系統: 顯示其他客製化選項 系統->>系統: 計算商品總額 系統->>系統: 系統->>客人: 顯示付款選項 deactivate 系統 客人->>系統: 選擇付款方式並付款 activate 系統 系統->>廚吧: 列出商品並印出廚吧單 系統->>系統: 計算成本及利潤 系統->>系統: 製成日報表和月報表 系統->>系統: deactivate 系統 </pre> <p>The diagram illustrates the workflow of a cafe system. It involves three main participants: the Customer (客人), the System (系統), and the Barista (廚吧). The process begins with the Customer selecting a product, which triggers the System to display prices and allow for customization (like sweetness and ice). After the Customer provides these details, the System calculates the total and presents payment options. Once payment is made, the System generates a receipt for the Barista, calculates costs/profits, and finally produces daily and monthly reports.</p>
	MyLearning	<ol style="list-style-type: none"> 1. 先用 chatgpt 生成一次，在自己針對錯誤和缺少的地方修改會比較快且較正確 2. Plantuml 有深色模式可以使用，連輸出的圖都是深色的 3. 最後一個 deactivate 沒辦法延長到底

ChatGPT	Items	Content
	SharedFile	https://drive.google.com/drive/folders/1klBiP7m7a5smAOg_qBD5ZA0dxuVlG0RQ?usp=sharing
	Type	時序圖
	ChatGPT Version	V3.5
	ChatGPT Prompt	<p>請依以下需求，用 plantuml 格式輸出時序圖：</p> <ol style="list-style-type: none"> 1.客人選擇商品，甜度冰塊及客製化選項 2.系統顯示商品價格以及詢問客人飲料甜度冰塊 3.系統顯示其他客製化選項 4.系統計算客人的商品總額並收款 5.系統列出客人的商品並印出廚吧單 6.系統以商品數量計算成本及利潤製成日報表和月報表
	ChatGPT Code	<pre>@startuml participant 客人 participant 系統 participant 廚吧 客人 -> 系統: 選擇商品、甜度、冰塊、客製化選項 activate 系統 系統 -> 系統: 顯示商品價格 系統 -> 客人: 詢問甜度、冰塊 activate 客人 客人 -> 系統: 回覆甜度、冰塊 deactivate 客人</pre>

		<p>系統 -> 系統: 顯示其他客製化選項</p> <p>系統 -> 系統: 計算商品總額</p> <p>系統 -> 系統: 收款</p> <p>系統 -> 系統: 列出商品並印出廚吧單</p> <p>系統 -> 系統: 計算成本及利潤</p> <p>系統 -> 系統: 製成日報表和月報表</p> <p>activate 廚吧</p> <p>系統 -> 廚吧: 廚吧單</p> <p>deactivate 廚吧</p> <p>deactivate 系統</p> <p>@enduml</p>
--	--	--

	Diagram	<pre> sequenceDiagram participant 客人 participant 系統 participant 廚吧 客人->>系統: 選擇商品、甜度、冰塊、客製化選項 activate 系統 系統->>客人: 顯示商品價格 deactivate 系統 客人->>系統: 詢問甜度、冰塊 activate 系統 系統->>客人: 回覆甜度、冰塊 deactivate 系統 系統->>系統: 顯示其他客製化選項 系統->>系統: 計算商品總額 系統->>系統: 收款 系統->>系統: 列出商品並印出廚吧單 系統->>系統: 計算成本及利潤 系統->>系統: 製成日報表和月報表 系統->>廚吧: 廚吧單 deactivate 系統 </pre> <p>The diagram illustrates the workflow of a cafe system. It involves three main actors: the Customer (客人), the System (系統), and the Barista (廚吧). The process begins with the Customer selecting items and customization options, which triggers the System to display prices. The Customer then asks for specific details like sweetness and ice, leading the System to provide feedback. Subsequently, the System performs internal calculations and generates reports, finally sending the barista order (廚吧單) to the Barista.</p>
	MyLearning	<ol style="list-style-type: none"> 1. 善用 chatgpt 可以快速整理出圖表架構 2. 這次 chatgpt 給的 code 格式不正確，剛開始沒辦法運作，後來幾經嘗試財政卻顯示圖片 3. 把需求編號 chatgpt 比較好讀懂