

國立中央大學

資訊管理學系

系統分析與設計

系統軟體設計規格書

第13組

資管三 A 107403503 吳明明

資管三A 107403504 陳婕儀

資管三B 107403532 康家慈

資管三B 107403534 林湘翎

資管三B 107403538 何滢潔

指導教授：許智誠 教授、陳以錚 教授

第二章 資料庫設計

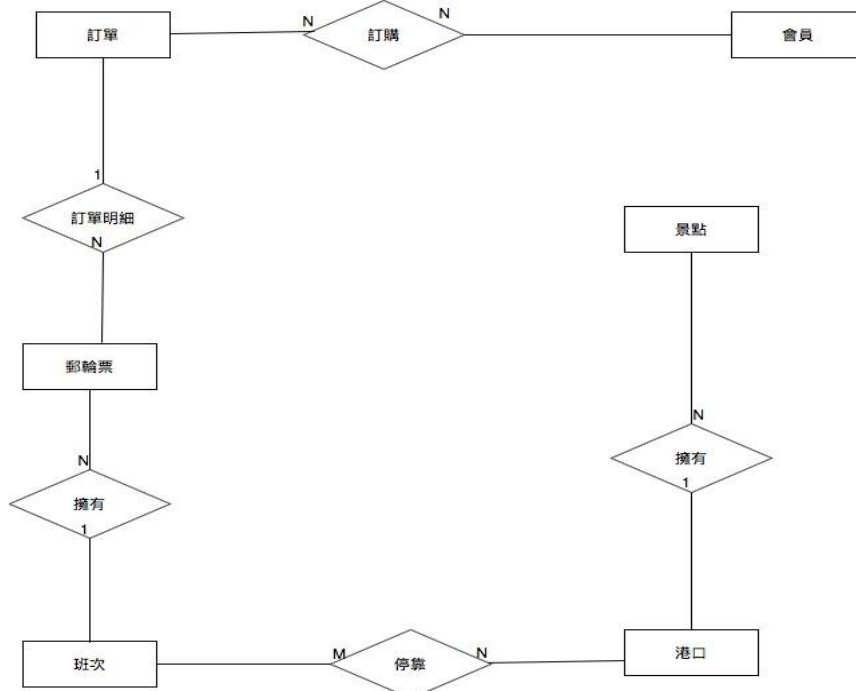
本專案之電子商務系統提供使用者簡單與便利之線上預定、訂單管理與會員資料維護之服務，對於管理者來說，亦能以簡便方式進行郵輪班次管理等後台作業，希冀不僅能提供即時與詳盡之郵輪航線介紹，更能提供良好之預定體驗。

以下分析階段之資料庫設計採用實體關係圖 (Entity-Relation Diagram)表示，並根據管理者與使用者之需求進行歸納與整理初步之系統條件。

以下詳述系統之資料庫需求，並將其整理成下圖 (圖 1)之實體關係圖共計包含 5 個實體 (Entity)、6 個關係 (Relationship)、0 個複合性實體(Compound Entity)：

- 1.一般訪客可以註冊成為會員且必須以電子郵件作為登入之帳號使用，同時系統會自動給予每位會員編號。
- 2.一般訪客與會員皆可將欲搭乘之郵輪班次加入購物車，但伺服器不儲存該資料，而存於使用者之本地端。
- 3.會員可以將購物車之郵輪班次進行結帳，並輸入住址與電話成為客戶，以完成訂單，同時可以查閱訂單之詳情。
- 4.管理者可以管理郵輪班次之異動與會員之資料，並且進行維護作業，同時要記錄帳號建立時間與異動之動作。

圖 1：實體關係圖



根據實體關係圖分析本專案所需之資料庫架構，以下將針對每張資料表進行描述：

1. 會員資料表 (members)

表 1：會員資料表(members) 之資料結構

key	名稱	類型	預設值	空值	自動增加	編碼
P.K	member_id	Int(11)	無	否	是	
	member_name	Varchar(250)	無	否		utf8mb4_general_ci
	member_password	Varchar(250)	無	否		utf8mb4_general_ci
	createdate	Datetime	無	否		
	modifieddate	Datetime	無	否		
	logindate	Datetime	0	否		
	creditcard	Int(20)	無	否		
	phone	Int(20)	無	否		
	email	Varchar(250)	無	否		utf8mb4_general_ci

member_id:為自動增加作為會員編號，不可更動，由資料庫自動產生。

modifieddate：用於紀錄會員資料最後更新時間。

createdate：用於紀錄該名會員註冊時間。

logindate：用於紀錄最近一次登入的時間。

2. 郵輪資料表 (cruises)

表 2：郵輪行程資料表 (cruises) 之資料結構

key	名稱	類型	預設值	空值	自動增加	編碼
P.K	cruises_id	Int(11)	無	否	是	
	sailing_date	Datetime	無	否		
	cruises_name	Varchar(250)	無	否		utf8mb4_general_ci
	cruises_price	Double	無	否		
	image	Varchar(250)	無	否		
	describe	Varchar(1024)	無	否		utf8mb4_general_ci

cruises_id：為自動增加作為商品編號，不可更動，由資料庫系統自動產生。

image：欄位用於紀錄郵輪圖檔之路徑位置。

3. 船票資料表 (ticket)

表 3：船票資料表 (ticket) 之資料結構

key	名稱	類型	預設值	空值	自動增加	編碼
P.K	ticket_id	Int(11)	無	否	是	
F.K	cruises_id	Int(11)	無	否	是	
	sailing_date	Datetime	無	否		
	ticket_price	Double	無	否		

ticket_id：為自動增加作為商品編號，不可更動，由資料庫系統自動產生。

sailing_date：郵輪離港日期時間

4. 訂單資料表(order)

表 4：訂單資料表(order)之資料結構

key	名稱	類型	預設值	空值	自動增加	編碼
P.K	order_id	Int(11)	無	否	是	
F.K	member_id	Int(11)	無	否	是	
F.K	cruises_id	Int(11)	無	否	是	
F.K	ticket_id	Int(11)	無	否	是	
	order_time	Datetime	0	否		
	order_status	Varchar(250)	無	否		
	order_price	Double	無	否		

order_id:為自動增加作為商品編號，不可更動，由資料庫系統自動產生。

order_time:訂單成立時間。

order_status:訂單的付款狀態，分為已付款、未付款。

5. 管理員資料表(manager)

key	名稱	類型	預設值	空值	自動增加	編碼
P.K	manager_id	Int(11)	無	否	是	
	manager_name	Varchar(250)	無	否		utf8mb4_general_ci
	manager_password	Varchar(250)	無	否		
	createdate	Datetime	無	否		
	modifieddate	Datetime	無	否		
	logindate	Datetime	0	否		

表 5：管理員資料表(manager)之資料結構

manager_id:為自動增加作為商品編號，不可更動，由資料庫系統自動產生。

6. 景點資料表(spot)

key	名稱	類型	預設值	空值	自動增加	編碼
P.K	spot_id	Int(11)	無	否	是	
F.K	stop_port_id	Int(11)	無	否	是	
	spot_name	Varchar(250)	無	否		utf8mb4_general_ci
	image	Varchar(250)	無	否		utf8mb4_general_ci
	description	Varchar(250)	無	否		utf8mb4_general_ci

表 6：景點資料表(spot)之資料結構

7. 停靠站資料表(stop port)

表 7：停靠站資料表(stop port)之資料結構

key	名稱	類型	預設值	空值	自動增加	編碼
P.K	stop_port_id	Int(11)	無	否	是	
	stop_port_name	Varchar(250)	無	否		utf8mb4_general_ci

8. 郵輪停靠站資料表(cruise stop port)

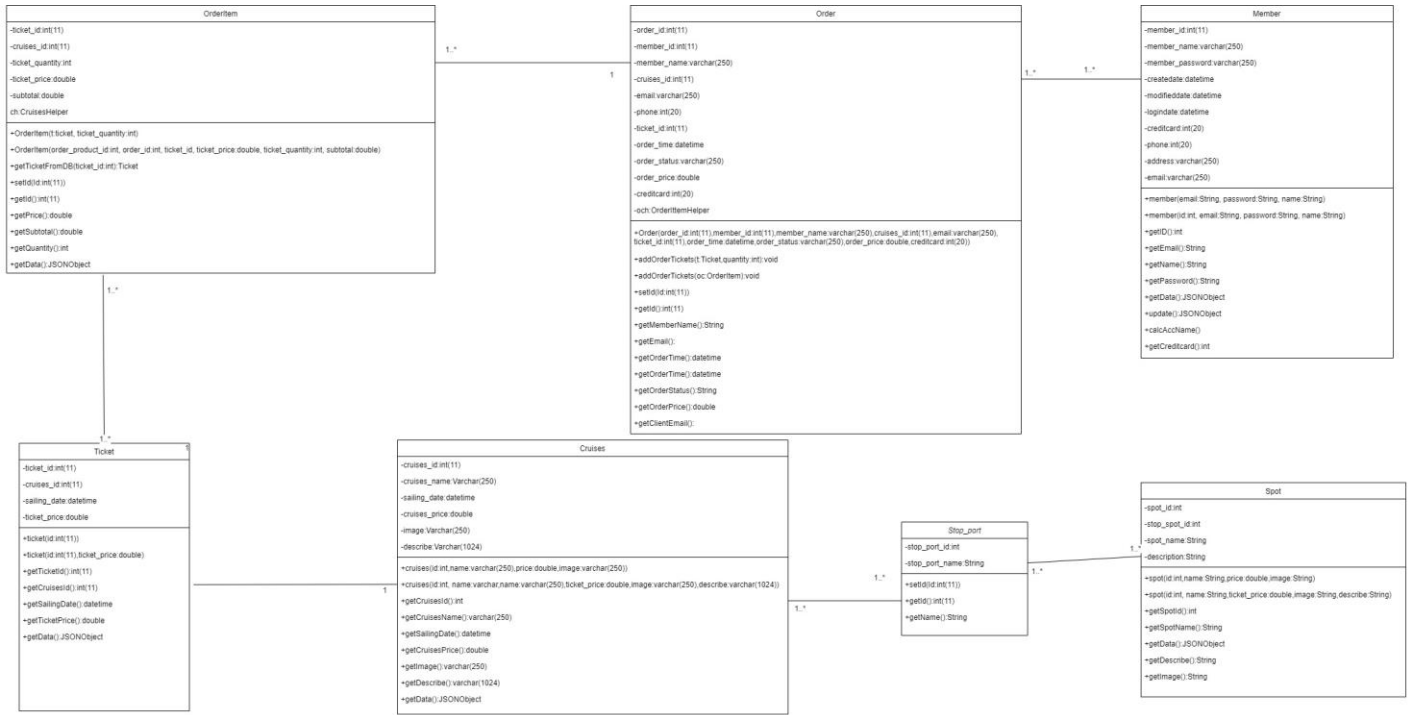
表 8：郵輪停靠站資料表(cruise stop port)之資料結構

key	名稱	類型	預設值	空值	自動增加	編碼
P.K	cruises_id	Int(11)	無	否	是	
P.K	stop_port_id	Int(11)	無	否	是	

第三章 類別圖

分析階段之類別圖 (class diagram)係依據第一份文件所述之使用案例找出並分析類別，同時也參照前章節 (第三章 資料庫設計)以建立本專案之郵輪電子商務訂購系統分析模型之類別圖。

該階段之類別圖僅列出控制 (controller)和實體之類別，其內部之詳細屬性與方法僅大略進行定義，詳細之設計細節與使用之參數屬性與方法於第三份文件—設計 (design)詳細描述：



JsonReader
-request_string String
-request HttpServletResponse
+JsonReader(request HttpServletResponse)
+getParameter(String) String
+getArray() JSONArray
+getObjet() JSONObject
response/resp_string String, response HttpServletResponse
response/resp JSONObject, response HttpServletResponse
response/status_code int, resp JSONObject, response HttpServletResponse

DBMyr
+JOBDC_DRIVER String
+DB_URL String
+USER String
+PASS String
+DBMyr()
+getConnection() Connection
+close() Statement, conn Connection
+close() ResultSet, stmt Statement, conn Connection

CruisesHelper
-ch CruisesHelper
-conn Connection
-pres PreparedStatement
+CruisesHelper()
+getHelper() CruisesHelper
+getById(id int) JSONObject
+getById(id int) JSONObject
+getLoginTimesStatus(m Member) JSONObject
+checkDuplicate(m Member) boolean
+create(m Member) JSONObject
+update(m Member) JSONObject
+updateLoginTimes(m Member) void
+updateStatus(m Member, status String) void
+getByStopPort(String id) JSONArray

StopPortHelper
-soph StopPortHelper
-conn Connection
-pres PreparedStatement
+StopPortHelper()
+getHelper() StopPortHelper
+getAK() JSONObject
+getById(id int) String data, JSONObject
+getById(id int) String id, Cruises

CruisesHelper
-ch CruisesHelper
-conn Connection
-pres PreparedStatement
+CruisesHelper()
+getHelper() CruisesHelper
+getAK() JSONObject
+getById(id int) String data, JSONObject
+getById(id int) String id, Cruises
+getByStopPort(String id) JSONArray

OrderHelper
-oh OrderHelper
-conn Connection
-pres PreparedStatement
-oh OrderItemHelper
+OrderHelper()
+getHelper() OrderHelper
+createOrder Order, JSONObject
+getAK() JSONObject
+getById(id int) String id, JSONObject

OrderItemHelper
-oh OrderItemHelper
-conn Connection
-pres PreparedStatement
+OrderItemHelper()
+getHelper() OrderItemHelper
+createBy(List order_id long, ordercruises List-OrderItem-) JSONArray
+getOrderCruisesByOrder(id int) Array List-OrderItem-

SpotHelper
-sh SpotHelper
-conn Connection
-pres PreparedStatement
+SpotHelper()
+getHelper() SpotHelper
+getAK() JSONObject
+getById(id int) String data, JSONObject
+getById(id int) String id, Cruises
+getByStopPort(String id) JSONArray

OrderDetailController
-spot_name
-description
+calcSubTotal()
+calcTax()
+calcTotal()
+calcTotalWeight()

OrderController
-serialVersionUID long
-ch CruisesHelper
-oh OrderHelper
+OrderController()
+doPost(request HttpServletRequest, response HttpServletResponse)
+doGet(request HttpServletRequest, response HttpServletResponse)
+doDelete(request HttpServletRequest, response HttpServletResponse)
+doPut(request HttpServletRequest, response HttpServletResponse)

CruisesController
-serialVersionUID long
-ch CruisesHelper
+CruisesController()
+doPost(request HttpServletRequest, response HttpServletResponse)
+doGet(request HttpServletRequest, response HttpServletResponse)
+doDelete(request HttpServletRequest, response HttpServletResponse)
+doPut(request HttpServletRequest, response HttpServletResponse)

StopPortController
-message String
-output String
+StopPortController()
+doPost(request HttpServletRequest, response HttpServletResponse)
+doGet(request HttpServletRequest, response HttpServletResponse)
+doDelete(request HttpServletRequest, response HttpServletResponse)
+doPut(request HttpServletRequest, response HttpServletResponse)

CheckoutController
-message String
-output String
+init()
+doPost(request HttpServletRequest, response HttpServletResponse)
+doGet(request HttpServletRequest, response HttpServletResponse)
+doPut(request HttpServletRequest, response HttpServletResponse)

MemberController
-serialVersionUID long
-mh MemberHelper
+doPost(request HttpServletRequest, response HttpServletResponse)
+doGet(request HttpServletRequest, response HttpServletResponse)
+doDelete(request HttpServletRequest, response HttpServletResponse)
+doPut(request HttpServletRequest, response HttpServletResponse)

OrderItemController
-message String
-output String
+init()
+doPost(request HttpServletRequest, response HttpServletResponse)
+doGet(request HttpServletRequest, response HttpServletResponse)
+doDelete(request HttpServletRequest, response HttpServletResponse)
+doPut(request HttpServletRequest, response HttpServletResponse)

SpotController
-serialVersionUID long
-sh SpotHelper
+SpotController()
+doPost(request HttpServletRequest, response HttpServletResponse)
+doGet(request HttpServletRequest, response HttpServletResponse)
+doDelete(request HttpServletRequest, response HttpServletResponse)
+doPut(request HttpServletRequest, response HttpServletResponse)

圖 2：分析階段之類別圖

第四章 系統循序圖

本章節主要依照第一份文件需求所產生之使用案例圖(use case)與第二份 文件分析之邏輯階段活動圖與強韌圖為基礎，進行設計階段之循序圖設計，將 每個使用案例進行闡述。於此階段，需要有明確之類別(class)名稱與呼叫之 方法(method)與傳入之變數名稱與型態等細部設計之內容。

4.1 使用案例圖

依據第一份文件—系統軟體需求規格書 (Software Requirement Specification)，本郵輪電子商務線上訂購系統預計共有 4 位動作者與 28 個使用案例，並依照不同之模組區分成不同子系統共計八個子系統，其中包含以下：01 會員管理子系統、02 景點資訊子系統、03 郵輪班次資訊子系統、04 預定郵輪班次子系統、05 結帳船票子系統、06 訂單管理子系統、07 郵輪班次管理系統、08 管理者子系統，如下圖所示：

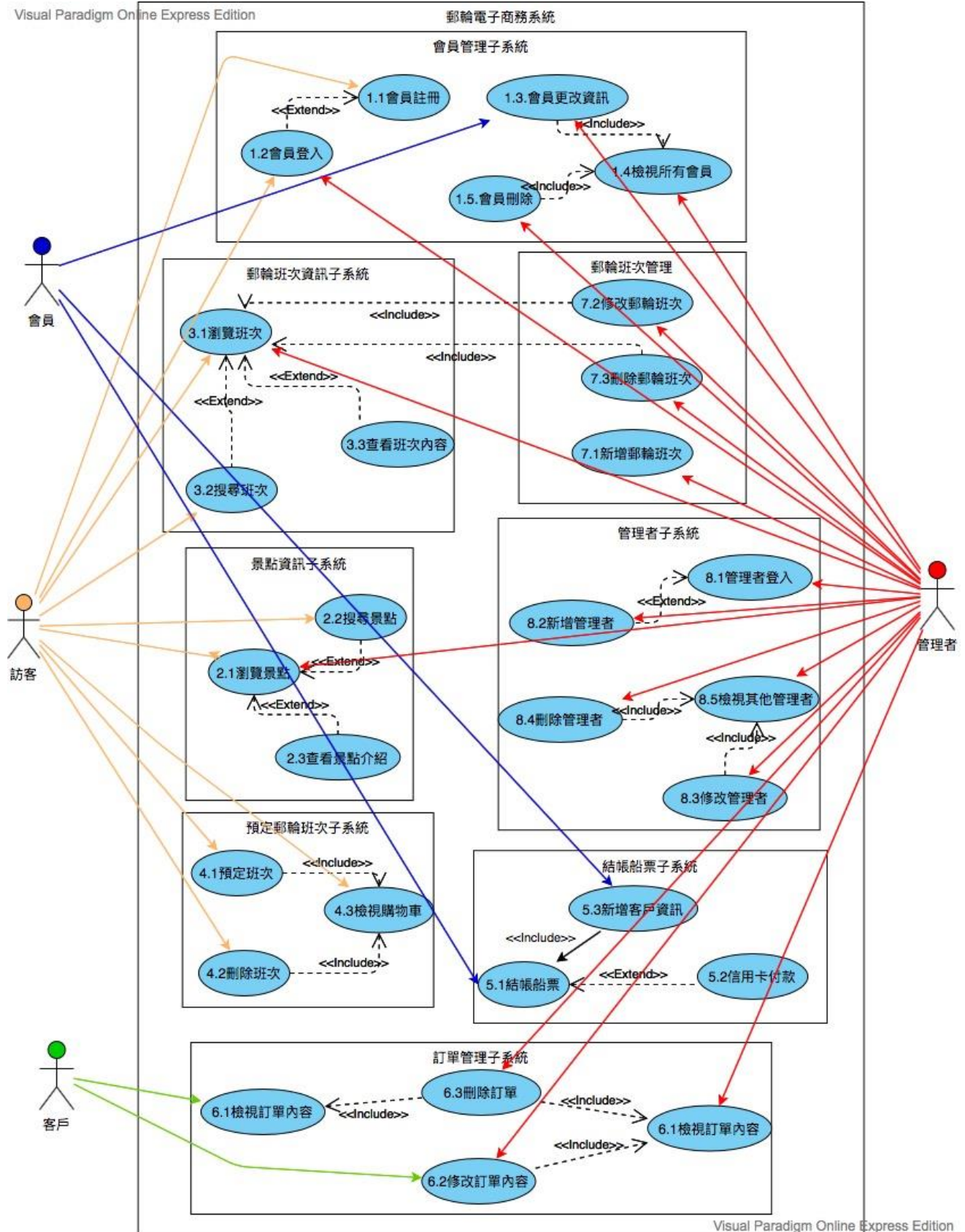


圖 3：使用案例關係圖

4.2 Use Case 實做之循序圖

4.2.1 商業流程編號 2.0：景點模組

在 Use Case 2.0 中所使用到的功能（包含：瀏覽、搜尋等），以下簡要說明各功能，由於本範例僅實作後台郵輪班次搜尋，因此僅就此部分進行詳細說明，實際上所有模組之所有功能皆需要詳細說明：

1. Use Case 2.1：瀏覽景點

此使用案例即為景點之目錄頁面，可供一般訪客跟會員查看景點

2. Use Case 2.2：搜尋景點

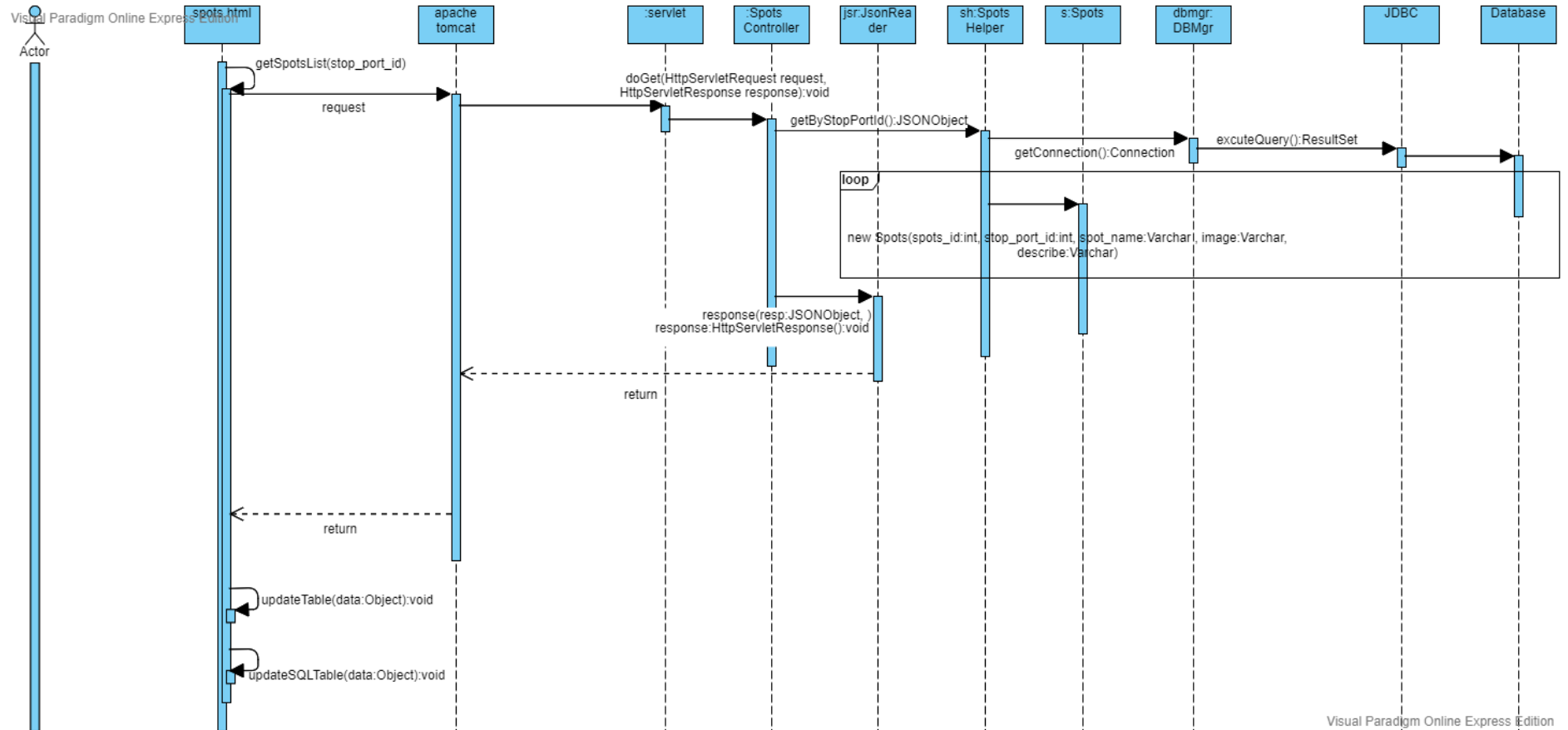
此使用案例用於說明如何協助一般訪客或會員透過停靠地搜尋景點。比對後方資料庫資料找出符合該地點擁有的景點，並傳回呈現目錄頁面。

3. Use Case 2.3：查看景點介紹

此使用案例限點選任意景點目錄頁面之景點才得以執行，呈現景點之名稱、圖片、詳細介紹。

與該模組相關之頁面於下表（表 9）進行說明：

HTML	關聯 Use Case	說明
cruises.html status.html	2.2 搜尋景點	會員透過停靠站搜尋附近相關景點



1. 會員進到 `spots.html` 在搜尋欄填入停靠站後，按下搜尋。
2. 完成讀取後，透過 JavaScript 之 `getCruisesList()`送出 GET 請求。
3. 後端以 `SpotController` 之 `doGet()`進行處理，以 `JsonReader` 以 `JsonReader` 判斷不存在參數後，使用 `SpotHelper` 物件的 `getByStopPortId()`方法直接取得對應景點。
4. 透過迴圈將 `ResultSet` 內所有景點取出，透過資料庫檢索並逐一將每個景點 new 每個 `Spots` 物件，最終將所有景點資料封裝進行回傳。
5. 查詢成功則透過 JavaScript 之 `updateTable()`更新會員列表之表格與 `updateSQLTable()`更新 SQL 表格之內容。

4.2.2 商業流程編號 3.0：郵輪班次模組

在 Use Case 3.0 中所使用到的功能（包含：瀏覽、搜尋等），以下簡要說明各功能，由於本範例僅實作後台郵輪班次搜尋，因此僅就此部分進行詳細說明，實際上所有模組之所有功能皆需要詳細說明：

1. Use Case 3.1：瀏覽郵輪班次

此使用案例即為郵輪之目錄頁面，可供一般訪客與會員瀏覽郵輪班次之名稱與圖片。

2. Use Case 3.2：搜尋郵輪班次

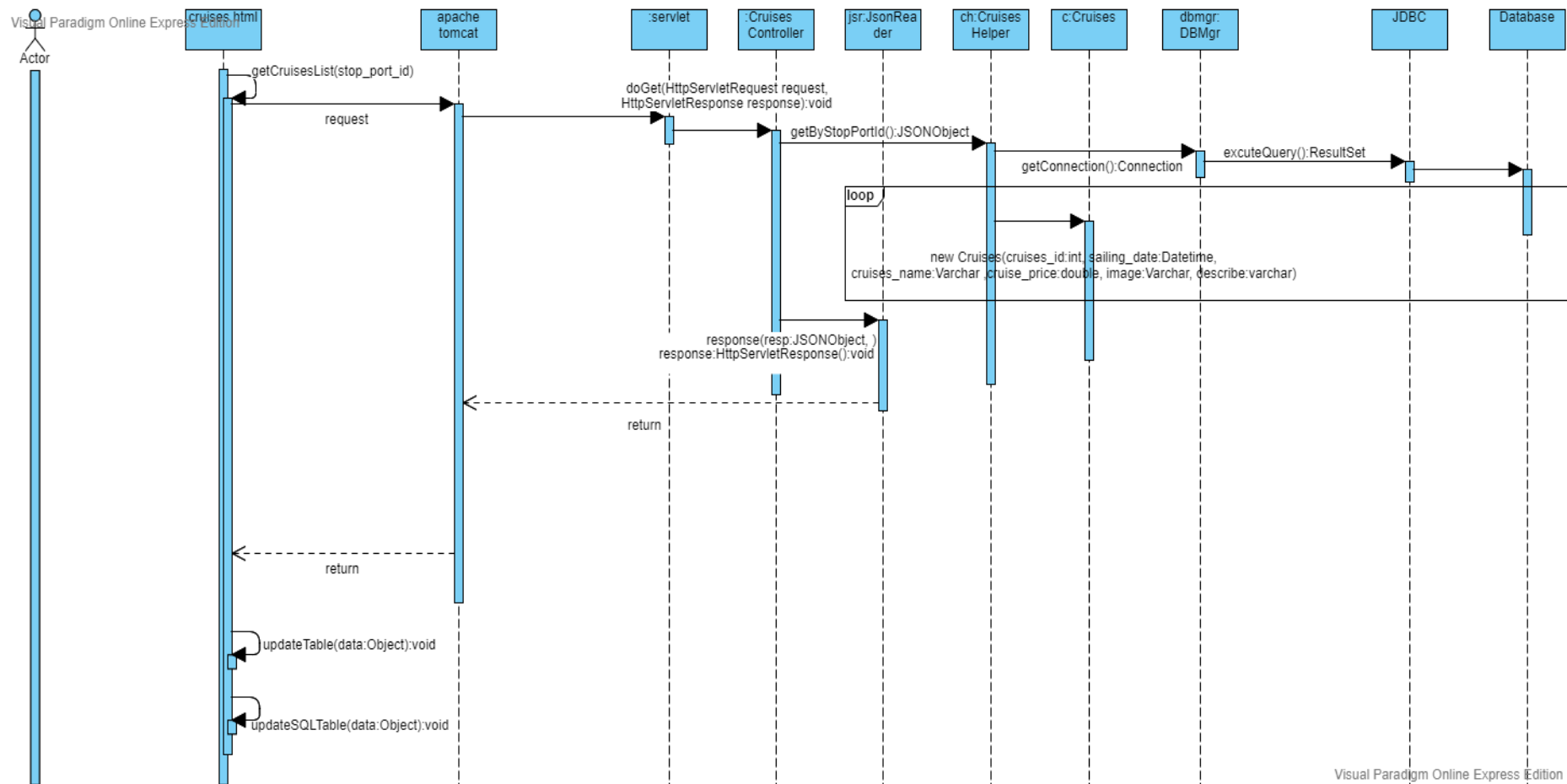
此使用案例用於說明如何協助一般訪客或會員透過停靠地搜尋郵輪班次。比對後方資料庫資料找出符合該地點之郵輪班次，並傳回呈現目錄頁面。

3. Use Case 3.3：查看郵輪班次內容

此使用案例限點選任意郵輪目錄頁面之郵輪才得以執行，呈現郵輪之名稱、圖片、詳細介紹、地點、時間與價格。

與該模組相關之頁面於下表（表 9）進行說明：

HTML	關聯 Use Case	說明
cruises.html status.html	3.2 搜尋郵輪班次	會員透過停靠站搜尋郵輪班次



1. 會員進到 cruses.html 在搜尋欄填入停靠站後，按下搜尋。
2. 完成讀取後，透過 JavaScript 之 getCruisesList()送出 GET 請求。
3. 後端以 CruisesController 之 doGet()進行處理，以 JsonReader 以 JsonReader 判斷不存在參數後，使用 CruisesHelper 物件的 getByStopPortId()方法直接取得對應郵輪
4. 透過迴圈將 Resultset 內所有遊輪資料取出，透過資料庫檢索並逐一將每個郵輪 new 每個 Cruises 物件，最終將所有郵輪資料封裝進行回傳。
5. 查詢成功則透過 JavaScript 之 updateTable()更新會員列表之表格與 updateSQLTable()更新 SQL 表格之內容

