

國 立 中 央 大 學

資 訊 管 理 學 系

111 (一) 系 統 分 析 與 設 計

系統軟體分析規格書

第五組

資管三 B 109403048 林子恩

資管三 B 109403533 林采璇

資管三 B 109403535 洪琬哲

資管三 B 109403538 江珮彤

資管三 B 109403543 賴立彬

資管三 B 109403546 謝謹煜

指導教授： 葉羅堯 教授

中 華 民 國 1 1 1 年 1 1 月 3 0 日

分 析 文 件 評 分 標 準

各項合起來就是這個文件的分數(100%)。

活動圖部分(10%):

是否將使用各使用案例的細節適當的轉成正確的活動圖
(能夠合理的轉成強韌圖，並且精確反映使用案例)

強韌圖(20%):

強韌圖是否有適當的切割出 boundary, control, 跟
entity 物件強韌圖的連接順序是否正確

ER 圖(20%):

是否有包含”所有”必須的資料表

各表格的欄位定義是否能與使用個案配合

各表格的鍵值設定(主鍵與外鍵 primary key/ foreign key)是

否正確類別圖(30%):

是否有包含”所有”必須的 class (反映強韌圖與 ER 描述
的類別) 各個類別的欄位、方法定義是否正確

各個類別之間相互關係的表達是否正確

整體文件部分(20%):

在最後，將第二、三、四章以外的部分正確撰寫，反映該專案的實際規格

目錄

目錄	iii
表目錄	iv
圖目錄	v
版本修訂	1p
第 1 章 簡介	2p
1.1 文件目的	2p
1.2 系統範圍	2p
1.3 參考文件	2p
1.4 文件架構	3p
第 2 章 系統動態分析圖	4p
2.1 使用案例圖	4p
2.2 使用案例 4.0：活動參加管理	6p
2.2.1 使用案例 4.1：報名活動	6p
2.2.1.1 活動圖	6p
2.2.1.2 強韌圖	8p
2.3 使用案例 5.0：活動管理模組	9p
2.3.1 使用案例 5.3：修改活動資訊	9p
2.3.1.1 活動圖	9p
2.3.1.2 強韌圖	11p
第 3 章 資料庫設計	12p
第 4 章 類別圖	17p
第 5 章 系統開發環境	19p
5.1 系統架構圖	19p
5.2 MVC 架構	20p

表目錄

表 1：商業流程編號 4.0 活動參加管理模組	6p
表 2：商業流程編號 5.0 活動管理模組	9p
表 3：分析階段之會員資料表 (TB_Member) 資料結構	14p
表 4：分析階段之會員隱密資料表 (TB_MemberCredential) 資料結構	14p
表 5：分析階段之管理員資料表 (TB_Administrator) 資料結構	14p
表 6：分析階段之管理員隱密資料表 (TB_AdministratorCredential) 資料結構	15p
表 7：分析階段之活動資料表 (TB_Event) 資料結構	15p
表 8：分析階段之活動類別資料表 (TB_EventType) 資料結構	15p
表 9：分析階段之活動場次資料表 (TB_EventSessions) 資料結構	16p
表 10：分析階段之活動場次參加管理資料表 (TB_SessionMemberDetail) 資料結構	16p
表 11：分析階段之活動報名狀態資料表 (TB_ApplyStatus) 資料結構	16p
表 12：分析階段之收藏資料表 (TB_Collection) 資料結構	16p

圖目錄

圖 1：中央大學藝文中心活動報名系統使用案例圖	5p
圖 2：使用案例 4.1 報名活動之活動圖	7p
圖 3：使用案例 4.1 報名活動之強韌圖	8p
圖 4：使用案例 5.3 修改活動資訊之活動圖	10p
圖 5：使用案例 5.3 修改活動資訊之強韌圖	11p
圖 6：實體關係圖	13p
圖 7：分析階段之類別圖	18p
圖 8：分析階段之架構圖	19p
圖 9：MVC 架構	20p

版本修訂

版本	修訂者	修訂簡述	日期
V0.1.0	全員	Draft	2022/11/18
V0.1.1	全員	實體關係圖、類別圖	2022/11/21
V0.1.2	全員	細節修訂	2022/11/27
V1.1.0	全員	細節修訂	2022/12/28

第 1 章 簡介

軟體分析規格書 (software analysis description, SAD) 係依據軟體產品、專案之主要使用者之需求規格文件 (software requirements specification, SRS)，主要用於描述邏輯之軟體架構與系統範圍之文件。藉由本文件得以分析軟體系架構之目的，並作為軟體設計階段之依據。

本專案之文件採用統一塑模語言 (Unified Modeling Language, UML) 說明與建構本系統之方法與架構，包含：使用案例圖 (Use Case Diagram)、實體關係圖 (Entity-Relation Diagram) 與類別圖 (conceptual modelling) 等。

1.1 文件目的

本文件之目的用於提供軟體系統開發人員分析之規範與藍圖，透過軟體分析規格書，開發人員可以明確了解軟體系統之邏輯與運作方式，並得以此為據遵照共同訂定之規格設計軟體系統。

本文件針對系統之分析為邏輯階段 (logical phase) 而非是實際設計階段 (physical phase) 之內容，分析模型與系統設計與實作環境無關之邏輯結構 (logical Structure)，得以使用邊界、控制和實體物件呈現系統資訊 (information)、行為 (behavior) 和展示 (presentation) 三個層面。

1.2 系統範圍

本系統範圍用於藝文中心報名系統，其中主要包含會員、活動資訊、活動收藏、活動參加管理、活動管理、管理員等六個模組，並且能進行相關新增、查閱與維護工作，藉由此系統支持完成活動報名所需的管理流程。詳細各模組之功能與內容可參閱第一份文件系統軟體需求規格書。

1.3 參考文件

1. 系統分析與設計—需求 (Requirement)

1.4 文件架構

本文件共分為五個章節，用以闡述本專案之分析相關內容：

1. 第 1 章針對本文件進行簡介，說明本文件重要之處
2. 第 2 章依據本專案前份文件之使用者案例依序進行分析，於本章節依照使用者案例將產出所需活動圖與強韌圖。
3. 第 3 章分析本專案所需之資料庫架構與資料表內容。
4. 第 4 章則是陳列出本專案所需之類別、屬性與方法的類別圖。
5. 第 5 章說明本專案所需之系統開發環境，其中包含系統架構圖與 MVC 架構之說明。

第 2 章 系統動態分析圖

在本章節中，將透過在前一份文件中所分析之使用者案例（use case）逐一進行詳細之系統動態分析。首先須先將使用者案例之主要流程轉換成活動圖，再者依照所分析之活動圖產生強韌圖以找出分析之類別。

2.1 使用案例圖

依據第一份文件—系統軟體需求規格書（Software Requirement Specification），本活動報名系統預計共有 3 位動作者與 24 個使用案例，並依照不同之模組區分成不同子系統共計六個子系統，其中包含以下：① 會員子系統、② 活動資訊子系統、③ 活動收藏子系統、④ 活動參加管理子系統、⑤ 活動管理子系統、⑥ 管理員子系統，如下圖（圖 1）為本系統之使用案例圖：

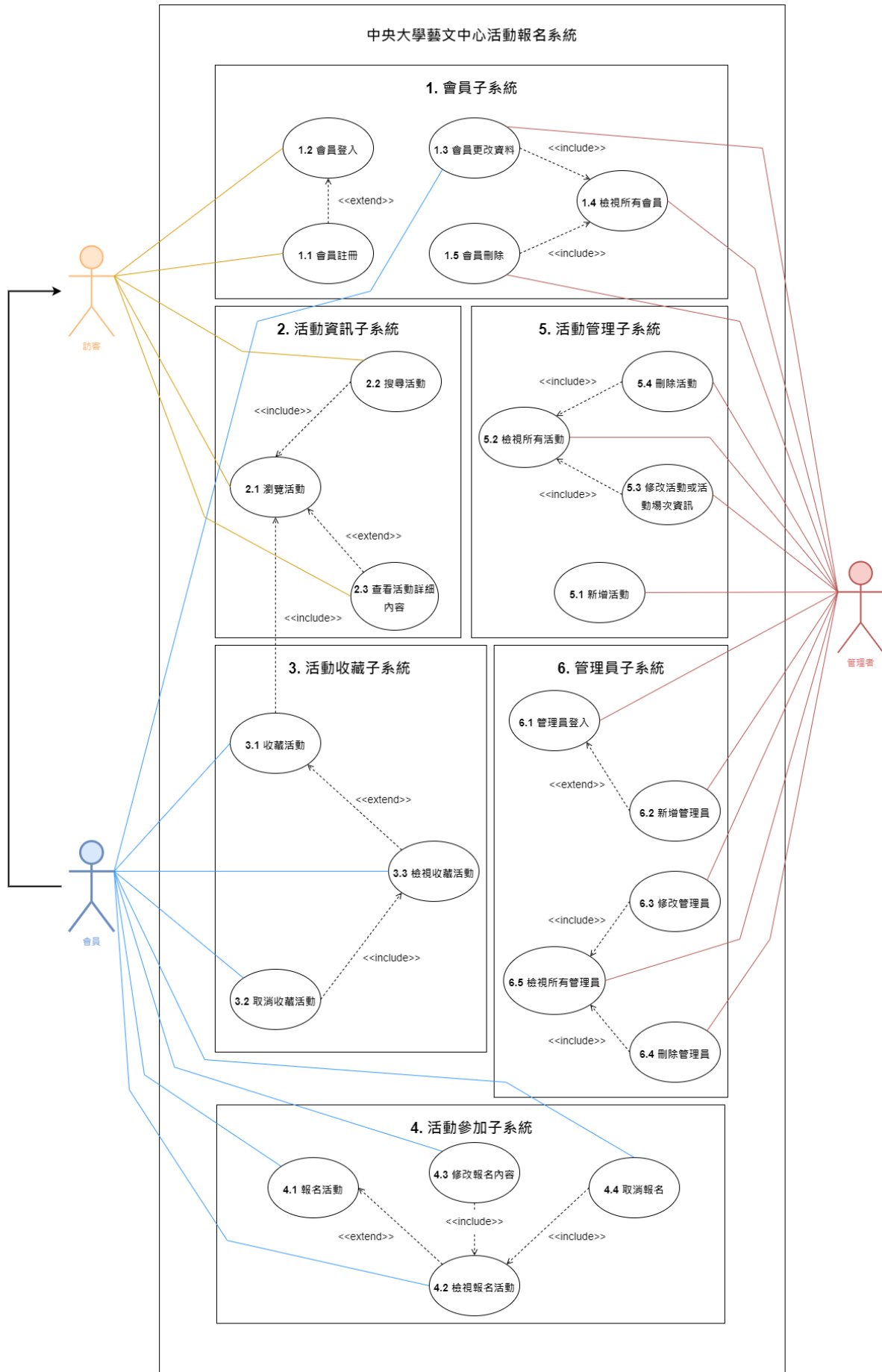


圖 1：中央大學藝文中心活動報名系統使用案例圖

根據上圖所分析之使用案例規格，需先逐一將每個使用案例轉換成活動圖與強韌圖，並逐一詳細闡述，並且進行分析。再者依據就前揭所述之活動圖、強韌圖產生所需之通訊圖與狀態機圖，進行統整並繪製出分析階段之循序圖，依照本系統之架構劃分而繪製而成。

2.2 使用案例 4.0：活動參加管理模組

本章節中，將針對活動參加管理模組進行系統動態分析圖之分析，活動參加管理模組一共包含四個功能：①報名活動、②檢視活動報名紀錄、③變更活動報名紀錄、④取消報名，以下並根據此些使用案例進行必要圖形之分析。

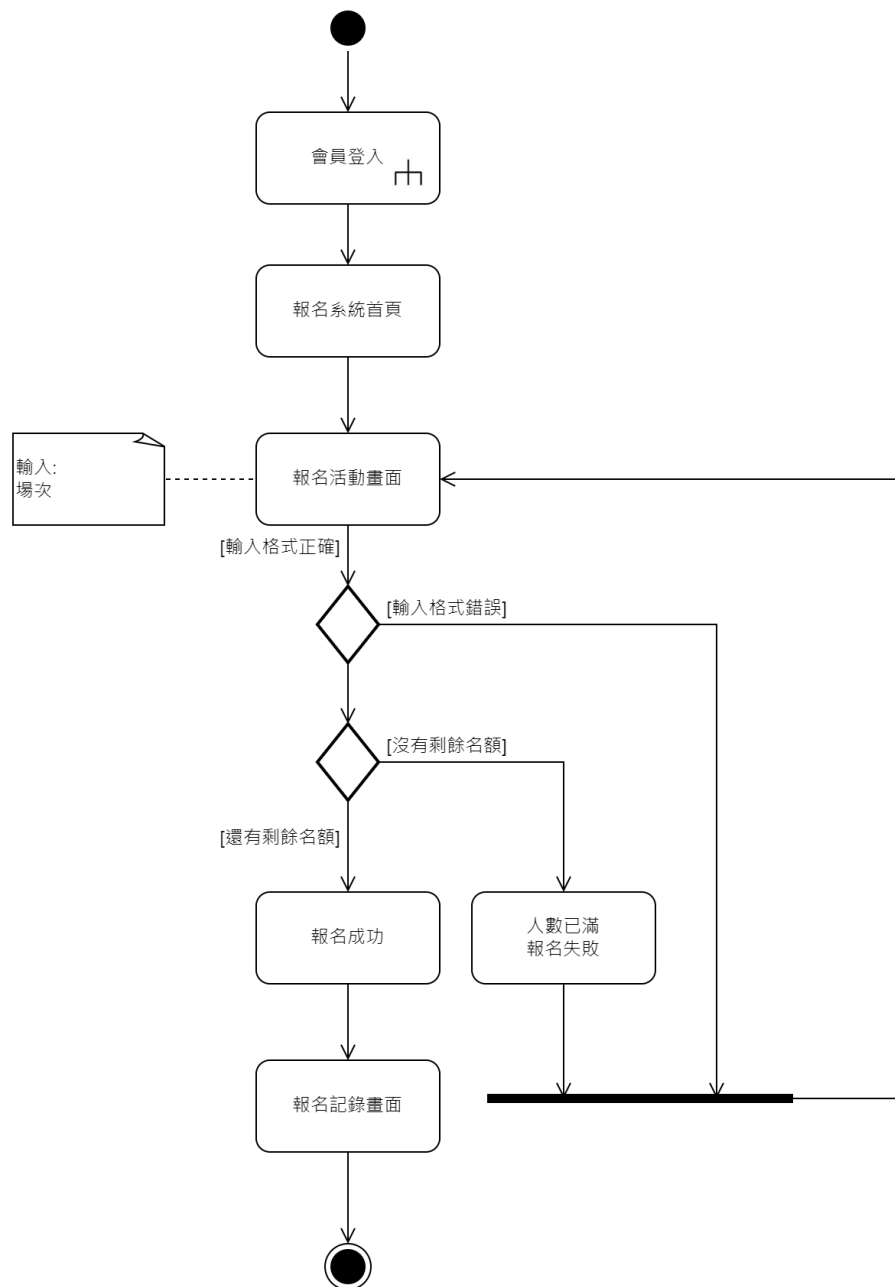
表 1：商業流程編號 4.0 活動參加管理模組

模組	功能名稱	說明
4.0 模組： 活動參加管理	4.1 報名活動	會員選擇報名活動後可產生報名相關明細
	4.2 檢視活動報名紀錄	會員可以檢視自己報名的活動
	4.3 變更活動報名紀錄	會員可以修改報名場次
	4.4 取消報名	會員可以取消報名

2.2.1 使用案例 4.1：報名活動

2.2.1.1 活動圖

- 商業流程編號「4.1 報名活動」之主要流程如下所示：
 - ✓ 1. 在報名活動畫面輸入日期與選擇場次。
 - ✓ 2. 系統驗證輸入是否正確。
 - ✓ 3. 系統驗證是否還有剩餘名額。
 - ✓ 4. 系統顯示報名失敗或顯示報名紀錄畫面。
- 透過以上主流程可將其轉換成活動圖，如下圖（圖 2）所示：



會員登入

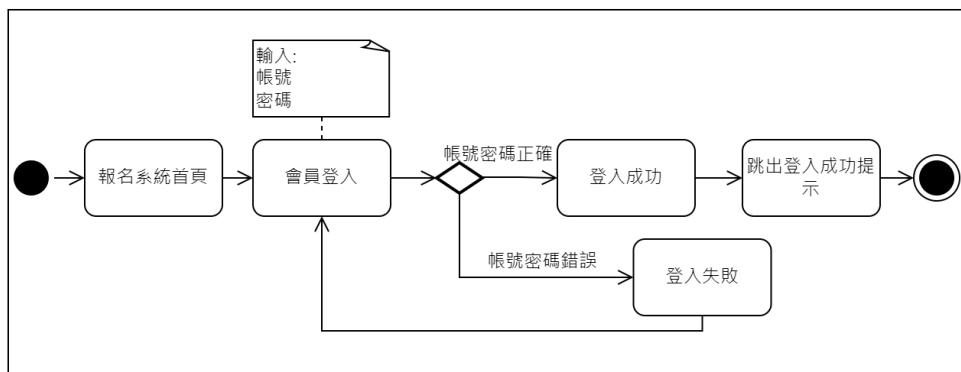


圖 2：使用案例 4.1 報名活動之活動圖

2.2.1.2 強韌圖

依據該使用案例之活動圖，可建立強韌圖以找出分析之類別，如下圖（圖 3）所示：

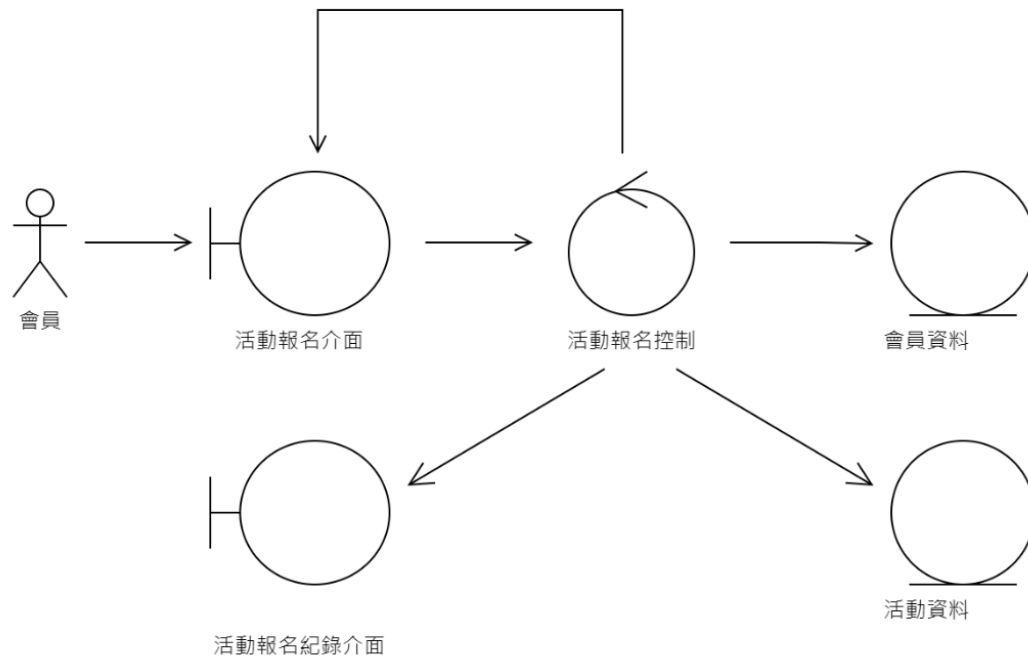


圖 3：使用案例 4.1 報名活動之強韌圖

2.3 使用案例 5.0 活動管理模組

本章節中，將針對活動管理模組進行系統動態分析圖之分析，活動管理模組一共包含四個功能：①新增活動、②檢視所有活動、③修改活動或活動場次資訊、④刪除活動，以下並根據此些使用案例進行必要圖形之分析。

表 2：商業流程編號 5.0 活動管理模組

模組	功能名稱	說明
5.0 模組： 活動管理	5.1 新增活動	管理員可以新增活動
	5.2 檢視所有活動	管理員可以檢視所有活動相關資訊
	5.3 修改活動或活動場次資訊	管理員可以修改活動名稱、活動時間、活動地點、活動場次、人數限制與圖片等資訊
	5.4 刪除活動	管理員可以刪除已結束或取消辦理的活動

2.3.1 使用案例 5.3 修改活動資訊

2.3.1.1 活動圖

○ 商業流程編號「5.3 修改活動資訊」之主要流程如下所示：

- ✓ 1. 在活動資料修改表單變更欲修改之項目。
- ✓ 2. 系統驗證資料格式。
- ✓ 3. 系統更新活動、場次資料。
- ✓ 4. 顯示更新成功提示。

- 透過以上主流程可將其轉換成活動圖，如下圖（圖 4）所示：

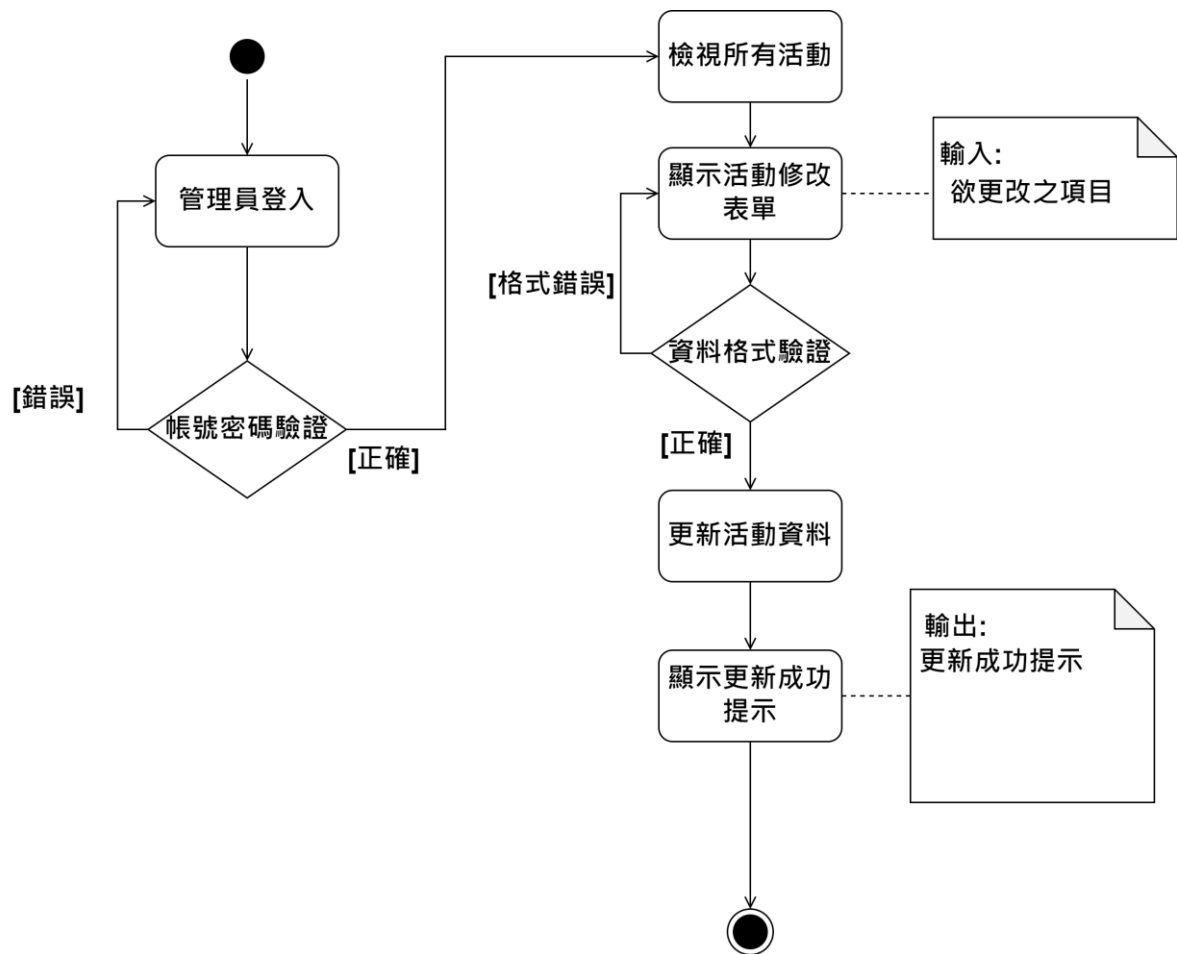


圖 4：使用案例 5.3 修改活動資訊之活動圖

2.3.1.2 強韌圖

依據該使用案例之活動圖，可建立強韌圖以找出分析之類別，如下圖（圖 5）所示：

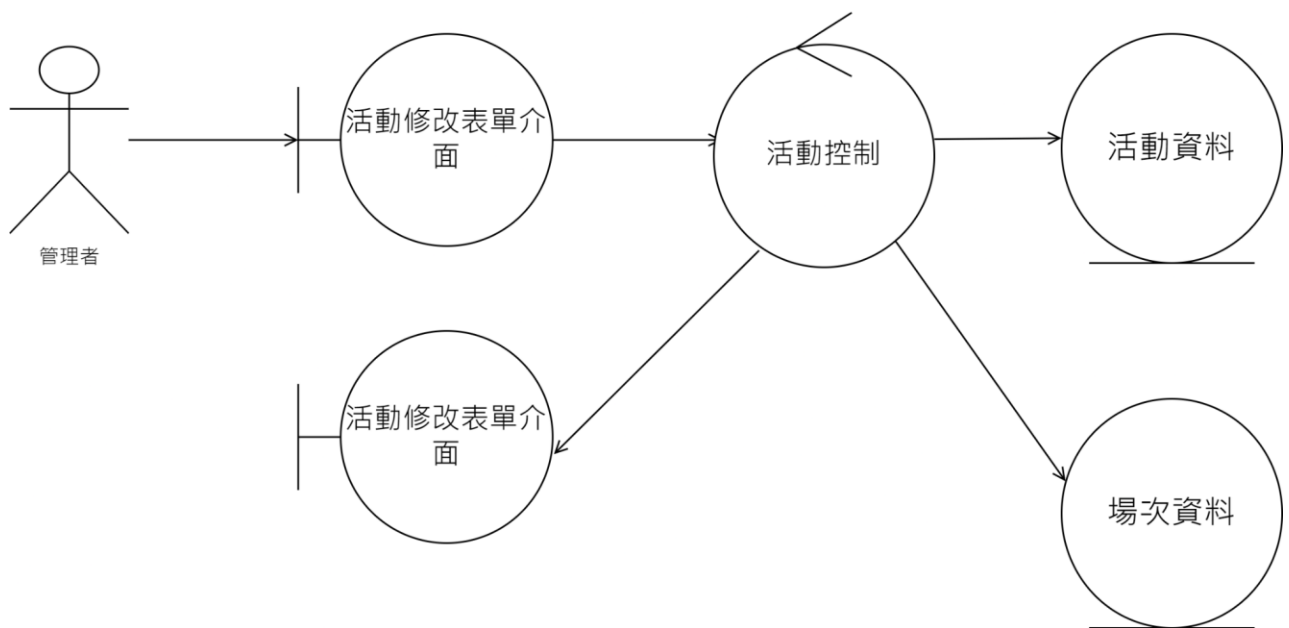


圖 5：使用案例 5.3 修改活動資訊之強韌圖

第 3 章 資料庫設計

本專案之活動報名系統提供使用者簡單與便利之活動報名、活動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活動維護等後台作業，希冀不僅能提供最新與詳盡之活動說明，更能提供良好之報名體驗。

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 6）之實體關係圖共計包含 7 個實體（Entity）、6 個關係（Relationship）、2 個複合性實體（Compound Entity）、1 個弱實體（Weak 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並記錄建立時間。
2. 會員可以異動會員資料，同時要記錄會員異動之時間。
3. 只有會員可以將活動加入收藏清單，如果一般會員要收藏活動要先登入或註冊。
4. 會員可以從收藏清單中選取欲報名之活動，進入報名頁面、選擇欲參加之場次進行報名，同時可以至報名紀錄查看報名之詳情。
5. 管理者可以管理與異動活動，並且進行維護作業，同時要記錄活動建立時間與異動之時間。
6. 管理者可以管理與異動會員資料，並且進行維護作業，同時要記錄會員異動之時間。

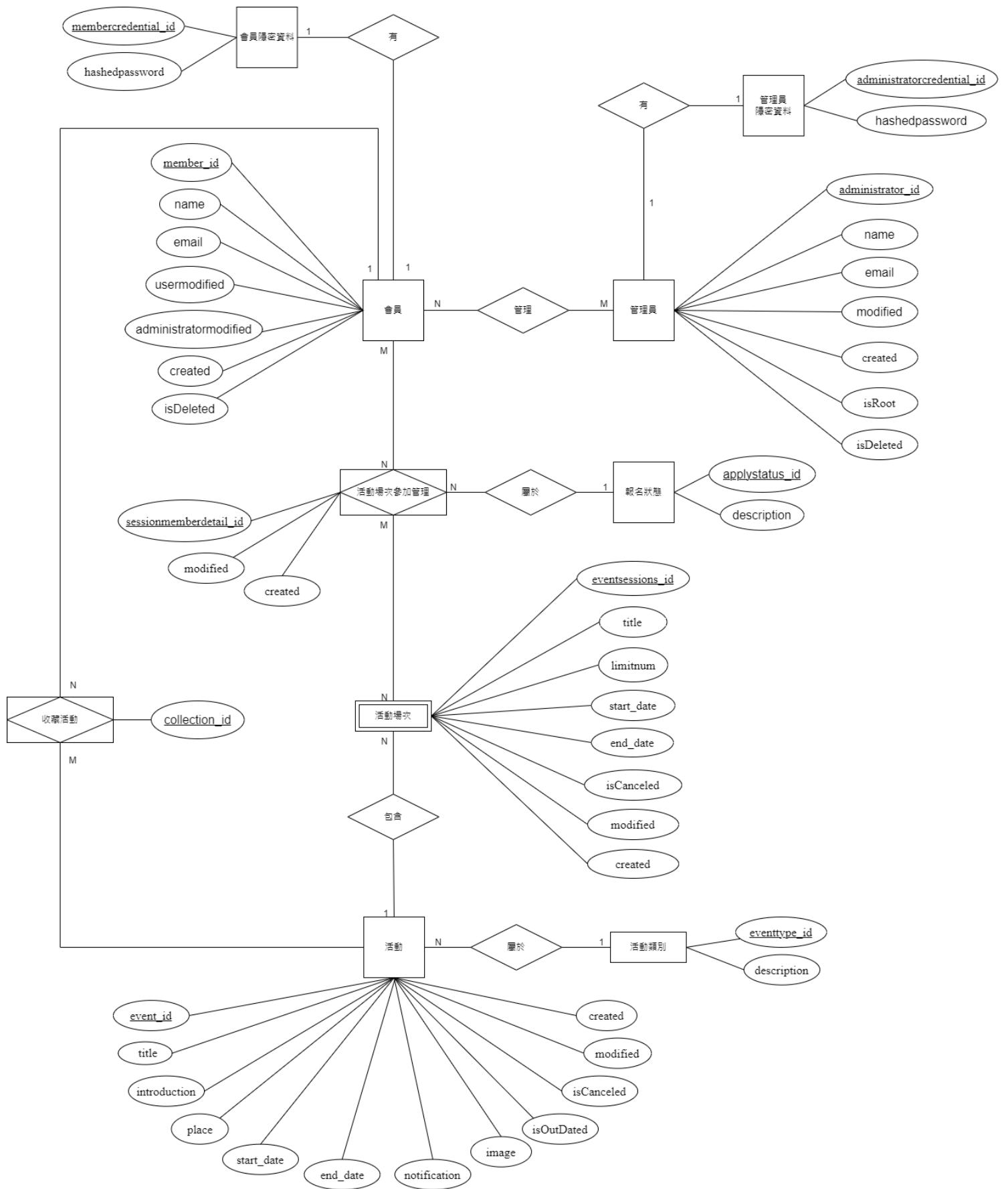


圖 6：實體關係圖

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 (TB_Member)

表 3：分析階段之會員資料表 (TB_Member) 資料結構

Key	名稱	類型
P. K.	member_id	Int
	name	Varchar
	email	Varchar
	usermodified	Datetime
	administratormodified	Datetime
	created	Datetime
	isDeleted	TinyInt

2. 會員隱密資料表 (TB_MemberCredential)

表 4：分析階段之會員隱密資料表 (TB_MemberCredential) 資料結構

Key	名稱	類型
P. K.	membercredential_id	Int
F. K.	member_id	Int
	hashedpassword	Varchar

3. 管理員資料表 (TB_Administrator)

表 5：分析階段之管理員資料表 (TB_Administrator) 資料結構

Key	名稱	類型
P. K.	administrator_id	Int
	name	Varchar
	email	Varchar
	modified	Datetime
	created	Datetime
	isRoot	TinyInt
	isDeleted	TinyInt

4. 管理員隱密資料表 (TB_AdministratorCredential)

表 6：分析階段之管理員隱密資料表 (TB_AdministratorCredential) 資料結構

Key	名稱	類型
P. K.	administratorcredential_id	Int
F. K.	administrator_id	Int
	hashedpassword	Varchar

5. 活動資料表 (TB_Event)

表 7：分析階段之活動資料表 (TB_Event) 資料結構

Key	名稱	類型
P. K.	event_id	Int
	title	Varchar
F. K.	eventtype_id	Int
	introduction	Varchar
	place	Varchar
	start_date	Datetime
	end_date	Datetime
	notification	Varchar
	image	Varchar
	isOutDated	TinyInt
	isCanceled	TinyInt
	modified	Datetime
	created	Datetime

6. 活動類別資料表 (TB_EventType)

表 8：分析階段之活動類別資料表 (TB_EventType) 資料結構

Key	名稱	類型
P. K.	eventtype_id	Int
	description	Varchar

7. 活動場次資料表 (TB_EventSessions)

表 9：分析階段之活動場次資料表 (TB_EventSessions) 資料結構

Key	名稱	類型
P. K.	eventsessions_id	Int
F. K.	event_id	Int
	title	Varchar
	limitnum	Int
	start_date	Datetime
	end_date	Datetime
	isCanceled	TinyInt
	modified	Datetime
	created	Datetime

8. 活動場次參加管理資料表 (TB_SessionMemberDetail)

表 10：分析階段之活動場次參加管理資料表 (TB_SessionMemberDetail) 資料結構

Key	名稱	類型
P. K.	sessionmemberdetail_id	Int
F. K.	member_id	Int
F. K.	eventsessions_id	Int
	modified	Datetime
	created	Datetime
F. K.	applystatus_id	Int

9. 活動報名狀態資料表 (TB_ApplyStatus)

表 11：分析階段之活動報名狀態資料表 (TB_ApplyStatus) 資料結構

Key	名稱	類型
P. K.	applysatus_id	Int
	description	Varchar

10. 收藏資料表 (TB_Collection)

表 12：分析階段之收藏資料表 (TB_Collection) 資料結構

Key	名稱	類型
P. K.	collection_id	Int
F. K.	event_id	Int
F. K.	member_id	Int

第 4 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 3 章 資料庫設計）以建立本專案之報名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。

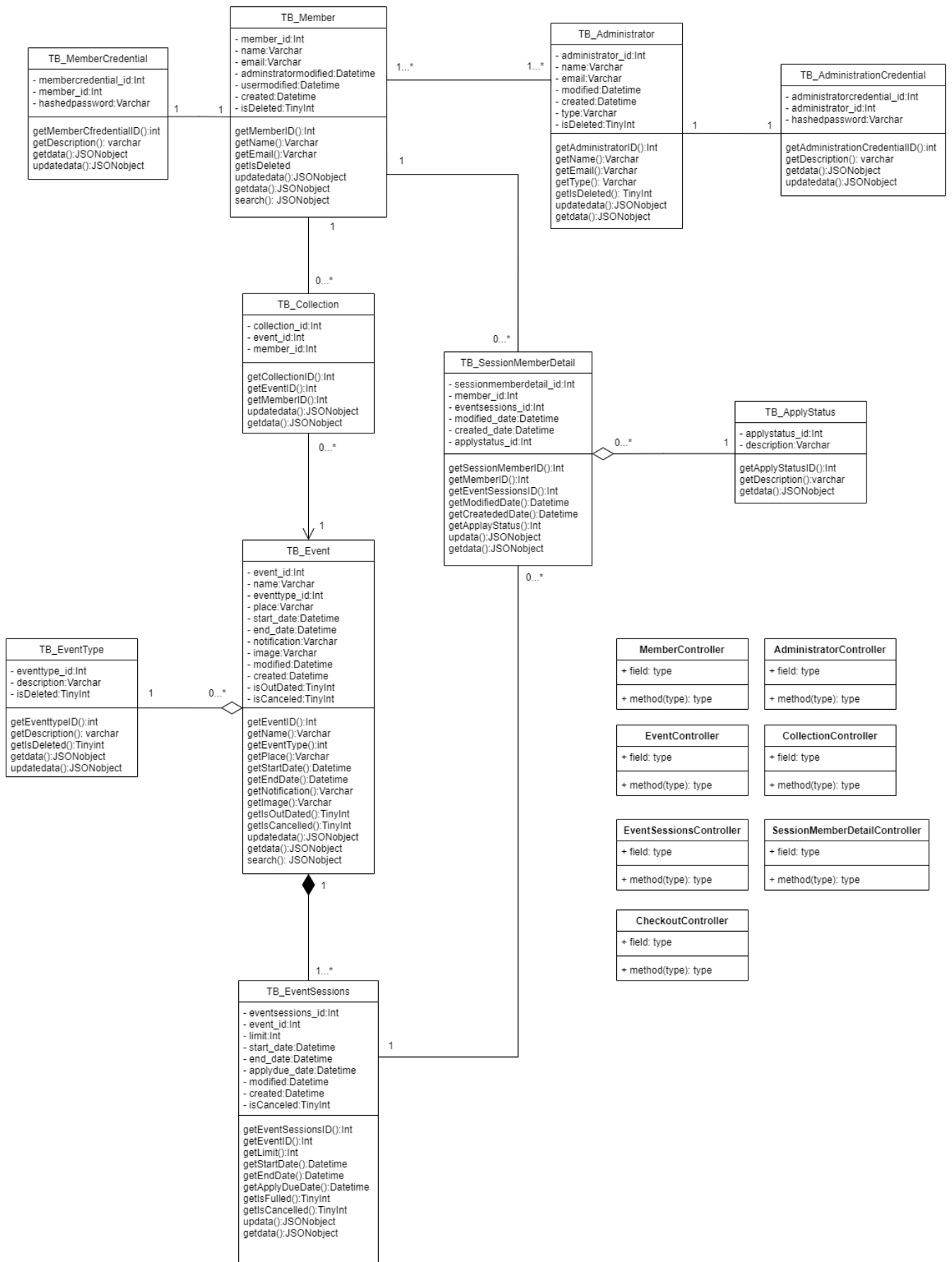


圖 7：分析階段之類別圖

第 5 章 系統開發環境(參考)

5.1 系統架構圖

本專案之整體架構如下圖（圖 8）所示，主要採用 Java 語言所撰寫之電子商務網站之應用程式，並預期採用 Java 平台技術之 Servlet 框架建構 Web 應用程式：

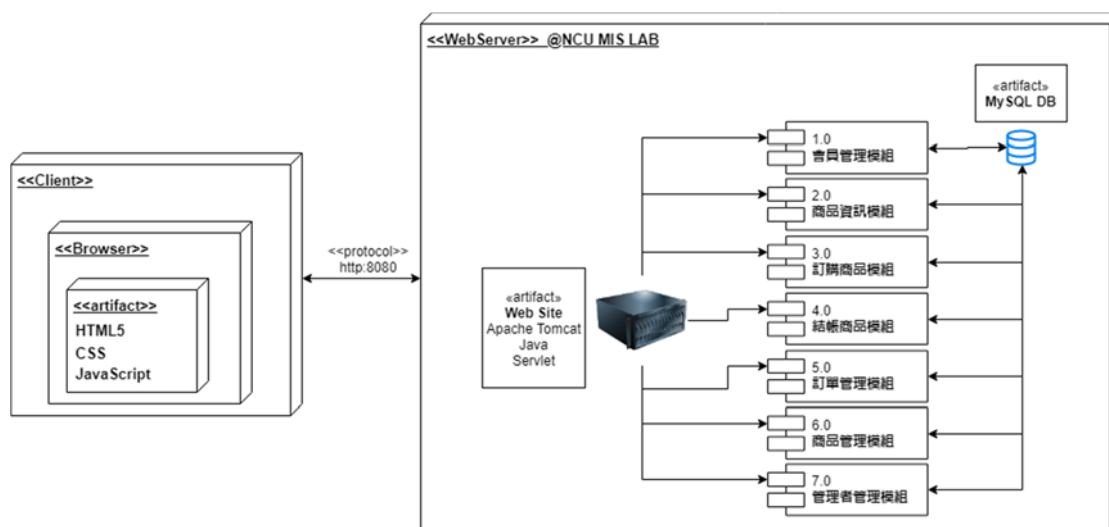


圖 8：分析階段之系統架構圖

1. 由於本專案之撰寫語言為 Java 因此需要採用 Apache Tomcat 作為伺服器軟體，預設 http 之埠號（port number）為 8080。
2. 資料庫採用關聯性資料庫 Oracle MySQL 進行使用，開發階段使用 community 版本即可。
3. 本專案依照 use case 共計有六個模組，每個模組在下份文件中必須進行細部之設計與說明。
4. 使用者之裝置僅須透過瀏覽器即可瀏覽本電子商務網站。

5.2 MVC 架構

本專案預期採用 MVC 架構，將應用程式的資料模型、使用介面和控制邏輯分割成 Model、View 和 Controller 三種元件，詳細如下圖（圖 13）所示。

其優點在於明確分割使用者介面和應用程式邏輯，將使用介面從商業邏輯中獨立出來，可以大幅降低建立大型應用程式時的複雜度，讓程式碼更有結構，容易測試、維護和重複使用。

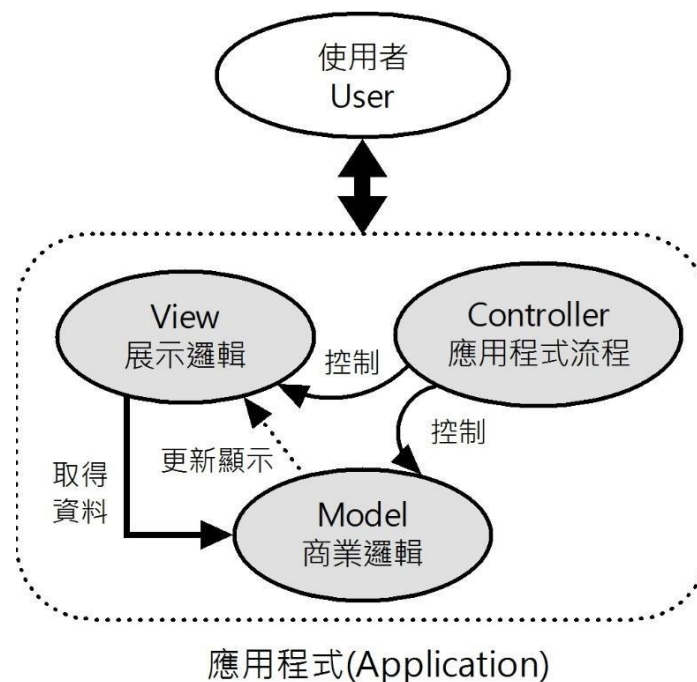


圖 9：MVC 架構

1. Model：

- ✓ 在 MVC 的 Model 元件是實作如何儲存應用程式的資料，包含資料和驗證規則。
- ✓ 以 Web 應用程式來說，Model 元件負責 Web 應用程式的資料存取和處理，即存取和處理儲存在資料庫、文字檔案或 XML 檔案的資料。

2. View：

- ✓ 實作展示邏輯（Presentation Logic）的物件。
- ✓ Web 應用程式是建立使用者在瀏覽器看到的 HTTP 回應訊息，通常就是 HTML 網頁。

- ✓ 使用 Model 物件儲存的資料來產生輸出結果，所以，View 元件可以透過 Model 元件取得資料庫的資料，然後將資料庫的資料轉換成有用的資訊來呈現給使用者檢視。

3. Controller：

- ✓ 整個應用程式的中心，連接 View 和 Model 元件來協調和控制應用程式的執行。
- ✓ Web 應用程式的 Controller 元件是控制資料處理流程的控制器，負責接收使用者從瀏覽器送出的 HTTP 請求，依請求執行所需操作，即下達指令給 Model 取出所需的資料，然後送至 View 元件來產生顯示結果的 HTML 網頁。