**國立勤益科技大學資訊管理系**

**109學年度四技學生實務專題報告書**

**整合租屋資訊與租屋管理網站開發與建置**

組別：第四組

專題生：徐千素 3A932003

賴妤典 3A932004

曾翊晴 3A932045

鄭羽庭 3A932051

黃佳怡 3A932117

指導老師：廖文忠 老師

中華民國 年 月 日

**摘要**

目錄

[第一章 緒論 4](#_Toc129011990)

[\*\*1-1專題背景與動機\*\* 4](#_Toc129011991)

[\*\*1-2 專題目的\*\* 5](#_Toc129011992)

[\*\*1-3 專題成效\*\* 7](#_Toc129011993)

[第二章 技術探討 8](#_Toc129011994)

[2-1 Laravel MVC架構與Routing 8](#_Toc129011995)

[2-2 物件導向系統分析與設計 9](#_Toc129011996)

[2-3 Laravel 寄信 10](#_Toc129011997)

[2-4 Laravel 檔案上傳與使用 12](#_Toc129011998)

[第三章 系統分析與設計 13](#_Toc129011999)

[3-1 需求描述 13](#_Toc129012000)

[3-2 事件分析 15](#_Toc129012001)

[3-3 系統使用案例分析 18](#_Toc129012002)

[3-4 實體關聯圖 21](#_Toc129012003)

[3-5 關聯式資料庫綱要圖 22](#_Toc129012004)

[3-6 資料表欄位設計 23](#_Toc129012005)

[3-7 訪客及會員相關使用案例分析與設計 28](#_Toc129012006)

[3-7-1 訪客/會員瀏覽平台首頁 28](#_Toc129012007)

[3-7-2 訪客/會員查詢出租房屋(地區) 30](#_Toc129012008)

[3-7-3 訪客/會員篩選租屋條件 32](#_Toc129012009)

[3-7-4 訪客/會員查看租屋資訊 34](#_Toc129012010)

[3-7-5 訪客訪客註冊會員 36](#_Toc129012011)

[3-7-6 訪客登入會員 38](#_Toc129012012)

[3-8 會員(房東/租客)相關使用案例分析與設計 40](#_Toc129012013)

[3-8-1 \*會員(房東/租客)顯示會員資料 40](#_Toc129012014)

[3-8-2 會員(房東/租客)修改會員資料 42](#_Toc129012015)

[3-8-3 會員(房東/租客)新增轉帳資訊 44](#_Toc129012016)

[3-8-4 會員(房東/租客)修改轉帳資訊 49](#_Toc129012017)

[3-8-5 會員(房東/租客)顯示租客頁面 51](#_Toc129012018)

[3-8-6 會員(房東/租客)顯示房東管理頁面 53](#_Toc129012019)

[3-8-7 會員登出 55](#_Toc129012020)

[3-9 會員(房東)相關使用案例分析與設計 56](#_Toc129012021)

[3-9-1 \*會員(房東)查看地點 56](#_Toc129012022)

[3-9-2 會員(房東)新增地點 58](#_Toc129012023)

[3-9-3 會員(房東)修改地點 60](#_Toc129012024)

[3-9-4 會員(房東)刪除地點 62](#_Toc129012025)

[3-9-5 \*會員(房東)查看房屋 64](#_Toc129012026)

[3-9-6 會員(房東)新增房屋 66](#_Toc129012027)

[3-9-7 會員(房東)修改房屋 68](#_Toc129012028)

[3-9-8 會員(房東)刪除房屋 70](#_Toc129012029)

[3-9-9 會員(房東)查看公告 72](#_Toc129012030)

[3-9-10 會員(房東)新增公告 73](#_Toc129012031)

[3-9-11 會員(房東)修改公告 75](#_Toc129012032)

[3-9-12 會員(房東)刪除公告 77](#_Toc129012033)

[3-9-13 會員(房東) 刊登房屋 79](#_Toc129012034)

[3-9-14 會員(房東)取消刊登房屋 81](#_Toc129012035)

[3-9-15 \*會員(房東)篩選某個狀態所有房屋資訊 83](#_Toc129012036)

[3-9-16 會員(房東)查看單一房屋資訊 85](#_Toc129012037)

[3-9-17 會員(房東)新增合約書 88](#_Toc129012038)

[3-9-18 會員(房東)刪除合約書 90](#_Toc129012039)

[3-9-19 會員(房東)查看合約書 92](#_Toc129012040)

[3-9-20 會員(房東)新增費用資訊 94](#_Toc129012041)

[3-9-21 會員(房東)修改費用資訊 95](#_Toc129012042)

[3-9-22 會員(房東)刪除費用資訊 95](#_Toc129012043)

[3-9-23 會員(房東)查看報修訊息 97](#_Toc129012044)

[3-9-24 會員(房東)查看詳細報修訊息 98](#_Toc129012045)

[3-9-25 會員(房東)更新報修狀態 99](#_Toc129012046)

[3-10 會員(租客)相關使用案例分析與設計 101](#_Toc129012047)

[3-10-1 會員(租客)查看公告 101](#_Toc129012048)

[3-10-2 會員(租客)查看合約 103](#_Toc129012049)

[3-10-3 會員(租客)查看費用資訊 104](#_Toc129012050)

[3-10-4 會員(租客)進行繳費 106](#_Toc129012051)

[3-10-5 會員(租客)查看報修訊息 108](#_Toc129012052)

[3-10-6 會員(租客)新增報修訊息 109](#_Toc129012053)

[3-10-7 會員(租客)修改報修訊息 111](#_Toc129012054)

[3-10-8 會員(租客)刪除報修訊息 113](#_Toc129012055)

[3-11 專案規劃 116](#_Toc129012056)

[3-12 系統運作環境 118](#_Toc129012057)

[3-13 系統開發工具 119](#_Toc129012058)

[第四章 進度與工作分配 120](#_Toc129012059)

[一、完成專題分配 120](#_Toc129012060)

[二、專題進度規劃 121](#_Toc129012061)

圖目錄

[圖 1 系統架構圖 9](#_Toc129180364)

[圖 2 MVC架構圖 11](#_Toc129180365)

[圖 3 房東新增房屋活動圖 12](#_Toc129180366)

[圖 4 Gmail應用程式密碼取得 13](#_Toc129180367)

[圖 5 與訪客/會員相關之使用案例圖 21](#_Toc129180368)

[圖 6 與會員(房東)相關之使用案例 22](#_Toc129180369)

[圖 7 與會員(租客)相關之使用案例 23](#_Toc129180370)

[圖 8 實體關聯圖 24](#_Toc129180371)

[圖 9 關聯式資料庫綱要圖 25](#_Toc129180372)

[圖 10 訪客/會員瀏覽平台首頁 31](#_Toc129180373)

[圖 11 訪客/會員查詢出租房屋 33](#_Toc129180374)

[圖 12 訪客/會員篩選租屋條件 34](#_Toc129180375)

[圖 13 訪客/會員查看租屋資訊 36](#_Toc129180376)

[圖 14 訪客註冊會員 38](#_Toc129180377)

[圖 15 訪客登入會員 40](#_Toc129180378)

[圖 16 會員(房東/租客)修改會員資料 42](#_Toc129180379)

[圖 17 會員(房東/租客)修改會員資料 44](#_Toc129180380)

[圖 18 會員(房東/租客)新增轉帳資訊 46](#_Toc129180381)

[圖 19 會員(房東/租客)修改轉帳資訊 48](#_Toc129180382)

[圖 20 會員(房東/租客)進入租客頁面 50](#_Toc129180383)

[圖 21 會員(房東/租客)進入房東管理頁面 52](#_Toc129180384)

[圖 22 會員(房東)新增地點 55](#_Toc129180385)

[圖 23 會員(房東)新增地點 57](#_Toc129180386)

[圖 24 會員(房東)修改地點 59](#_Toc129180387)

[圖 25 會員(房東)刪除地點 61](#_Toc129180388)

[圖 26 會員(房東)新增房屋 63](#_Toc129180389)

[圖 27 會員(房東)新增房屋 65](#_Toc129180390)

[圖 28 會員(房東)修改房屋 67](#_Toc129180391)

[圖 29 會員(房東)刪除房屋 69](#_Toc129180392)

[圖 30 會員(房東)查看公告 71](#_Toc129180393)

[圖 31 會員(房東)新增公告 72](#_Toc129180394)

[圖 32 會員(房東)修改公告 74](#_Toc129180395)

[圖 33 會員(房東)刪除公告 76](#_Toc129180396)

[圖 34 會員(房東)刊登待出租房屋 78](#_Toc129180397)

[圖 35 會員(房東)取消刊登房屋 80](#_Toc129180398)

[圖 36 會員(房東)查看某狀態房屋資訊 82](#_Toc129180399)

[圖 37 會員(房東)查看刊登中房屋資訊 84](#_Toc129180400)

[圖 38 會員(房東)新增合約書 87](#_Toc129180401)

[圖 39 會員(房東)刪除合約書 89](#_Toc129180402)

[圖 40 會員(房東)查看合約書 91](#_Toc129180403)

[圖 41 會員(房東)新增費用資訊 93](#_Toc129180404)

[圖 41 會員(房東)新增費用資訊 95](#_Toc129180405)

[圖 42 會員(房東)修改費用資訊 97](#_Toc129180406)

[圖 43 會員(房東)刪除費用資訊 99](#_Toc129180407)

[圖 44 會員(房東)查看報修訊息 101](#_Toc129180408)

[圖 45 會員(房東)查看詳細報修訊息 102](#_Toc129180409)

[圖 46 會員(房東)更新報修狀態 103](#_Toc129180410)

[圖 47 會員(租客)查看公告 105](#_Toc129180411)

[圖 48 會員(租客)查看合約 107](#_Toc129180412)

[圖 49 會員(租客)查看費用資訊 108](#_Toc129180413)

[圖 50 會員(租客)進行繳費 110](#_Toc129180414)

[圖 51 會員(租客)查看報修訊息 112](#_Toc129180415)

[圖 52 會員(租客)新增報修訊息 113](#_Toc129180416)

[圖 53 會員(租客)修改報修訊息 115](#_Toc129180417)

[圖 54 會員(租客)刪除報修訊息 117](#_Toc129180418)

表目錄

[表 1 訪客事件表 18](#_Toc129180343)

[表 2 會員(房東、租客)事件表 18](#_Toc129180344)

[表 3 會員(房東)事件表 18](#_Toc129180345)

[表 4 會員(租客)事件表 20](#_Toc129180346)

[表 5 使用者資料表 26](#_Toc129180347)

[表 6 房東資料表 26](#_Toc129180348)

[表 7租客資料表 26](#_Toc129180349)

[表 8 系統管理員資料表 26](#_Toc129180350)

[表 9 系統公告資料表 27](#_Toc129180351)

[表 10 公告資料表 27](#_Toc129180352)

[表 11 合約紀錄資料表 27](#_Toc129180353)

[表 12 目前租客資料表 28](#_Toc129180354)

[表 13 房屋 28](#_Toc129180355)

[表 14 地點資料表 28](#_Toc129180356)

[表 15 房屋圖片資料表 29](#_Toc129180357)

[表 16 房屋特色資料表 29](#_Toc129180358)

[表 17 房屋設備資料表 29](#_Toc129180359)

[表 18 報修資料表 29](#_Toc129180360)

[表 19 報修回報資料表 30](#_Toc129180361)

[表 20 應繳款項資料表 30](#_Toc129180362)

[表 21 租屋費用資料表 30](#_Toc129180363)

# 緒論

## \*\*1-1專題背景與動機\*\*

(主題:租屋 出租房屋 房東要出租房屋 要管理房屋，背景就是談你的主題)

現代人經濟壓力大，其中一原因就是房價不斷上漲，買房可能要背負幾十年的貸款壓力，要有一間屬於自己的房產著實不易，因此為了提升生活品質，不少人將”住”的價值轉移在租屋上。

當願意租屋的人變多，就增加了房東們的收入機會，但租屋族這麼多，房東要如何將要出租的房間曝光？又該如何管理自己出租的房屋和租客？目前市面上的租屋資訊及租屋管理都是分開狀態，房東刊登房屋和管理租客需使用不同的網站或app，不同網站就需要不同帳號，以及不同的使用方法，房東在使用上較為不便。

於是本專題旨在增加房東管理出租房屋以及刊登房屋的便利性，開發與建置整合租屋資訊與租屋管理網站，使房東在管理和刊登出租房屋時不必兩方操作，如房東在租屋資訊網站刊登房屋，收到租客後又得換另一個平台管理租客；並提供租客能更快的找到適合自己需求的房屋，以及租屋期間所需的回報功能。

目前市面上的租屋資訊及租屋管理都是分開狀態，租客找房只能在特定網站或社群平台上，而要使用租屋管理系統則需使用不同網站或app，在使用上較為不便，且在租屋資訊網站租出後須在網站上修改狀態或取消刊登，又須在租屋管理系統新增已租出之房屋，或租約到期後又須前往租屋資訊網站進行刊登，過程較為繁雜，於是本專題旨在提供房東管理出租房屋的便利性，開發與建置整合租屋資訊與租屋管理網站，使房東在管理出租房屋時不必兩方操作。

## \*\*1-2 專題目的\*\*

本專題目的為規劃開發一個整合租屋資訊與租屋管理的平台，在該平台房東可以進行刊登，上傳電子合約，並進行租屋管理，可設定租金、水電費、管理費，並線上收取費用，且可收到租客維修通知，設定公告、提醒租客注意事項。而租客能夠在平台上尋找租屋資訊，查看電子合約，進行線上支付費用、維修回報、查看公告。

從租屋資訊到租屋管理皆可在本平台完成，使房東不必跨平台使用，省去繁雜手續。

本專題網站之系統架構圖，如圖1所示，系統提供房東從刊登到租屋管理、租客從找房到簽約後繳費、報修進行整合。

系統主要功能包括：

會員(房東/租客)主要功能

* 查找房屋資訊
* 選擇房東/租客頁面
* 進行收付款

房東主要功能

* 可新增地點並決定是否刊登
* 管理房屋資訊，包括費用、維修、合約等

租客主要功能

* 回報房屋應維修區域
* 選取費用並進行繳費

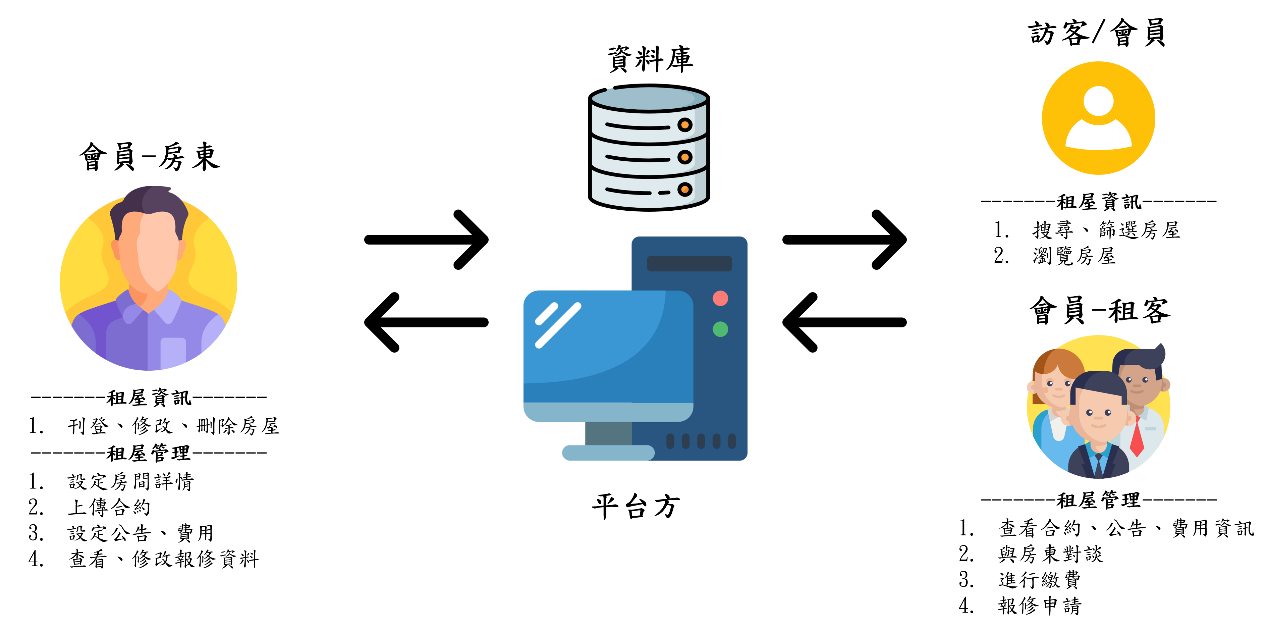


圖 1 系統架構圖

## \*\*1-3 專題成效\*\*

藉由此系統的建立與運作，使用本系統預期可達到下列成效：

1. **不須另外註冊帳號即可使用租屋資訊及租屋管理功能：**

透過平台的整合，房東只需使用一個帳號，便可使用到刊登租屋資訊，以及管理租客的功能。

1. **減去房東管理時間：**

合約到期時、應繳款時自動通知房東和租客，不怕有任一方遺忘，房東也無須撥空時間一個一個提醒。

# 技術探討

## 2-1 Laravel MVC架構與Routing

MVC（Model–View–Controller）是軟體工程中的一種軟體設計的架構也被Laravel框架所採用，MVC架構把軟體分為三個基本部分：模型（Model）、視圖（View）和控制器（Controller）[2]，如圖2所示。

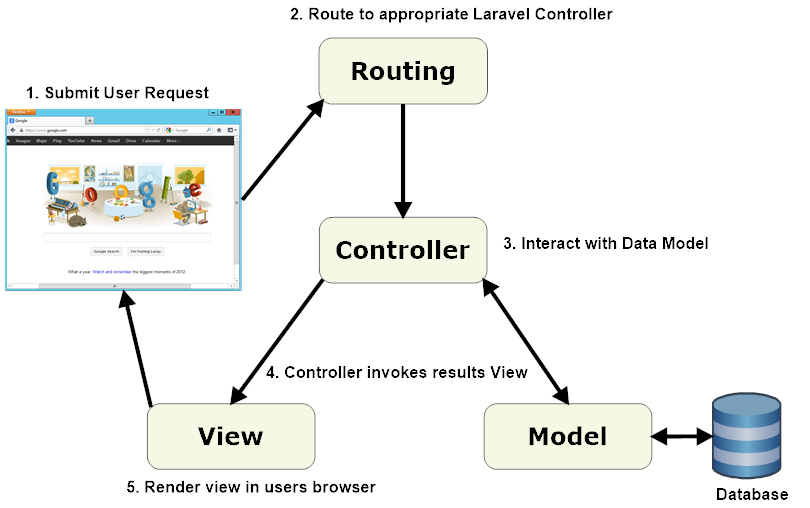


圖 2 MVC架構圖

當使用者輸入送出HTTP請求時，請求送到HTTP伺服器後，如上圖2，Laravel的路由機制(Routing)會依照程式設計者Routing的設計，決定由那一個控制器(Controller)的方法(method)來處理使用者送出的HTTP請求；若處理過程當中需存取資料庫，此時Controller可調用適當的Model進行資料庫存取；Controller處理請求完畢後，若需將結果以網頁呈現，此時Controller可將網頁上需要呈現的資料，傳遞給View，由View產生真正網頁內容後，呈現給使用者，如上圖2所示。

## 2-2 物件導向系統分析與設計

本專題使用物件導向方式進行系統分析與設計，物件導向系統分析是由了解使用者需求開始，由使用者需求分析出所有可能發生的系統事件，例如：「房東新增房屋」、「租客進行報修」；一個事件代表系統當中的一個使用案例，使用案例通常包含某一種使用者的參與，例如：「房東」會參與「新增房屋」的使用案例；而「租客」會參與「進行報修」的使用案例，理由同上述。確認系統的使用案例後，就開始分析每一個使用案例的情節，描述使用者與系統在使用案例當中如何互動，使用案例的情節也可以活動圖或系統循序圖呈現，如圖3。

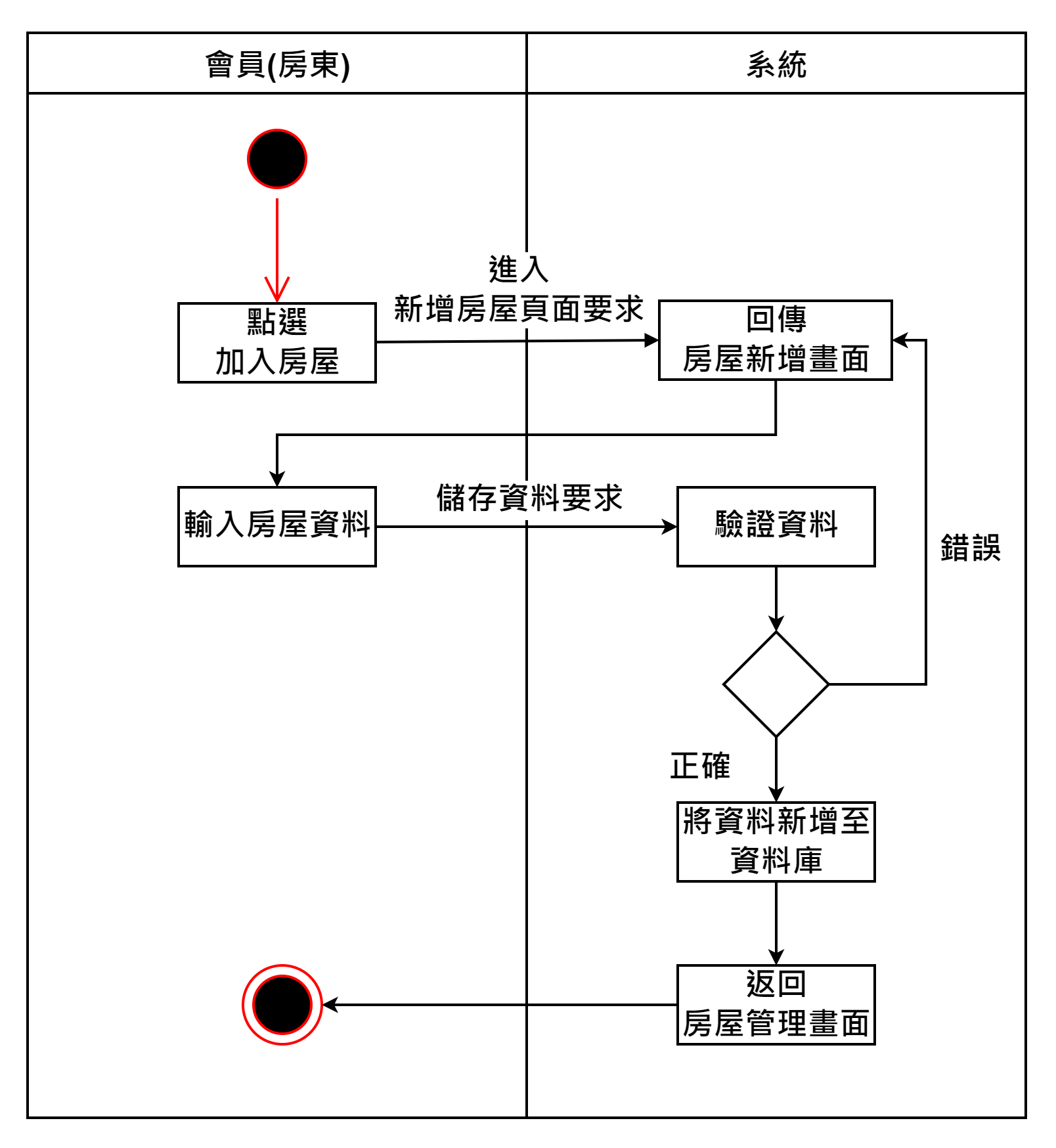


圖 3 房東新增房屋活動圖

## 2-3 Laravel 寄信

若要實現Laravel寄信功能，並利用Gmail進行寄送信件，首先必須完成.env檔的MAIL設定，類似如下：，

* MAIL\_MAILER=smtp
* MAIL\_HOST=smtp.gmail.com
* MAIL\_PORT=587
* MAIL\_USERNAME=Gmail帳號
* MAIL\_PASSWORD=Gmail帳號的密碼或應用程式密碼
* MAIL\_ENCRYPTION=tls
* MAIL\_FROM\_ADDRESS=null或Gmail的帳號address
* MAIL\_FROM\_NAME="${APP\_NAME}"

上述設定也可以直接設定在config/mail.php當中，要注意的是若使用Gmail進行操作，MAIL\_PASSWORD最好填入該Google帳號的應用程式密碼，取得方式為進入Google帳戶管理，在安全性中選擇應用程式密碼，如下圖所示，並選擇郵件和Windows電腦，按下產生按鈕即可取得應用程式密碼，置入.env檔案當中。

一張含有 文字 的圖片

自動產生的描述

圖 4 Gmail應用程式密碼取得

應用程式密碼指的是一組 16 位數密碼，可讓低安全性應用程式或裝置存取Google 帳戶，且自2022年5月30日起，如果第三方應用程式或裝置只要求您以使用者名稱和密碼登入 Google 帳戶，Google 將不再支援這些應用程式或裝置，因此當您要登入不支援雙步驟驗證的應用程式和裝置時，必須使用應用程式密碼來代替一般密碼來使用。

當.env檔設定完畢後，即可自訂email的view，並利用Mail::send()、Mail::queue()或Mail::later()方法來寄送信件[4] ，簡單示範如下；

* **電子郵件內容View設計**(假設View檔名為post.blade.php，且在emails資料夾當中)：

<body>

<h3>Hello, {{$name}}</h3>

<h4>提醒:</h4>

<p>{ $content }</p>

</body>

* **寄送電子郵件的方法**：假設程式碼放置於某一控制器(MailController)當中的send()方法當中，如下：

send(){

$recipient\_name='xxxx';

$recipient\_address='xxxx@gmail.com ';

$content='Welcome to Laravel 8. <br>This is a mail testing.<br>\*\*\*\*\*\*';

$msg = ['name' => $recipient\_name, 'content'=> $content ];

$to = ['name' => $recipient\_name, 'address' => $recipient\_address];

$from = ['name' => 'yyyy', 'address'=>'yyyy@gmail.com'];

$subject = 'Laravel Mail Testing';

Mail::send('emails.post', $msg, function($message) use($to, $from, $subject) {

$message->subject($subject);

$message->to($to['address'], $to['name']);

$message->from($from['address'], $from['name']); });

}

}

* 電子郵件寄送(測試)路由：Route::get('/sendmail', [MailController::class, 'send']);

Laravel寄送電子郵件主要使用Mail::send方法，也可以改用Mail::queue或Mail::later，Mail::send方法第一個參數是電子郵件的 View，第二個參數$msg是要傳入View當中的資料，第三個參數是callback函數，需在$message->subject()方法填入信件主旨，$message->to()方法填入收信對象，$message->from()填入寄信者，最後可透過寄送路由測試寄出該信件。

## 2-4 Laravel 檔案上傳與使用

因本專題是整合租屋資訊與租屋管理網站，房東必須將租約上傳至平台，租客在檢視合約時，合約才能呈現出來；本節將以此例，介紹Laravel如何讓租約檔案由賣家的電腦上傳及儲存至平台

# 系統分析與設計

## 3-1 需求描述

依照使用者的不同，分別找出其在系統上所需要的功能需求，加以描述。

1. 訪客/會員需求描述
2. 可以瀏覽平台的首頁。
3. 可以查詢出租房屋(關鍵字)。
4. 可以篩選出租房屋(特色、設備、房型、坪數、樓層、租金間距等)
5. 可以查看租屋資訊，包含房屋名稱、房屋圖片、介紹、房型、坪數、樓層、附家具、每月租金、水電費、房東聯絡方式、租屋限制。
6. 可以註冊成為會員。
7. 可以利用註冊的帳號密碼登入。
8. 會員基本功能需求描述
9. 可以在個人頁面修改會員基本資料。
10. 可以設定轉帳資訊
11. 可以修改轉帳資訊。
12. 可以瀏覽租客身份頁面。
13. 可以瀏覽房東身份頁面。
14. 可以登出會員。
15. 租屋資訊與租屋管理需求描述

**會員(房東)需求描述**

* 租屋資訊

1. 可刊登、取消刊登租屋資訊。

* 租屋管理

1. 可以新增、修改、刪除地點。
2. 可以新增、修改、刪除房屋。
3. 可以新增、修改、刪除公告。
4. 可檢查看某狀態(閒置、出租中、刊登中)下所有房屋資訊
5. 可以查看單一房屋資訊
6. 可以新增、修改、刪除費用資訊
7. 可以檢視所有費用收入
8. 可以查看報修訊息
9. 可以更新報修狀態
10. 可以新增、刪除、查看合約書

**會員(租客)需求描述**

* 租屋管理

1. 可以查看公告。
2. 可以查看合約。
3. 可以查看費用資訊(水費、電費、保證金、租金、管理費等)。
4. 可以進行繳費
5. 可以新增、修改、刪除報修訊息。

## 3-2 事件分析

由以上需求描述，可歸納出以下不同使用者所產生的51個事件，依不同使用者將事件分析結果分別列於表。

表 1 訪客事件表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 事件 | 觸發器 | 來源 | 活動 | 回應 | 目的地 |
| 瀏覽平台首頁 | 首頁 | 訪客 | 瀏覽平台首頁 | 平台 | 訪客 |
| 查詢出租房屋 | 查詢房屋 | 訪客 | 查詢出租房屋 | 查詢後租屋資訊頁面 | 搜尋結果 |
| 篩選租屋條件 | 查詢房屋 | 訪客 | 篩選出租房屋 | 篩選後租屋資訊頁面 | 搜尋結果 |
| 查看租屋資訊 | 租屋資訊 | 訪客 | 瀏覽租屋資訊 | 租屋資訊頁面 | 租屋資訊頁面 |
| 註冊會員 | 註冊會員 | 訪客 | 註冊會員 | 註冊畫面 | 訪客 |
| 登入會員 | 登入會員 | 訪客 | 確認登入資料 | 平台 | 會員中心 |

表 2 會員(房東、租客)事件表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 事件 | 觸發器 | 來源 | 活動 | 回應 | 目的地 |
| 修改會員資料 | 修改會員資料 | 會員 | 修改個人資料 | 成功畫面 | 會員中心 |
| 新增轉帳資訊 | 新增轉帳資訊 | 會員 | 新增轉帳銀行帳號 | 成功畫面 | 會員中心 |
| 修改轉帳資訊 | 新增轉帳資訊 | 會員 | 新增轉帳銀行帳號 | 成功畫面 | 會員中心 |
| 顯示租客頁面 | 租客頁面 | 會員 | 進入租客頁面 | 租屋頁面 | 會員 |
| 顯示房東管理頁面 | 房東管理 | 會員 | 進入房東頁面 | 房東管理頁面 | 房東管理頁面 |
| 會員登出 | 會員登出 | 會員 | 登出 | 成功畫面 | 首頁 |

表 3 會員(房東)事件表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 事件 | 觸發器 | 來源 | 活動 | 回應 | 目的地 |
| 新增地點 | 新增地點 | 會員(房東) | 新增地點資料 | 成功畫面 | 房屋管理頁面 |
| 修改地點 | 修改地點 | 會員(房東) | 修改地點資料 | 成功畫面 | 房屋管理頁面 |
| 刪除地點 | 刪除地點 | 會員(房東) | 刪除地點資料 | 成功畫面 | 房屋管理頁面 |
| 新增房屋 | 新增按鈕 | 會員(房東) | 新增房屋 | 成功畫面 | 房屋管理頁面 |
| 修改房屋 | 修改按紐 | 會員(房東) | 修改房屋 | 成功畫面 | 房屋管理頁面 |
| 刪除房屋 | 刪除按鈕 | 會員(房東) | 刪除房屋 | 成功畫面 | 房屋管理頁面 |
| 查看公告 | 公告按鈕 | 會員(房東) | 公告按鈕 | 成功畫面 | 房屋公告頁面 |
| 新增公告 | 新增公告 | 會員(房東) | 新增公告 | 成功畫面 | 房屋公告頁面 |
| 修改公告 | 修改公告 | 會員(房東) | 修改公告 | 成功畫面 | 房屋公告頁面 |
| 刪除公告 | 刪除公告 | 會員(房東) | 刪除公告 | 成功畫面 | 房屋公告頁面 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 事件名稱 | 觸發器 | 來源 | 活動 | 回應 | 目的地 |
| 刊登房屋 | 刊登按鈕 | 會員(房東) | 更改房屋狀態 | 成功畫面 | 刊登管理頁面 |
| 取消刊登 | 取消刊登按鈕 | 會員(房東) | 更改房屋狀態 | 成功畫面 | 刊登管理頁面 |
| 篩選某個狀態所有房屋資訊 | 狀態按鈕 | 會員(房東) | 顯示該狀態的所有房屋 | 成功畫面 | 房屋管理頁面 |
| 查看單一房屋資訊 | 房屋按鈕 | 會員(房東) | 顯示房屋資訊 | 成功畫面 | 房屋資訊頁面 |
| 新增合約書 | 新增按鈕 | 會員(房東) | 新增合約書 | 成功畫面 | 房屋管理頁面 |
| 刪除合約書 | 刪除按鈕 | 會員(房東) | 刪除合約書 | 成功畫面 | 房屋管理頁面 |
| 查看合約書 | 合約按鈕 | 會員(房東) | 查看合約書 | 成功畫面 | 房屋管理頁面 |
| 新增費用資訊 | 新增費用 | 會員(房東) | 新增費用 | 成功畫面 | 費用資訊頁面 |
| 修改費用資訊 | 修改費用 | 會員(房東) | 修改費用 | 成功畫面 | 費用資訊頁面 |
| 刪除費用資訊 | 刪除費用 | 會員(房東) | 刪除費用 | 成功畫面 | 費用資訊頁面 |
| 查看報修訊息 | 報修 | 會員(房東) | 查看報修 | 成功畫面 | 報修訊息畫面 |
| 查看詳細報修訊息 | 報修訊息 | 會員(房東) | 查看詳細報修訊息 | 成功畫面 | 詳細報修訊息畫面 |
| 更新報修狀態 | 更新狀態 | 會員(房東) | 更新狀態 | 成功畫面 | 報修訊息畫面 |

表 4 會員(租客)事件表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 事件名稱 | 觸發器 | 來源 | 活動 | 回應 | 目的地 |
| 查看公告 | 公告按鈕 | 會員(租客) | 查看公告 | 公告畫面 | 公告頁面 |
| 查看信件 | 信件按鈕 | 會員(租客) | 查看信件 | 信件畫面 | 信件頁面 |
| 查看合約 | 合約按鈕 | 會員(租客) | 查看合約 | 合約畫面 | 合約頁面 |
| 查看費用資訊 | 查看費用 | 會員(租客) | 查看費用 | 費用總覽畫面 | 費用資訊頁面 |
| 進行繳費 | 繳費按鈕 | 會員(租客) | 繳費 | 成功畫面 | 費用總覽頁面 |
| 查看報修訊息 | 報修按鈕 | 會員(租客) | 查看報修訊息 | 成功畫面 | 報修訊息頁面 |
| 新增報修訊息 | 新增報修 | 會員(租客) | 新增報修訊息 | 成功畫面 | 報修訊息頁面 |
| 修改報修訊息 | 報修按鈕 | 會員(租客) | 修改報修訊息 | 成功畫面 | 報修訊息頁面 |
| 刪除報修訊息 | 報修按鈕 | 會員(租客) | 刪除報修訊息 | 成功畫面 | 報修訊息頁面 |

## 3-3 系統使用案例分析

圖5到圖7分別為本系統依照訪客/會員、會員(房東) 、會員(租客)三種不同使用者的事件分析結果以及功能需求所繪製之使用案例圖。

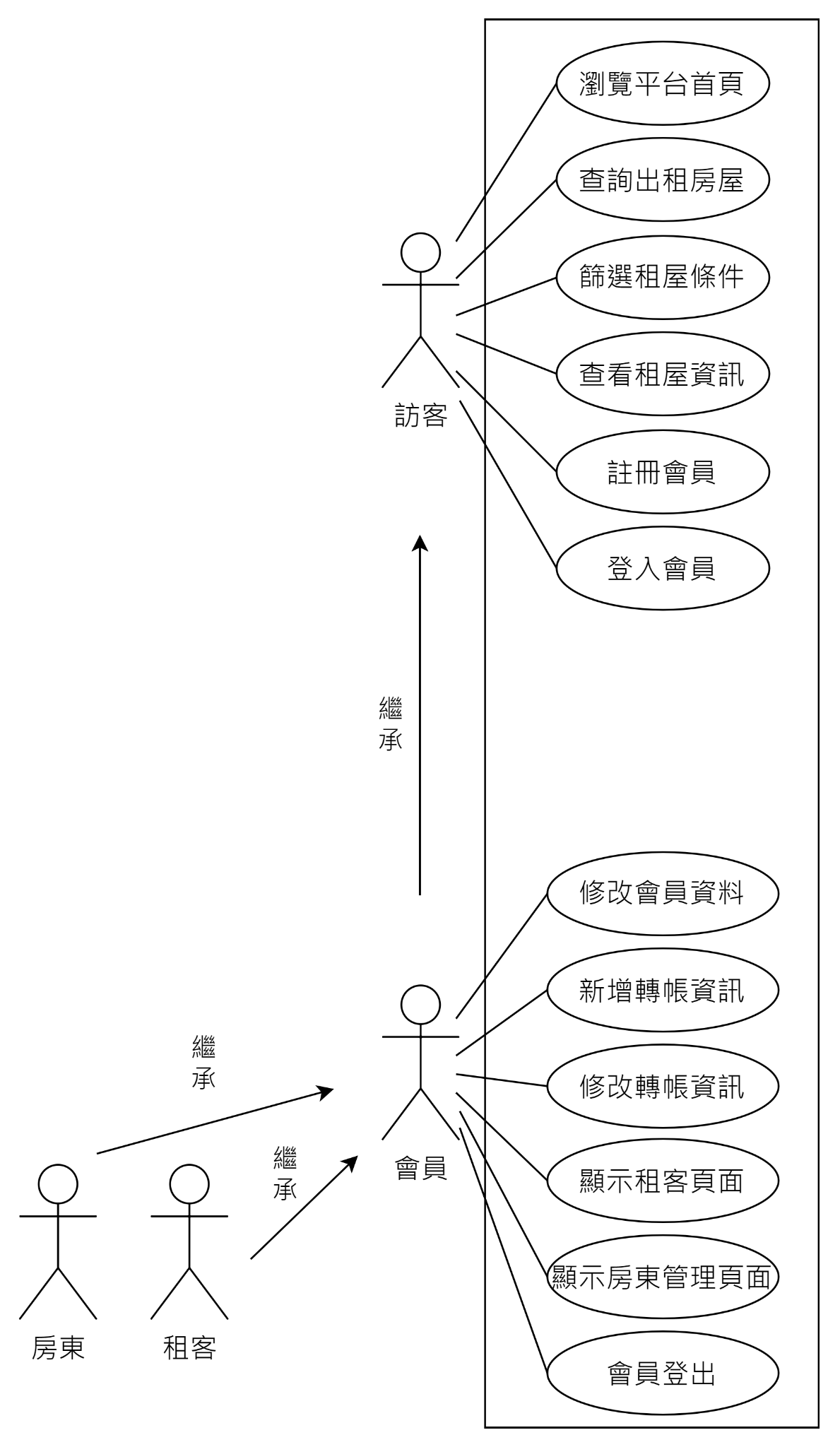


圖 5 與訪客/會員相關之使用案例圖

一張含有 文字, 金屬器皿, 圈狀彈簧, 樹葉 的圖片

自動產生的描述

圖 6 與會員(房東)相關之使用案例

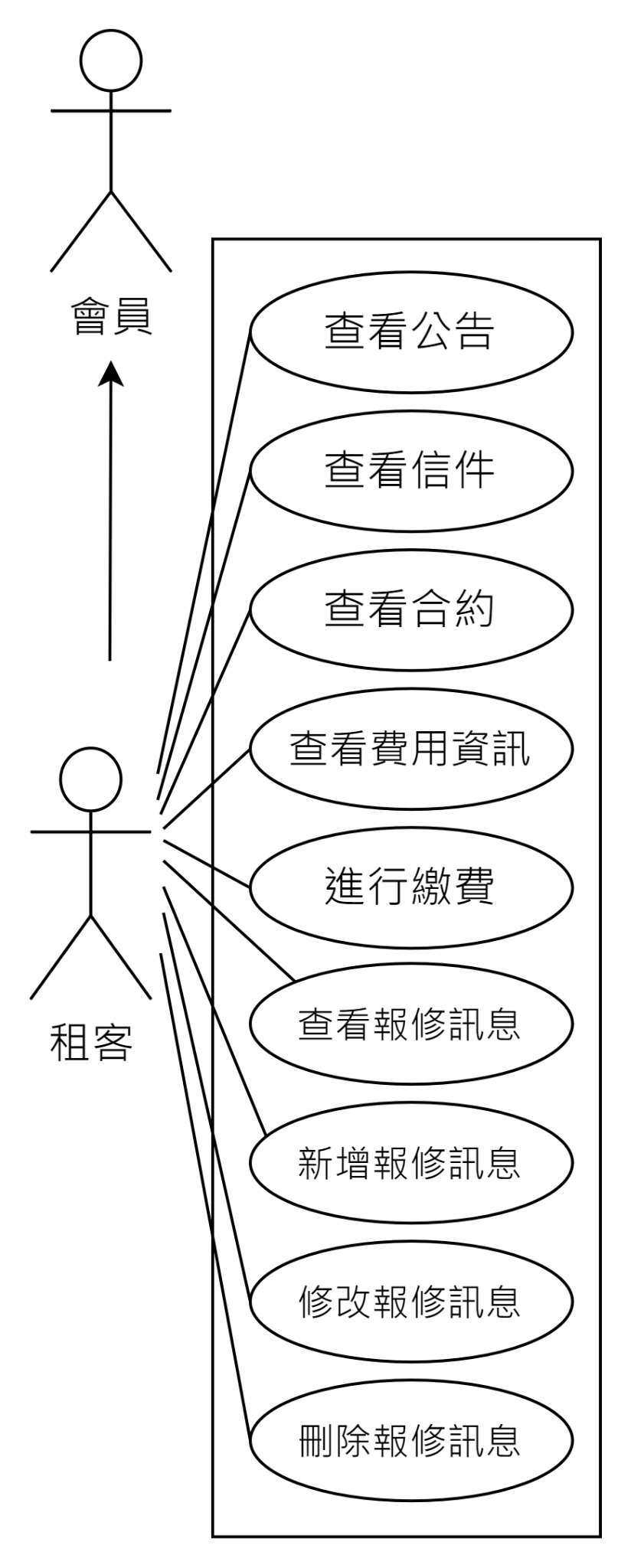
****

圖 7 與會員(租客)相關之使用案例

## 3-4 實體關聯圖

根據需求所做分析可得圖8之實體關聯圖，從這張圖可以看出與本系統相關的資料架構。

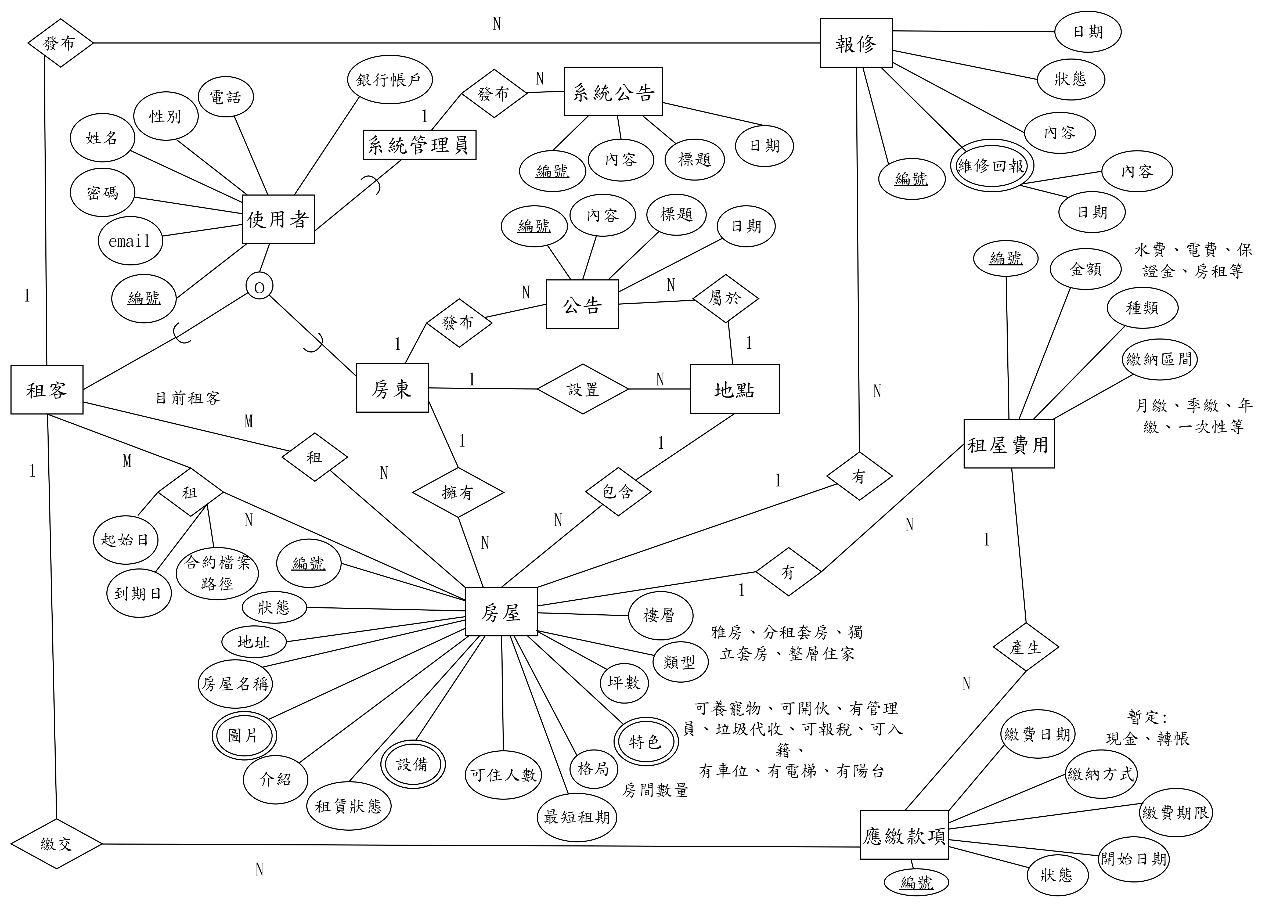


圖 8 實體關聯圖

## 3-5 關聯式資料庫綱要圖

圖9為系統之關聯式資料庫綱要圖，依據圖8實體關係圖轉換而來，可看出系統中將有17個資料表，以及每個資料表的主鍵、外來鍵和相關欄位。

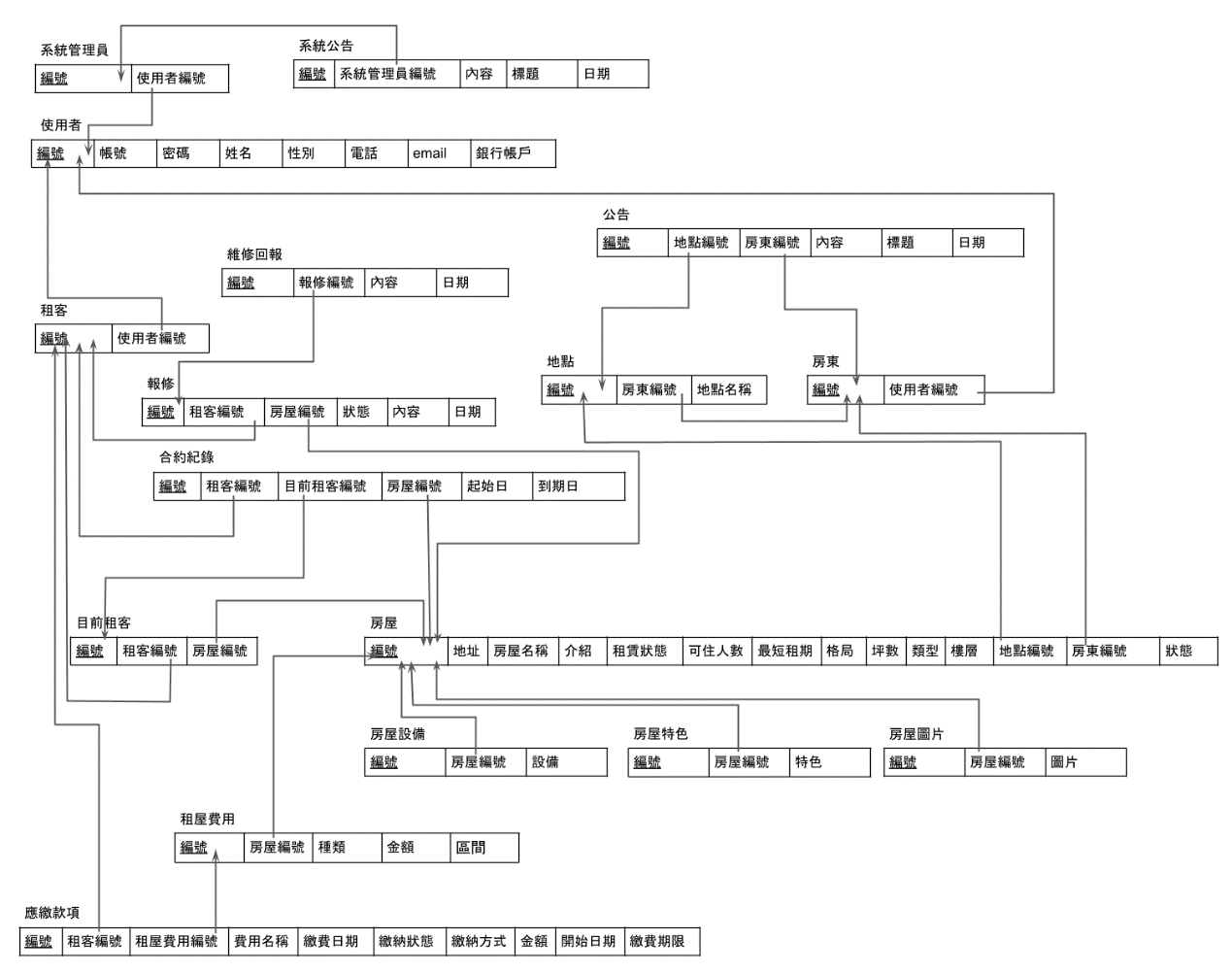


圖 9 關聯式資料庫綱要圖

## 3-6 資料表欄位設計

表5至表21為資料庫各個資料表欄位設計，針對圖9每個資料表，我們說明了英文欄位名稱、資料型態、可否為空值、備註、外來鍵或主鍵。

表 5 使用者資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 使用者users | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| email | email | varchar(255) | No | 帳號 |  |
| 密碼 | password | varchar(255) | No |  |  |
| 姓名 | name | varchar(255) | No |  |  |
| 性別 | sex | varchar(255) | No |  |  |
| 電話 | phone | varchar(255) | No |  |  |
| 銀行帳戶 | account | varchar(255) | No |  |  |

表 6 房東資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 房東 owners | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 使用者編號 | user\_id | int(11) | No |  | 外來鍵 |

表 7租客資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 租客 renters | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 使用者編號 | user\_id | int(11) | No |  | 外來鍵 |

表 8 系統管理員資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 系統管理員 managers | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 使用者編號 | user\_id | int(11) | No |  | 外來鍵 |

表 9 系統公告資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 系統公告 manager\_posts | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 系統管理員 編號 | manager\_id | int(11) | No |  | 外來鍵 |
| 內容 | content | varchar(255) | No |  |  |
| 標題 | title | varchar(255) | No |  |  |
| 日期 | date | time | No |  |  |

表 10 公告資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 公告 posts | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 地點編號 | location\_id | int(11) | No |  | 外來鍵 |
| 房東編號 | owner\_id | int(11) | No |  | 外來鍵 |
| 內容 | content | varchar(255) | No |  |  |
| 標題 | title | varchar(255) | No |  |  |
| 日期 | date | time | No |  |  |

表 11 合約紀錄資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 合約紀錄 contracts\_detail | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 租客編號 | renter\_id | int(11) | No |  | 外來鍵 |
| 目前租客編號 | now\_renter\_id | int(11) | No |  | 外來鍵 |
| 房屋編號 | house\_id | int(11) | No |  | 外來鍵 |
| 起始日 | start\_date | time | No |  |  |
| 到期日 | end\_date | time | No |  |  |

表 12 目前租客資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 目前租客 now\_renter | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 租客編號 | renter\_id | int(11) | No |  | 外來鍵 |
| 房屋編號 | house\_id | int(11) | No |  | 外來鍵 |

表 13 房屋

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 房屋 houses | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 地址 | address | varchar(255) | No |  |  |
| 房屋名稱 | name | varchar(255) | No |  |  |
| 介紹 | introduce | varchar(255) | No |  |  |
| 租賃狀態 | lease\_status | varchar(255) | No |  |  |
| 可住人數 | people | varchar(255) | No |  |  |
| 最短租期 | min\_period | varchar(255) | No |  |  |
| 格局 | pattern | varchar(255) | No |  |  |
| 坪數 | square\_meters | int(11) | No |  |  |
| 類型 | type | varchar(255) | No |  |  |
| 樓層 | floor | int(11) | No |  |  |
| 狀態 | status | varchar(255) | No |  |  |
| 地點編號 | location\_id | int(11) | No |  | 外來鍵 |
| 房東編號 | owner\_id | int(11) | No |  | 外來鍵 |

表 14 地點資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 地點 locations | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 房東編號 | owner\_id | int(11) | No |  | 外來鍵 |
| 地點名稱 | name | varchar(255) | No |  |  |

表 15 房屋圖片資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 房屋圖片 images | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 房屋編號 | house\_id | int(11) | No |  | 外來鍵 |
| 圖片 | image | varchar(255) | No |  |  |

表 16 房屋特色資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 房屋特色 features | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 房屋編號 | house\_id | int(11) | No |  | 外來鍵 |
| 特色 | feature | varchar(255) | No |  |  |

表 17 房屋設備資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 房屋設備 devices | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 房屋編號 | house\_id | int(11) | No |  | 外來鍵 |
| 設備 | device | varchar(255) | No |  |  |

表 18 報修資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 報修 repairs | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 租客編號 | renter\_id | int(11) | No |  | 外來鍵 |
| 房屋編號 | house\_id | int(11) | No |  | 外來鍵 |
| 狀態 | status | varchar(255) | No |  |  |
| 內容 | content | varchar(255) | No |  |  |
| 日期 | date | time | No |  |  |

表 19 報修回報資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 維修回報 repairs\_returns | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 報修編號 | repair\_id | int(11) | No |  | 外來鍵 |
| 內容 | content | varchar(255) | No |  |  |
| 日期 | date | time | No |  |  |

表 20 應繳款項資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 應繳款項 payments | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 租客編號 | renter\_id | int(11) | No |  | 外來鍵 |
| 租客費用編號 | cost\_id | int(11) | No |  | 外來鍵 |
| 費用名稱 | name | varchar(255) | No |  |  |
| 繳費日期 | date | time | No |  |  |
| 繳納狀態 | status | varchar(255) | No |  |  |
| 繳納方式 | method | varchar(255) | No |  |  |
| 金額 | amount | int(11) | No |  |  |
| 開始日期 | start\_date | time | No |  |  |
| 繳費期限 | deadline | time | No |  |  |

表 21 租屋費用資料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 租屋費用 costs | | | | | |
| **欄位名稱** | **英文欄位名稱** | **資料型態** | **可否為空值** | **備註** | **主鍵** |
| **/外來鍵** |
| 編號 | id | int(11) | No | 自動遞增 | 主鍵 |
| 房屋編號 | house\_id | int(11) | No |  | 外來鍵 |
| 種類 | type | varchar(255) | No |  |  |
| 金額 | amount | int(11) | No |  |  |
| 區間 | interval | varchar(255) | No |  |  |

## 3-7 訪客及會員相關使用案例分析與設計

### 訪客/會員瀏覽平台首頁

1. 簡述：瀏覽平台首頁，內容包含租屋資訊及類別
2. 參與者：使用者(訪客/會員)
3. 前提：開啟瀏覽器
4. 成功事後狀態：顯示平台首頁
5. 失敗事後狀態：顯示失敗訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 使用者點選平台頁面 | 1. 系統顯示平台首頁 |

1. 活動圖

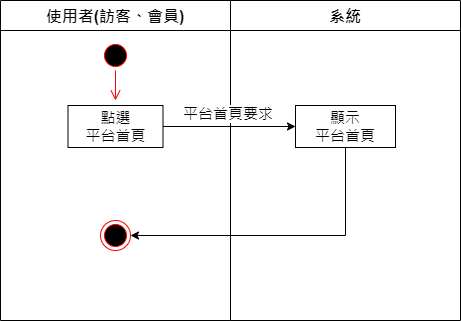


圖 10 訪客/會員瀏覽平台首頁

1. 路徑設計
2. 顯示平台(首頁)頁面
3. URL：/
4. Request method：get
5. Name：home.index
6. Request Data：none
7. Model： none
8. Controller and its method： HomeController@index

{

//找出所有刊登中資料，並依照創建日期排序

將資料傳入home視圖中呈現

}

1. View ：home.index

登入前

一張含有 桌 的圖片

自動產生的描述

登入後

一張含有 桌 的圖片

自動產生的描述

### 訪客/會員查詢出租房屋(地區)

1. 簡述：在搜尋框輸入關鍵字(地區)，例如太平區
2. 參與者：使用者(訪客/會員)
3. 前提：點選查詢房屋
4. 成功事後狀態：顯示搜尋結果
5. 失敗事後狀態：顯示查無資料
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 使用者輸入關鍵字(keyword) | 1. 系統依關鍵字搜尋資料庫符合條件資料 |
|  | 1. 顯示搜尋結果 |

1. 活動圖

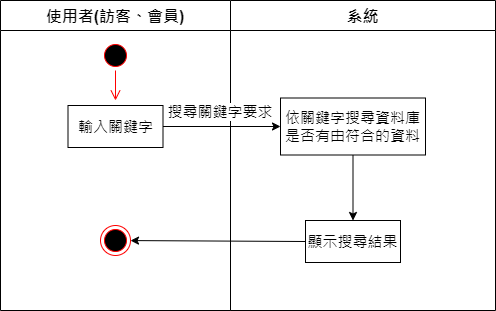


圖 11 訪客/會員查詢出租房屋

1. 路徑設計
2. 顯示包含關鍵字的所有房屋要求
3. URL：houses/search
4. Request method：get
5. Name：houses.search
6. Request Data：keyword
7. Model： House
8. Controller and its method： HouseController@search

search (Request $request)

{

//依輸入的關鍵字搜尋houses裡符合的資料

}

1. View ：home.index

### 訪客/會員篩選租屋條件

1. 簡述：篩選租屋條件，比如要幾坪，要可以寵物
2. 參與者：使用者(訪客/會員)
3. 前提：進入平台首頁
4. 成功事後狀態：顯示搜尋結果
5. 失敗事後狀態：顯示查無資料
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 選擇進階搜尋 | 1. 提供勾選條件 |
| 1. 選擇租屋條件 | 1. 顯示符合條件的篩選結果 |

1. 活動圖

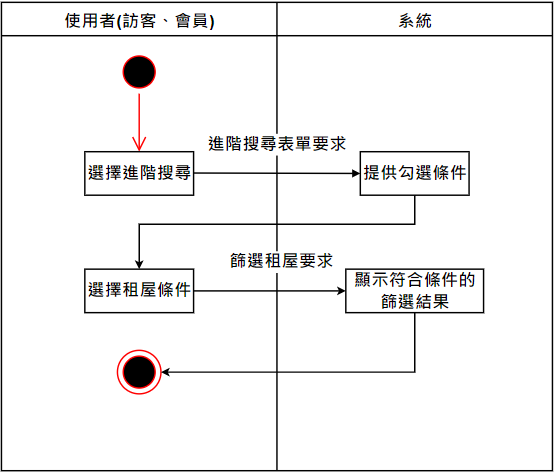


圖 12 訪客/會員篩選租屋條件

1. 路徑設計
2. 顯示進階搜尋表單要求
3. URL：houses/advance\_search/create
4. Request method：get
5. Name：houses.advance\_search.create
6. Request Data：none
7. Model： House
8. Controller and its method： HouseController@advance\_search\_create

advance\_search\_create (){

取得所有房屋資料$category

$date = ['category'=>$category,'area'=>$area];

return View('house.advance\_search',$date);

將資料傳入advance\_search視圖中呈現

}

1. View ：advance\_search($category, $area)
2. 訪客搜尋房屋要求
3. URL：houses/advance\_search
4. Request method：get
5. Name：houses.advance\_search
6. Request Data：房屋條件
7. Model： House
8. Controller and its method： HouseController@advance\_search

advance\_search(Request $request){

由$request取得輸入的條件

找出房屋資料表內符合的資料$houses

將資料傳入house視圖中呈現

}

1. View ：index($houses,$categories)

一張含有 文字 的圖片

自動產生的描述

### 訪客/會員查看租屋資訊

1. 簡述：查看房屋詳細資訊
2. 參與者：使用者(訪客/會員)
3. 前提：在平台頁面
4. 成功事後狀態：顯示房屋詳細資訊頁面
5. 失敗事後狀態：顯示錯誤訊息，返回租屋資訊頁面
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 使用者點選某一個房屋 | 1. 系統顯示租屋資訊頁面 |

1. 活動圖

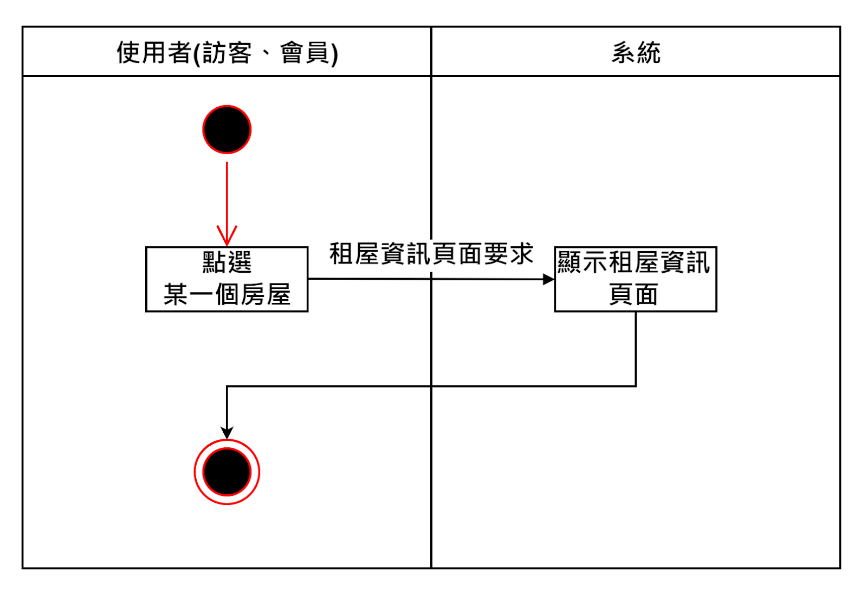


圖 13 訪客/會員查看租屋資訊

1. 路徑設計
2. 顯示進階搜尋表單要求
3. URL：houses/{house}
4. Request method：get
5. Name：houses.show
6. Request Data：none
7. Model： House
8. Controller and its method： HouseController@show

show(House $house)

{

//顯示租屋資訊頁面

}

1. View ：houses.show

一張含有 廣場 的圖片

自動產生的描述

### 訪客訪客註冊會員

1. 簡述：使用者註冊會員
2. 參與者：使用者(訪客)
3. 前提：進入到註冊頁面
4. 成功事後狀態：註冊成功，平台首頁
5. 失敗事後狀態：註冊失敗，返回註冊頁面
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 使用者點選註冊 | 1. 系統顯示註冊頁面 |
| 1. 使用者輸入基本資料 | 1. 驗證資料是否正確 |
|  | 1. 儲存會員資料至資料庫 |

1. 活動圖

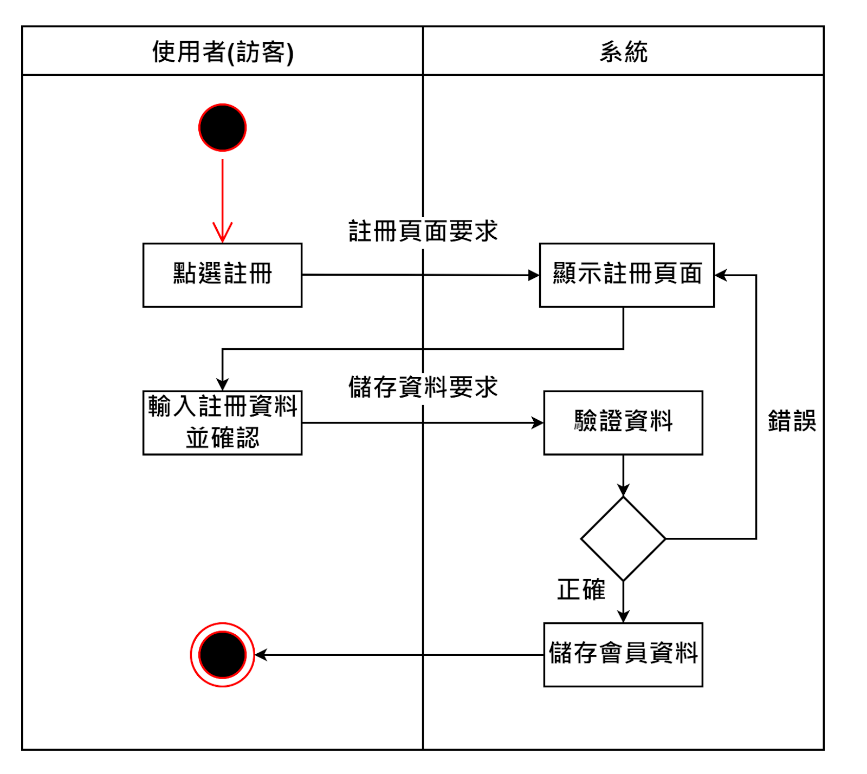


圖 14 訪客註冊會員

1. 路徑設計
2. 註冊頁面要求
3. URL：register
4. Request method：get
5. Name：register
6. Request Data：none
7. Model： User
8. Controller and its method： RegisteredUserController@create

create()

{

//註冊頁面

}

1. View ： register(回傳註冊頁面)
2. 儲存會員資料要求
3. URL：register
4. Request method：post
5. Name：users.store
6. Request Data：account, password
7. Model： User
8. Controller and its method： RegisteredUserController@store

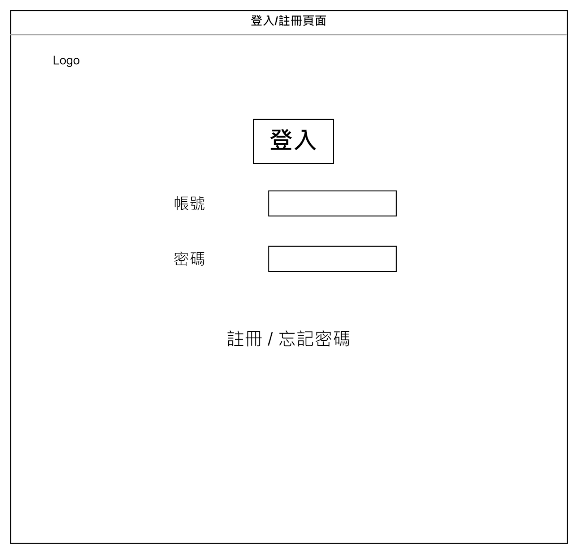
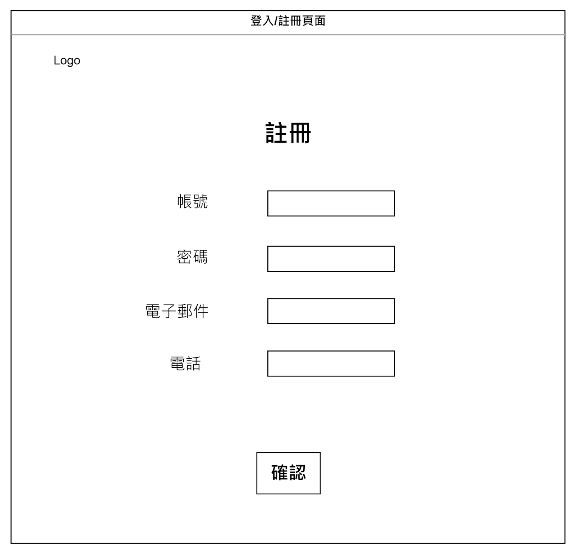
store(Request $request)

{

//儲存帳號密碼

}

1. View ：login

### 訪客登入會員

1. 簡述：訪客登入
2. 參與者：使用者(訪客)
3. 前提：擁有本平台會員
4. 成功事後狀態：成功登入，返回平台首頁
5. 失敗事後狀態：登入失敗，返回登入頁面
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 使用者點選登入 | 1. 系統顯示登入會員資料頁面 |
| 1. 使用者輸入帳號密碼 | 1. 驗證資料是否正確 |
|  | 1. 進入主頁面 |

其他例外路徑

* 步驟3若有輸入帳號密碼有誤則顯示系統顯示帳號密碼有誤訊息，請重新輸入

1. 活動圖

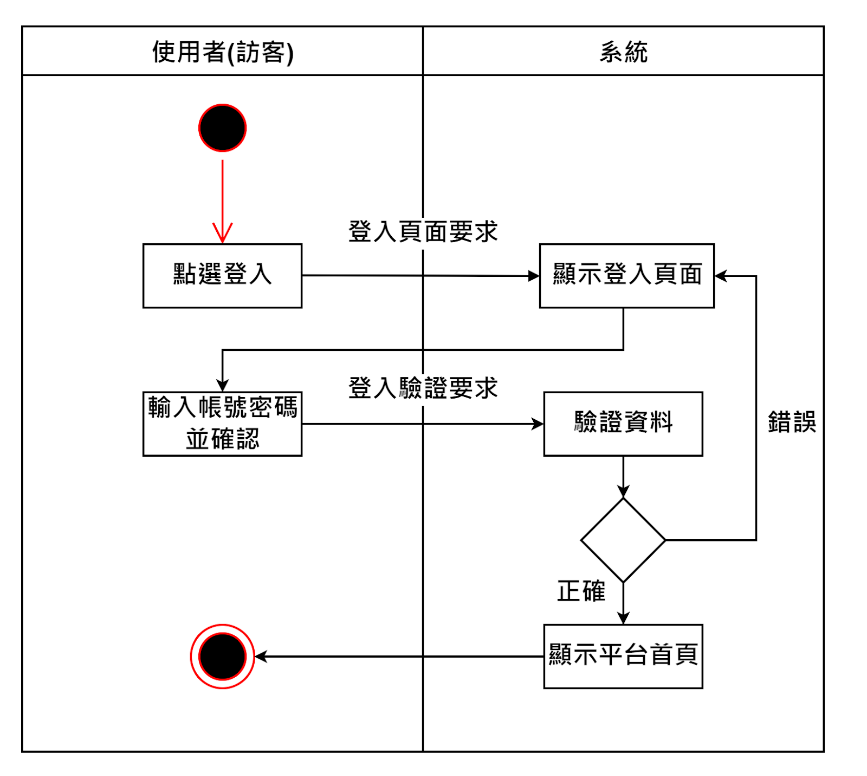


圖 15 訪客登入會員

1. 路徑設計
2. 會員登入頁面要求
3. URL：login
4. Request method：get
5. Name：login
6. Request Data：none
7. Model： User
8. Controller and its method： AuthenticatedSessionController@create

create

{

//登入頁面

}

1. View ：login
2. 登入驗證要求
3. URL：login
4. Request method：post
5. Name：users.data
6. Request Data：users.data
7. Model： User
8. Controller and its method： AuthenticatedSessionController@store

store(Request $request)

{

//驗證帳號密碼

}

1. View ：home.index



## 3-8 會員(房東/租客)相關使用案例分析與設計

### \*會員(房東/租客)顯示會員資料

1. 簡述：會員查看個人資料
2. 參與者：會員(房東/租客)
3. 前提：會員已登入
4. 成功事後狀態：顯示會員資料
5. 失敗事後狀態：顯示失敗訊息，重新進入會員中心頁面
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **會員動作** | **系統回應** |
| 1. 會員點選會員資料 | 1. 系統顯示會員資料頁面 |

1. 活動圖

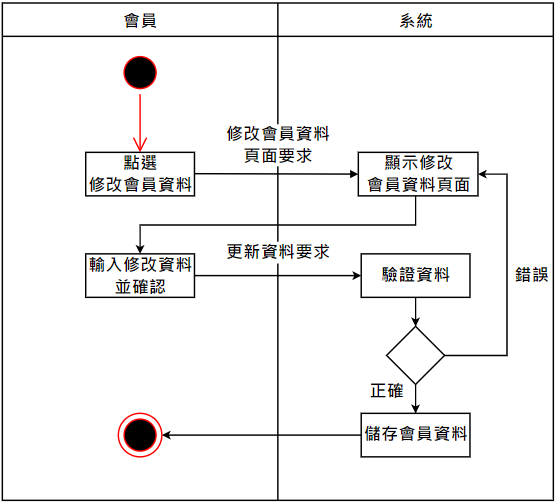


圖 16 會員(房東/租客)修改會員資料

1. 路徑設計
2. 修改會員資料頁面要求
3. URL：users/{user}/edit
4. Request method：get
5. Name：users.edit
6. Request Data：none
7. Model：  User
8. Controller and its method：UserProfileController@edit  
   edit(User $user)  
   {  
      //使用auth()->user()->id抓出並顯示會員資料

}

View ：users.edit

### 會員(房東/租客)修改會員資料

1. 簡述：會員修改資料
2. 參與者：會員(房東/租客)
3. 前提：會員已登入，且在會員中心頁面
4. 成功事後狀態：顯示修改成功，重新進入會員中心頁面
5. 失敗事後狀態：顯示失敗訊息，重新進入會員中心頁面
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **會員動作** | **系統回應** |
| 1. 會員點選修改會員資料 | 1. 系統跳出修改會員資料頁面 |
| 1. 會員修改會員資料 | 1. 驗證資料 |
|  | 1. 儲存會員資料 |

其他例外路徑

* 使用者修改會員資料：選擇取消，返回會員中心頁面
* 步驟3若有錯誤資訊則須重新輸入

1. 活動圖

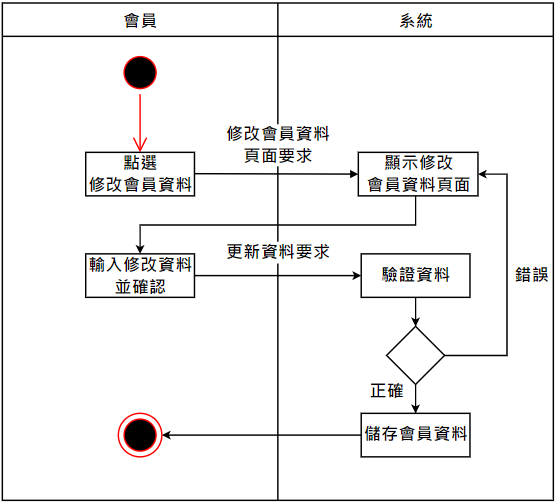


圖 17 會員(房東/租客)修改會員資料

1. 路徑設計
2. 修改會員資料頁面要求
3. URL：users/{user}/edit
4. Request method：get
5. Name：users.edit
6. Request Data：none
7. Model：  User
8. Controller and its method：UserProfileController@edit  
   edit(User $user)  
   {  
      //使用auth()->user()->id抓出並顯示會員資料

}

1. View ：users.edit
2. 更新會員資料要求
3. URL：users/{user}
4. Request method：patch
5. Name：users.update
6. Request Data：user data
7. Model：  User
8. Controller and its method：ProfileInformationController@update

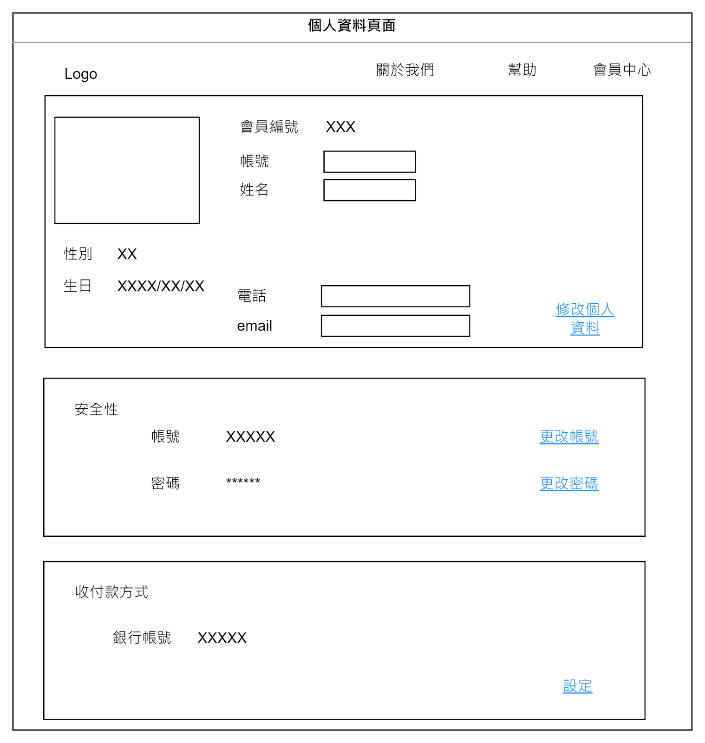
update(Request $request,User $user)

{

//依表單傳進來的資料，修改並儲存user的資料

}

1. Redirect to：personal，畫面設計如下



### 會員(房東/租客)新增轉帳資訊

1. 簡述：會員新增銀行帳戶資料
2. 參與者：會員(房東/租客)
3. 前提：會員已登入，且在會員中心頁面
4. 成功事後狀態：顯示新增成功，重新進入會員中心頁面
5. 失敗事後狀態：顯示失敗訊息，重新進入會員中心頁面
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **會員動作** | **系統回應** |
| 1. 點選新增按鈕 | 1. 回傳新增銀行帳戶資料頁面 |
| 1. 輸入銀行帳戶資料 | 1. 驗證資料 |
|  | 1. 將資料新增至資料庫 |
|  | 1. 跳回會員中心頁面 |

1. 活動圖

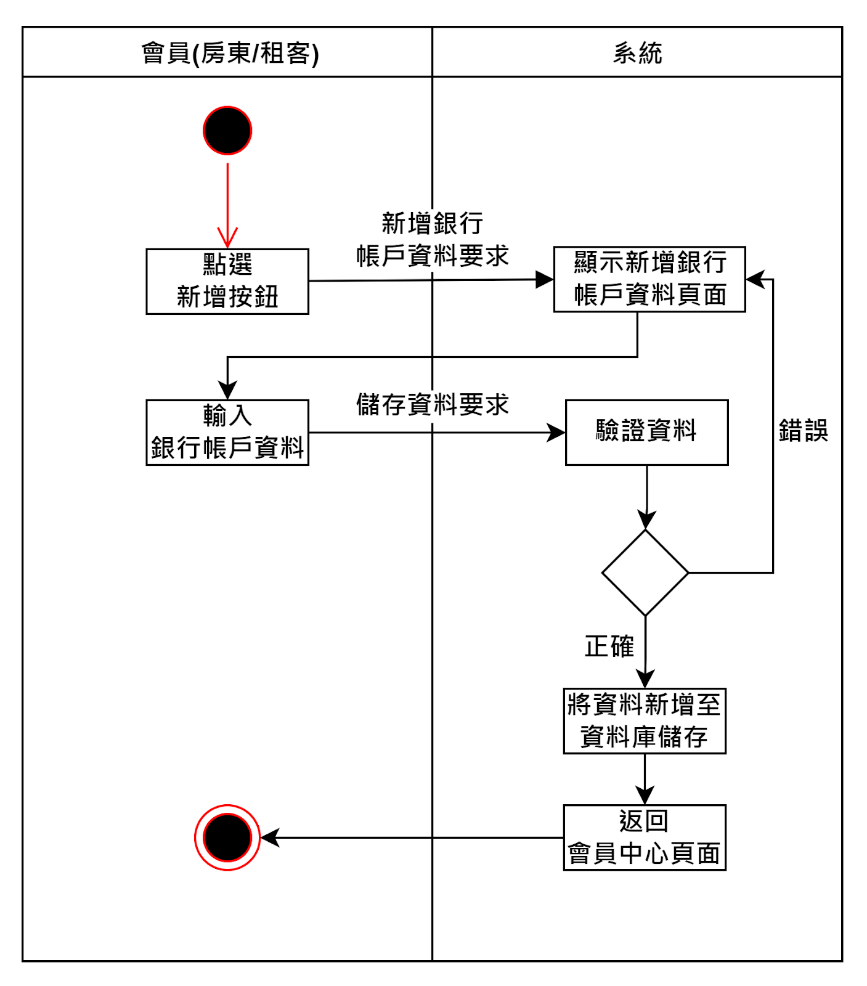


圖 18 會員(房東/租客)新增轉帳資訊

1. 路徑設計
2. 新增轉帳資料頁面要求
3. URL：users/{user}/payment/create
4. Request method：get
5. Name：users.payment.create
6. Request Data：none
7. Model：User
8. Controller and its method： UserPaymentController@create  
   create()  
   {  
      //回傳新增轉帳資料頁面

}

1. View ：users.payment.create
2. 儲存轉帳資料要求
3. URL：users/{user}/payment
4. Request method：post
5. Name：users.payment.store
6. Request Data：轉帳資料
7. Model：User
8. Controller and its method： UserPaymentController@store  
   store(Request $request)  
   {  
      //依表單傳進來的資料，儲存payment的資料

}

1. Redirect to：personal

### 會員(房東/租客)修改轉帳資訊

1. 簡述：會員修改銀行帳戶資料
2. 參與者：會員(房東/租客)
3. 前提：會員已登入，且在會員中心頁面，點擊修改按鈕
4. 成功事後狀態：修改成功
5. 失敗事後狀態：修改失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **會員動作** | **系統回應** |
| 1. 點選修改按鈕 | 1. 回傳修改銀行帳戶資料頁面 |
| 1. 輸入要修改的銀行帳戶資料 | 1. 驗證資料 |
|  | 1. 將資料更新至資料庫 |
|  | 1. 跳回會員中心頁面 |

1. 活動圖

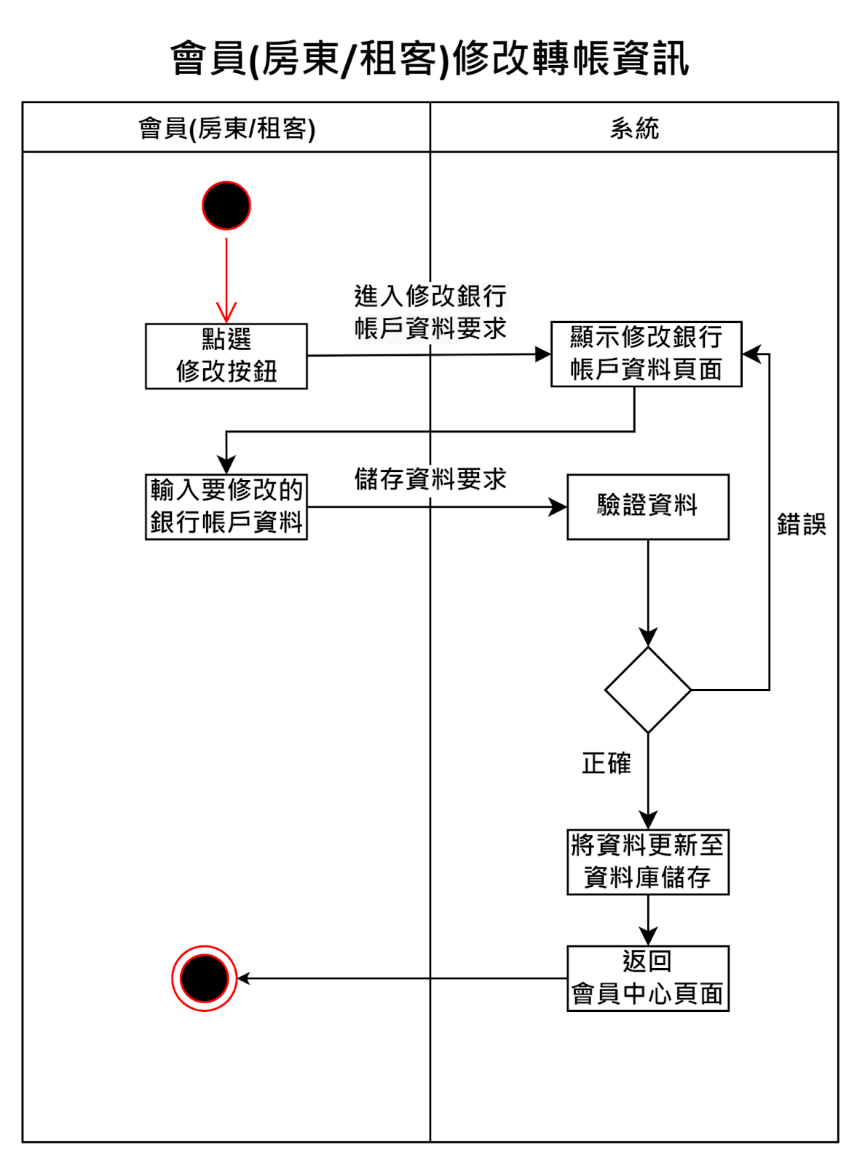


圖 19 會員(房東/租客)修改轉帳資訊

1. 路徑設計
2. 修改轉帳資料頁面要求
3. URL：users/{user}/payment/{payment}/edit
4. Request method：get
5. Name：users.payment.edit
6. Request Data：none
7. Model： User
8. Controller and its method： UserPaymentController@edit

edit(User $user)

{

//回傳修改轉帳資料頁面

}

1. View ：users.payment.edit
2. 儲存轉帳資料要求
3. URL：users/{user}/payment/{payment}
4. Request method：patch
5. Name：users.payment.update
6. Request Data：none
7. Model： User
8. Controller and its method： UserPaymentController@update

update(Request $request,User $user)

{

//依表單傳進來的資料，儲存修改後的payment資料

}

1. Redirect to：personal

### 會員(房東/租客)顯示租客頁面

1. 簡述：進入租客頁面，裡面可以檢視所承租的房屋的公告、房屋資訊、合約、費用、報修等
2. 參與者：會員(房東/租客)
3. 前提：已登入在平台頁面
4. 成功事後狀態：顯示租客頁面
5. 失敗事後狀態：顯示失敗訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **會員動作** | **系統回應** |
| 1. 會員點選租客頁面 | 1. 系統顯示租客頁面 |

1. 活動圖

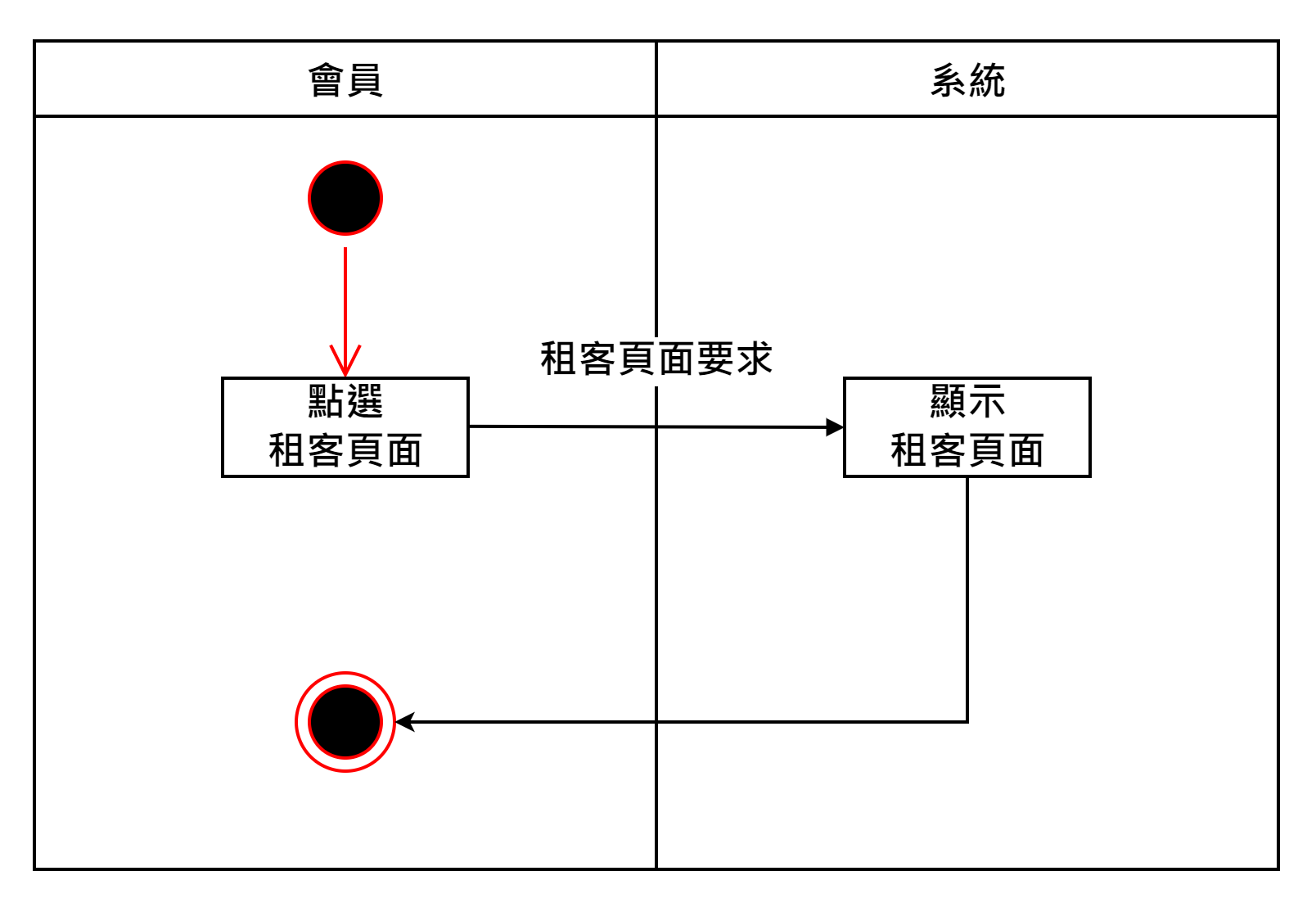


圖 20 會員(房東/租客)進入租客頁面

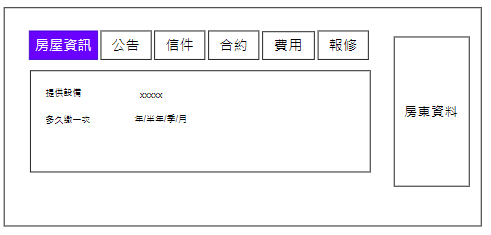
1. 路徑設計
2. 顯示進階搜尋表單要求
3. URL：users/renters/{renter}
4. Request method：get
5. Name：renters.home.index
6. Request Data：none
7. Model： none
8. Controller and its method： RenterController@index

index(){

//將租客資料傳入renters.home視圖中呈現;

}

1. View ：renters.home.index



### 會員(房東/租客)顯示房東管理頁面

1. 簡述：點選房東，進入房東管理頁面，表列所有地點、房屋及其狀態
2. 參與者：會員(房東/租客)
3. 前提：已登入在平台頁面
4. 成功事後狀態：顯示房東管理頁面
5. 失敗事後狀態：顯示失敗訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **會員動作** | **系統回應** |
| 1. 會員點選房東管理 | 1. 系統顯示房東管理頁面 |

1. 活動圖

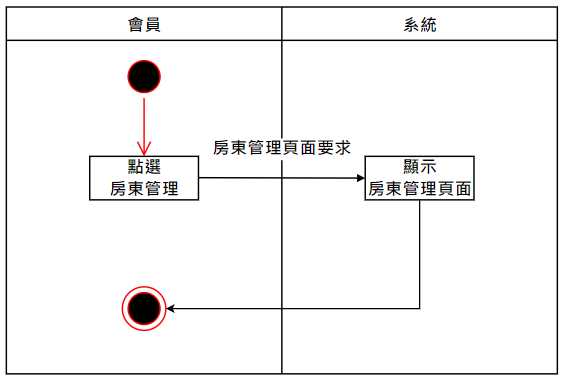


圖 21 會員(房東/租客)進入房東管理頁面

1. 路徑設計
2. 顯示房東管理頁面要求
3. URL：users/owners/{owner}
4. Request method：get
5. Name：owners.home.index
6. Request Data：none
7. Model：none
8. Controller and its method： OwnerController@index

index(){

//將房東資料傳入owners.home.index視圖中呈現;

}

1. View ：owners.home.index

一張含有 文字 的圖片

自動產生的描述

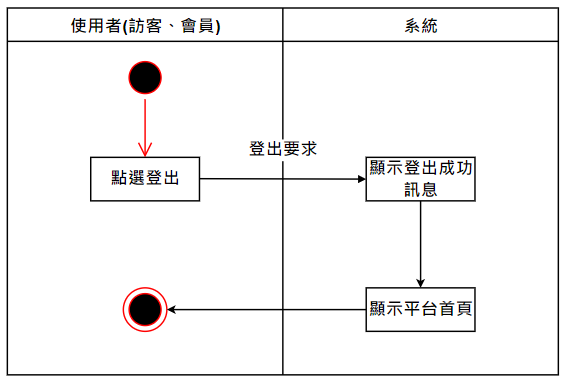
### 會員登出

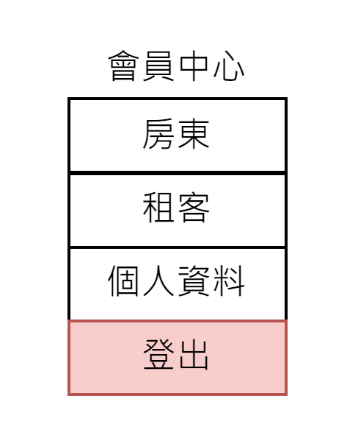
1. 簡述：會員登出
2. 參與者：會員
3. 前提：會員已登入
4. 成功事後狀態：登出成功，返回平台首頁
5. 失敗事後狀態：登出失敗，返回平台首頁
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **會員動作** | **系統回應** |
| 1. 使用者點選登出 | 1. 顯示登出成功 |
|  | 1. 顯示主頁面 |

1. 活動圖



1. 路徑設計
2. 登出要求
3. URL：logout
4. Request method：delete
5. Name：logout
6. Request Data：none
7. Model：User
8. Controller and its method： AuthenticatedSessionController@destroy

destroy()

{

一張含有 文字 的圖片

自動產生的描述

}

1. View ：home.index

## 3-9 會員(房東)相關使用案例分析與設計

### \*會員(房東)查看地點

1. 簡述：房東新增出租地點
2. 參與者：房東
3. 前提：在房東管理畫面下，點擊新增出租地點
4. 成功事後狀態：新增成功
5. 失敗事後狀態：新增失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 輸入地點資料 |  |
| 1. 點選新增按鈕 | 1. 驗證資料 |
|  | 1. 資料新增至資料庫 |
|  | 1. 跳回房屋管理畫面 |

其他例外路徑

* 若輸入資料驗證錯誤，則跳出錯誤訊息。

1. 活動圖

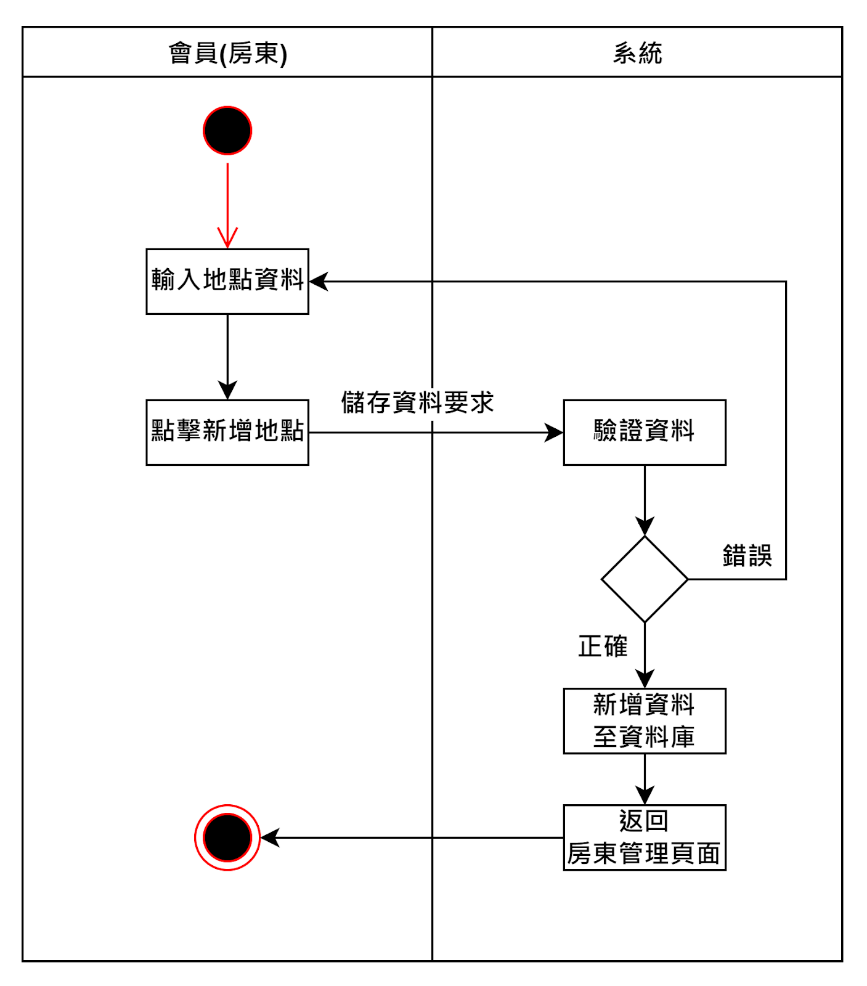


圖 22 會員(房東)新增地點

1. 路徑設計
2. 儲存地點資料要求
3. URL：owners/locations
4. Request method：post
5. Name：owners.locations.store
6. Request Data：新的地點
7. Model：Location
8. Controller and its method： LocationController@store

store(Request $request)

{

//在Location的Model中新增房屋資料，利用$request在頁面上存取的地點，回到房東管理頁面

}

1. Redirect to：owners.home.index

一張含有 桌 的圖片

自動產生的描述

### 會員(房東)新增地點

1. 簡述：房東新增出租地點
2. 參與者：房東
3. 前提：在房東管理畫面下，點擊新增出租地點
4. 成功事後狀態：新增成功
5. 失敗事後狀態：新增失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 輸入地點資料 |  |
| 1. 點選新增按鈕 | 1. 驗證資料 |
|  | 1. 資料新增至資料庫 |
|  | 1. 跳回房屋管理畫面 |

其他例外路徑

* 若輸入資料驗證錯誤，則跳出錯誤訊息。

1. 活動圖

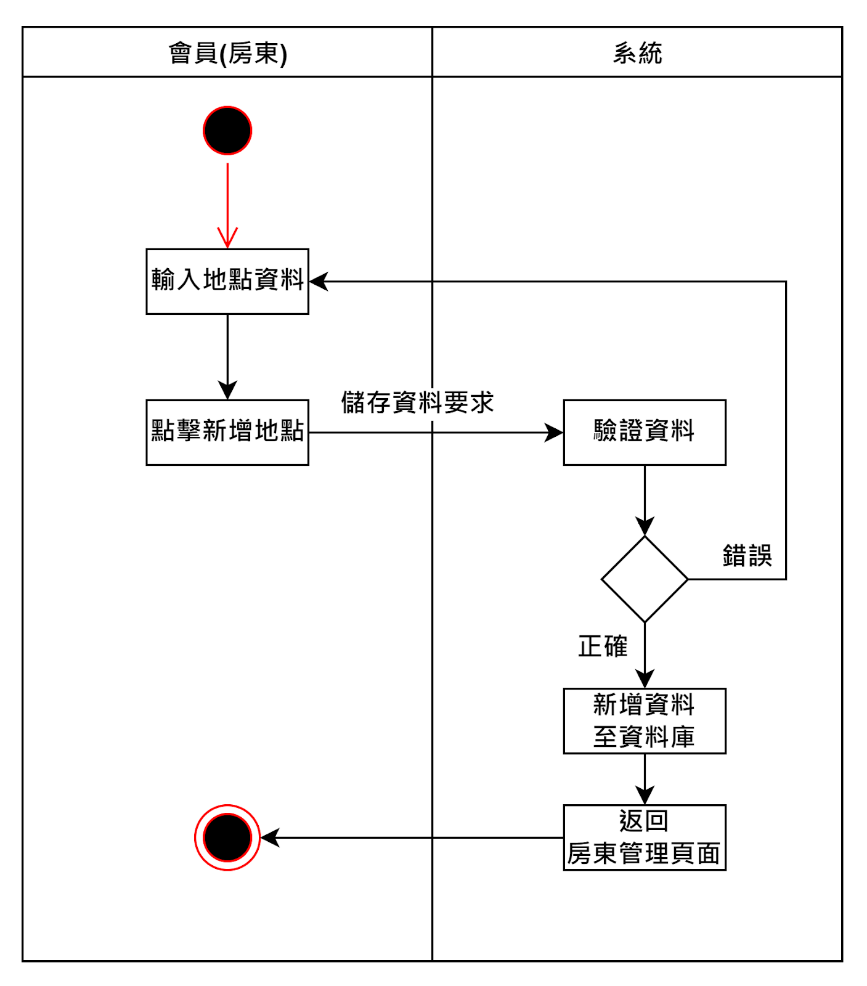


圖 23 會員(房東)新增地點

1. 路徑設計
2. 儲存地點資料要求
3. URL：owners/locations
4. Request method：post
5. Name：owners.locations.store
6. Request Data：新的地點
7. Model：Location
8. Controller and its method： LocationController@store

store(Request $request)

{

//在Location的Model中新增房屋資料，利用$request在頁面上存取的地點，回到房東管理頁面

}

1. Redirect to：owners.home.index

一張含有 桌 的圖片

自動產生的描述

### 會員(房東)修改地點

1. 簡述：房東修改地點
2. 參與者：房東
3. 前提：在房東管理畫面的地點下，點擊修改地點
4. 成功事後狀態：修改成功
5. 失敗事後狀態：修改失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選修改按鈕 | 1. 回傳修改地點頁面 |
| 1. 輸入要修改的地點資料 | 1. 驗證資料 |
|  | 1. 將資料更新至資料庫 |
|  | 1. 跳回房屋管理畫面 |

1. 活動圖

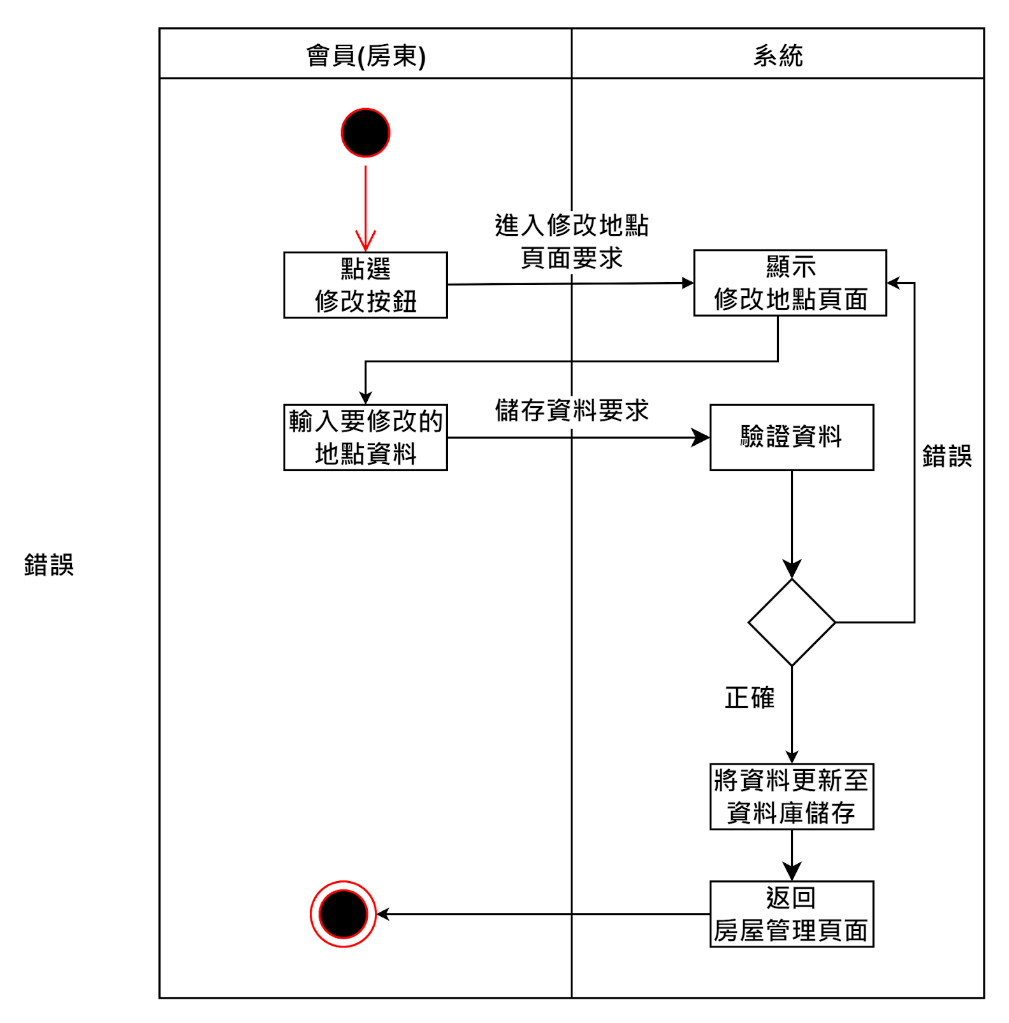


圖 24 會員(房東)修改地點

1. 路徑設計
2. 進入修改地點頁面要求
3. URL：owners/locations/{location}/edit
4. Request method：get
5. Name：owners.locations.edit
6. Request Data：none
7. Model：Location
8. Controller and its method： LocationController@edit

edit(Location $location)

{

//回傳修改地點頁面

}

1. View ：修改地點頁面 owners.locations.edit
2. 儲存房屋資料要求
3. URL：owners/locations/{location}
4. Request method：patch
5. Name：owners.locations.update
6. Request Data：修改後的地點資料
7. Model：Location
8. Controller and its method：LocationController@update

update(Request $request,Location $location)

{

//找到要更新的地點資料$location，利用$request在頁面上的地點資料來更新Location欄位在$location中的房屋資料，跳轉到房東租屋頁面。

}

1. Redirect to：owners.home.index

### 會員(房東)刪除地點

1. 簡述：房東刪除地點
2. 參與者：房東
3. 前提：在房東管理畫面的地點下，點擊刪除地點
4. 成功事後狀態：刪除成功
5. 失敗事後狀態：刪除失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 會員(房東)點選刪除按鈕 | 1. 跳出視窗，詢問確定刪除該房屋嗎? |
| 1. 點選確認 | 1. 刪除該地點 |
|  | 1. 返回房屋管理頁面 |

1. 活動圖

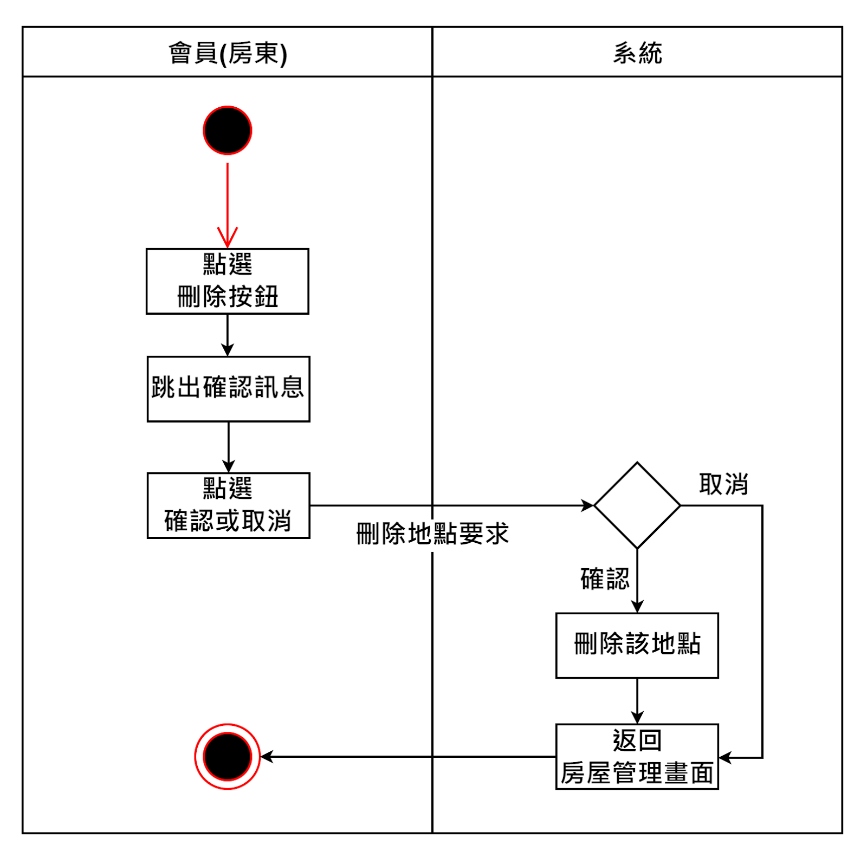


圖 25 會員(房東)刪除地點

1. 路徑設計
2. 刪除地點要求
3. URL：owners/locations/{location}
4. Request method：delete
5. Name：owners.locations.destroy
6. Request Data：none
7. Model：Location
8. Controller and its method： LocationController@destroy

destroy (Location $location )

{

Post::destroy($post);

}

1. Redirect to：owners.home.index

### \*會員(房東)查看房屋

1. 簡述：房東新增房屋
2. 參與者：房東
3. 前提：在房東管理畫面的地點下，點擊新增房屋
4. 成功事後狀態：新增成功
5. 失敗事後狀態：新增失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選新增按鈕 | 1. 回傳房屋新增畫面 |
| 1. 輸入房屋資料 | 1. 驗證資料 |
|  | 1. 將資料新增至資料庫 |
|  | 1. 跳回房屋管理畫面 |

1. 活動圖

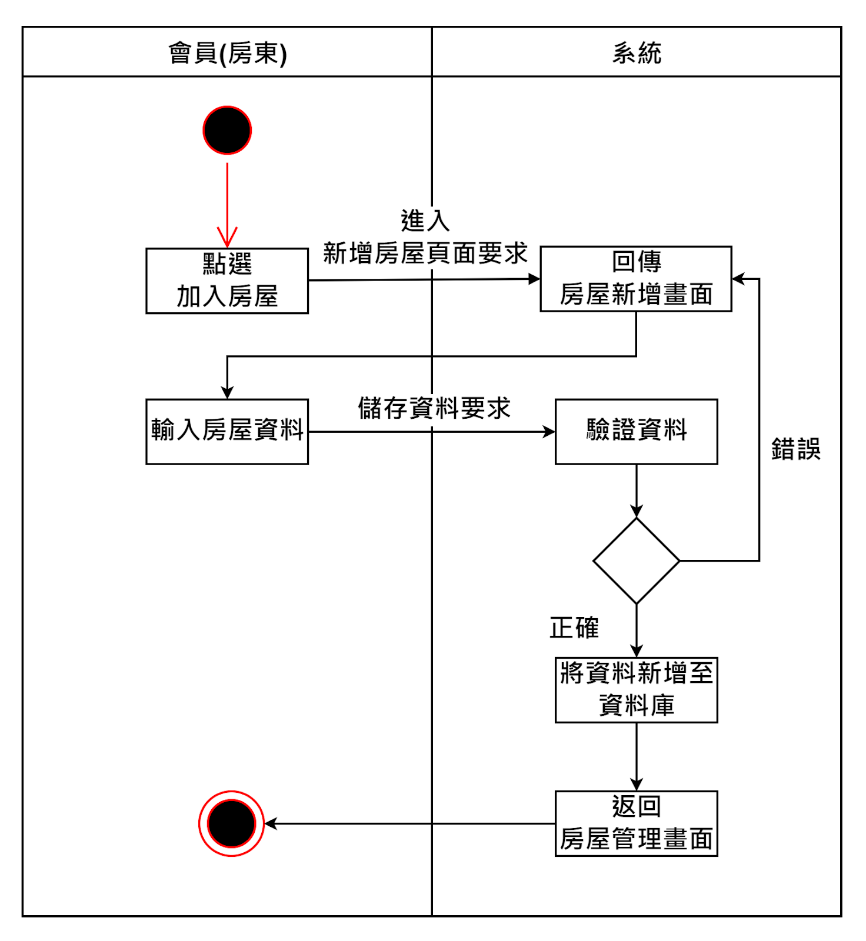


圖 26 會員(房東)新增房屋

1. 路徑設計
2. 進入新增房屋頁面要求
3. URL：owners/locations/{location}/houses/create
4. Request method：get
5. Name：owners.locations.houses.create
6. Request Data：none
7. Model：Location, House
8. Controller and its method： LocationHouseController@create

create( )

{

//回傳新增房屋頁面

}

1. View ：owners.locations.houses.create

一張含有 桌 的圖片

自動產生的描述

1. 儲存房屋資料要求
2. URL：owners/locations/{location}/houses
3. Request method：post
4. Name：owners.locations.houses.store
5. Request Data：房屋名稱、圖片、租金、詳細資料
6. Model：Location, House
7. Controller and its method： LocationHouseController@store

store(Request $request)

{

//在house的Model中新增房屋資料，利用$request在頁面上存取的房屋編號、房屋名稱、圖片、租金、詳細資料來儲存房屋資料表的各欄位，資料儲存後，回到房東管理畫面

}

1. Redirect to：owners.home.index

### 會員(房東)新增房屋

1. 簡述：房東新增房屋
2. 參與者：房東
3. 前提：在房東管理畫面的地點下，點擊新增房屋
4. 成功事後狀態：新增成功
5. 失敗事後狀態：新增失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選新增按鈕 | 1. 回傳房屋新增畫面 |
| 1. 輸入房屋資料 | 1. 驗證資料 |
|  | 1. 將資料新增至資料庫 |
|  | 1. 跳回房屋管理畫面 |

1. 活動圖

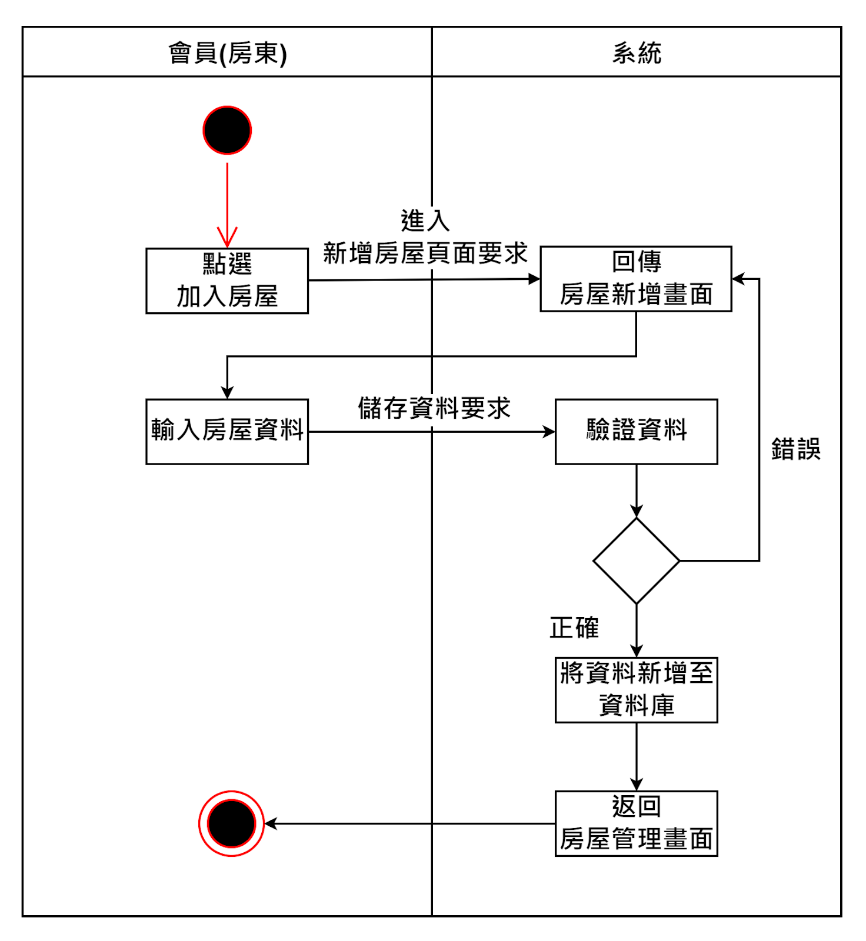


圖 27 會員(房東)新增房屋

1. 路徑設計
2. 進入新增房屋頁面要求
3. URL：owners/locations/{location}/houses/create
4. Request method：get
5. Name：owners.locations.houses.create
6. Request Data：none
7. Model：Location, House
8. Controller and its method： LocationHouseController@create

create( )

{

//回傳新增房屋頁面

}

1. View ：owners.locations.houses.create

一張含有 桌 的圖片

自動產生的描述

1. 儲存房屋資料要求
2. URL：owners/locations/{location}/houses
3. Request method：post
4. Name：owners.locations.houses.store
5. Request Data：房屋名稱、圖片、租金、詳細資料
6. Model：Location, House
7. Controller and its method： LocationHouseController@store

store(Request $request)

{

//在house的Model中新增房屋資料，利用$request在頁面上存取的房屋編號、房屋名稱、圖片、租金、詳細資料來儲存房屋資料表的各欄位，資料儲存後，回到房東管理畫面

}

1. Redirect to：owners.home.index

### 會員(房東)修改房屋

1. 簡述：修改房屋資料
2. 參與者：房東
3. 前提：已表列房屋資料，且點選修改按鈕
4. 成功事後狀態：修改成功
5. 失敗事後狀態：修改失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 房東點擊編輯按鈕 | 1. 系統回傳編輯房屋畫面 |
| 1. 房東輸入要編輯的房屋資料 | 1. 驗證資料 |
|  | 1. 將資料更新至資料庫 |
|  | 1. 系統跳回房屋管理畫面 |

1. 活動圖

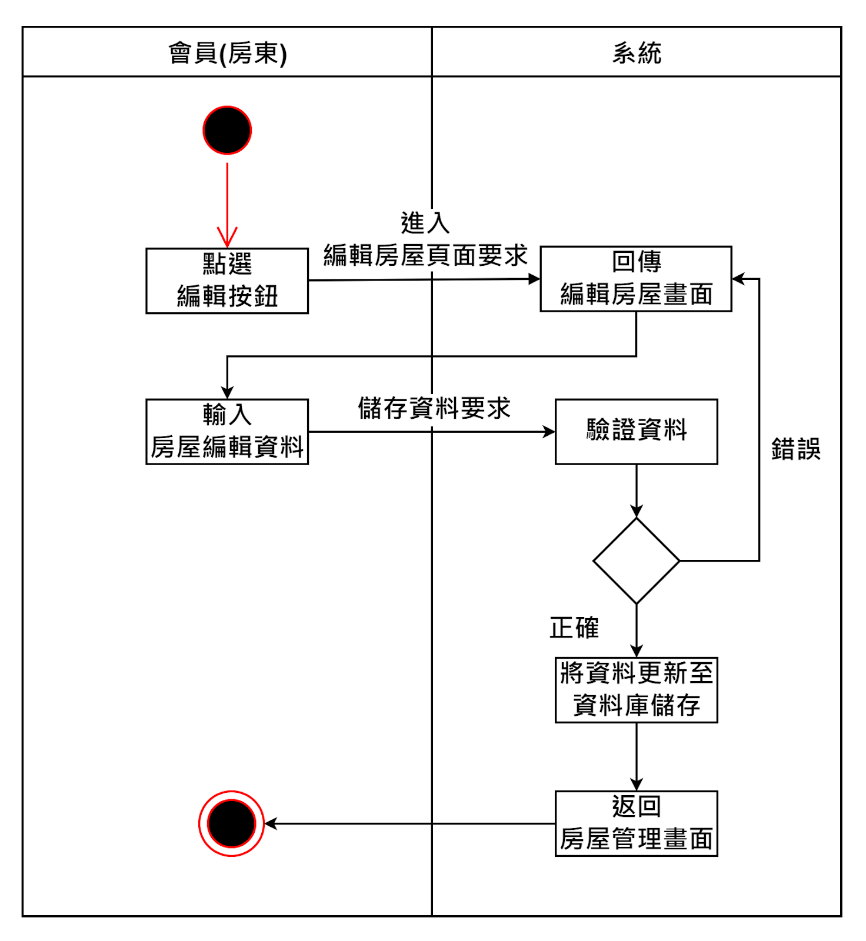


圖 28 會員(房東)修改房屋

1. 路徑設計
2. 進入修改房屋頁面要求
3. URL：owners/locations/{location}/houses/{house}/edit
4. Request method：get
5. Name：owners.locations.houses.edit
6. Request Data：none
7. Model：Location, House
8. Controller and its method： LocationHouseController@edit

edit(House $house )

{

//取得特定地點底下的某一筆房屋資料，進行修改

}

1. View ：locations.houses.edit
2. 儲存房屋資料要求
3. URL：owners/locations/{location}/houses/{house}
4. Request method：patch
5. Name：owners.locations.houses.update
6. Request Data：修改後的房屋資料
7. Model：House
8. Controller and its method：LocationHouseController@update

update(Request $request,House $house)

{

//找到要更新的房屋資料$house，利用$request在頁面上的房屋資料來更新House欄位在$ house中的房屋資料，跳轉到房東租屋頁面。

}

1. Redirect to：owners.home.index

### 會員(房東)刪除房屋

1. 簡述：房東刪除房屋資料
2. 參與者：房東
3. 前提：已表列房屋資料，且點選刪除按鈕
4. 成功事後狀態：刪除成功
5. 失敗事後狀態：刪除失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 會員(房東)點選刪除按鈕 | 1. 跳出視窗，詢問確定刪除該房屋嗎? |
| 1. 點選確認 | 1. 刪除該房屋 |
|  | 1. 返回房屋管理頁面 |

其他例外路徑

* 取消刪除房屋：在步驟三時選擇取消，返回房屋管理頁面

1. 活動圖

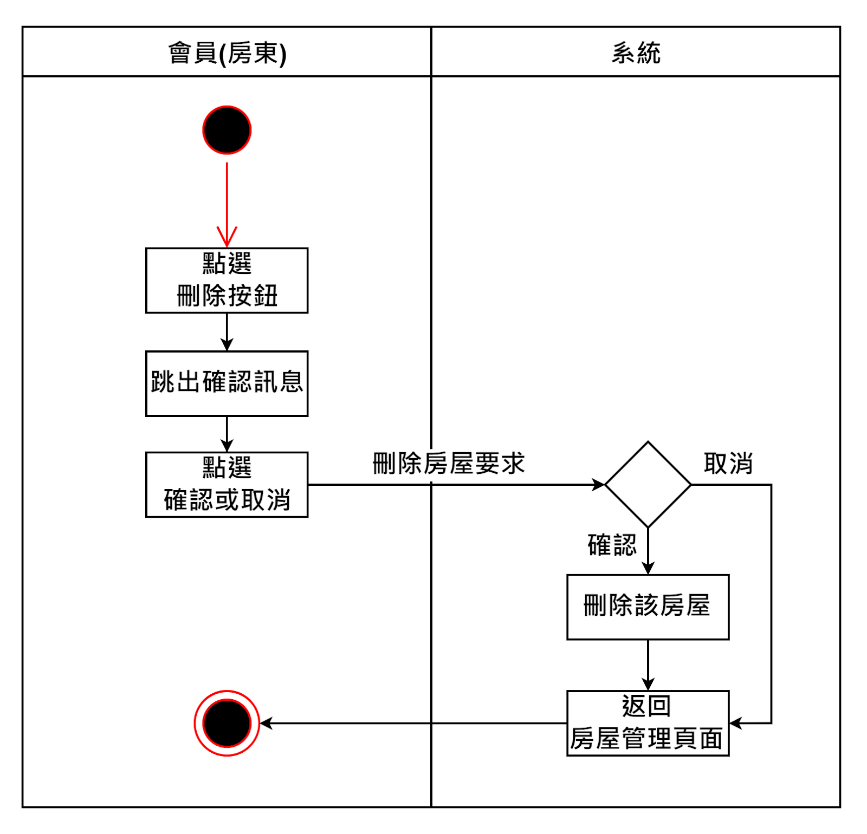


圖 29 會員(房東)刪除房屋

1. 路徑設計
2. 刪除房屋要求
3. URL：owners/locations/{location}/houses/{house}
4. Request method：delete
5. Name：owners.locations.houses.destroy
6. Request Data：none
7. Model：Location, House
8. Controller and its method：LocationHouseController@destroy

destroy (House $house)

{

Post::destroy($post);

}

1. Redirect to：owners.home.index

### 會員(房東)查看公告

1. 簡述：房東查看公告
2. 參與者：房東
3. 前提：進入房屋管理畫面的地點下，點擊公告按鈕
4. 成功事後狀態：顯示新增表單
5. 失敗事後狀態：顯示失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 選公告按鈕 | 1. 回傳公告頁面 |

1. 活動圖

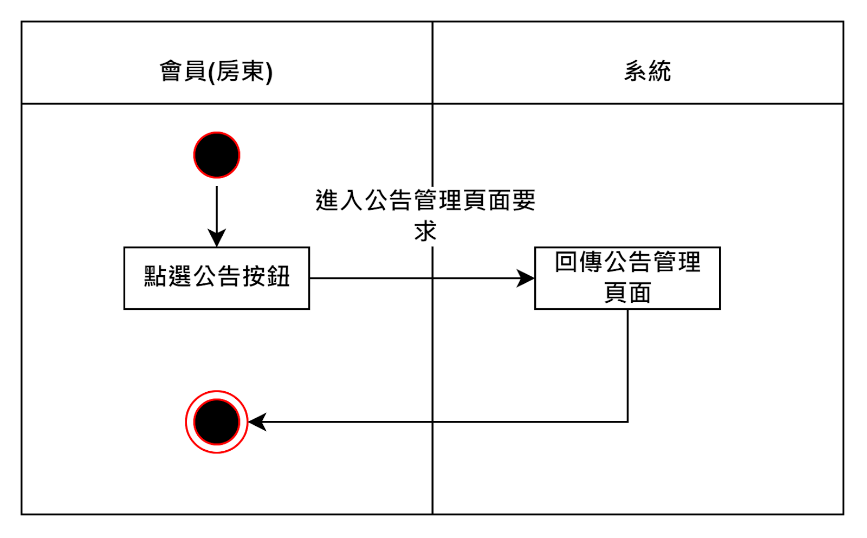


圖 30 會員(房東)查看公告

1. 路徑設計
2. 查看公告頁面要求
3. URL：owners/locations/{location}/posts
4. Request method：get
5. Name：owners.locations.posts.index
6. Request Data：none
7. Model：Location, Post
8. Controller and its method：LocationPostController@index

index( )

{

//回傳公告管理頁面

}

1. View ：owners.locations.posts.index

### 會員(房東)新增公告

1. 簡述：房東針對地點新增公告
2. 參與者：房東
3. 前提：進入房屋公告管理畫面的地點下，點擊新增按鈕
4. 成功事後狀態：新增成功
5. 失敗事後狀態：新增失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選新增按鈕 | 1. 回傳新增公告頁面 |
| 1. 輸入公告資料 | 1. 驗證資料 |
|  | 1. 將資料新增至資料庫儲存 |
|  | 1. 跳回房屋公告頁面 |

1. 活動圖

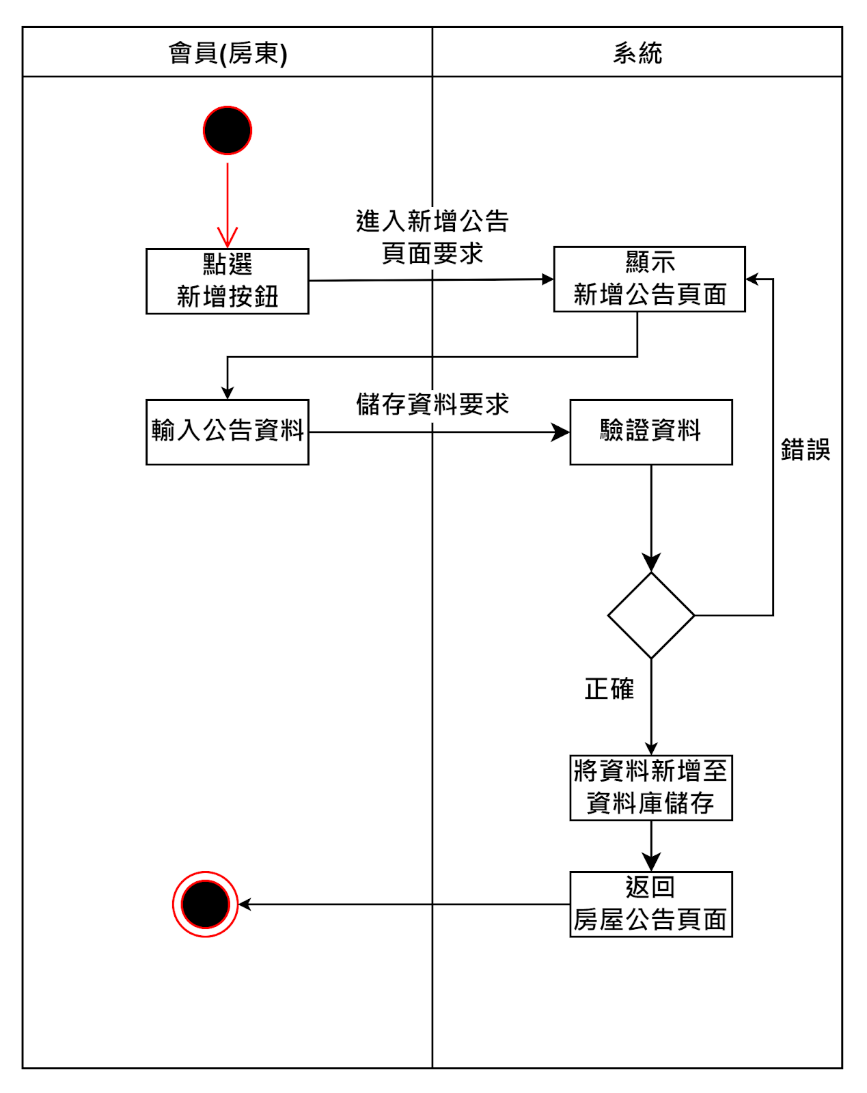


圖 31 會員(房東)新增公告

1. 路徑設計
2. 進入新增公告頁面要求
3. URL：owners/locations/{location}/posts/creat
4. Request method：get
5. Name：owners.locations.posts.create
6. Request Data：none
7. Model：Location, Post
8. Controller and its method：LocationPostController@create

create( )

{

//取得目前登入的房東資料，取得名下特定地點，回傳新增公告頁面

}

1. View ：owners.locations.posts.create
2. 儲存資料要求
3. URL：owners/locations/{location}/posts
4. Request method：post
5. Name：owners.locations.posts.store
6. Request Data：公告內容
7. Model：Location, Post
8. Controller and its method：LocationPostController@store

store(Request $request)

{

//在post的Model中新增公告資料，利用$request在頁面上存取的公告編號、詳細資料來儲存公告資料表的各欄位，資料儲存後，回到房屋公告頁面

}

1. View ：owners.locations.posts.index

### 會員(房東)修改公告

1. 簡述：房東針對房屋修改公告
2. 參與者：房東
3. 前提：進入房屋管理畫面
4. 成功事後狀態：修改成功
5. 失敗事後狀態：修改失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選編輯按鈕 | 1. 回傳編輯公告頁面 |
| 1. 輸入要修改的公告資料 | 1. 驗證資料 |
|  | 1. 將資料更新至資料庫儲存 |
|  | 1. 跳回房屋公告頁面 |

1. 活動圖

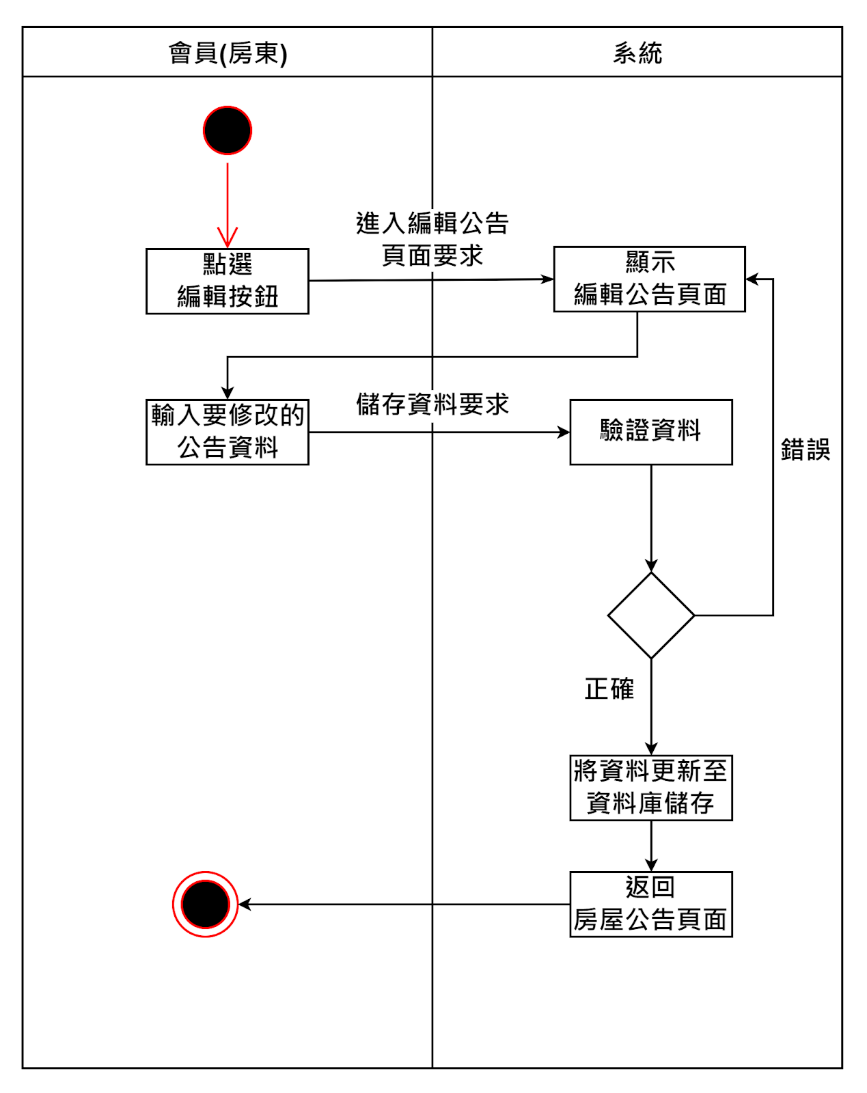


圖 32 會員(房東)修改公告

1. 路徑設計
2. 進入修改公告要求
3. URL：owners/locations/{location}/posts/{post}/edit
4. Request method：get
5. Name：owners.locations.posts.edit
6. Request Data：none
7. Model：Location, Post
8. Controller and its method：LocationPostController@edit

edit(Post $post)

{

//取得目前登入的房東資料，

取得名下特定地點，取得單一筆公告

進行修改

}

1. View ：owners.locations.posts.edit
2. 儲存公告要求
3. URL：owners/locations/{location}/posts/{post}
4. Request method：patch
5. Name：owners.locations.posts.update
6. Request Data：修改後的房屋資料
7. Model：Location, Post
8. Controller and its method：LocationPostController@update

update(Request $request,Post $post)

{

//找到要更新的公告資料$post，更新後跳轉到房屋公告頁面。

}

1. View ：owners.locations.posts.index

### 會員(房東)刪除公告

1. 簡述：房東針對地點刪除公告
2. 參與者：房東
3. 前提：進入房屋管理畫面，點擊刪除公告
4. 成功事後狀態：刪除成功
5. 失敗事後狀態：刪除失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選刪除按鈕 | 1. 跳出視窗，詢問確定刪除該公告嗎? |
| 1. 點選確認 | 1. 刪除該公告 |
|  | 1. 跳回房屋公告頁面 |

1. 活動圖



圖 33 會員(房東)刪除公告

1. 路徑設計
2. 刪除公告要求
3. URL：owners/locations/{location}/posts/{post}
4. Request method：delete
5. Name：owners.locations.posts.destroy
6. Request Data：none
7. Model：Location, Post
8. Controller and its method：LocationPostController@destroy

destroy (Post $post)

{

//刪除目前所選的公告

}

1. Redirect to：owners.locations.posts.index

### 會員(房東) 刊登房屋

1. 簡述：房東刊登房屋
2. 參與者：房東
3. 前提：在加入房屋畫面或編輯房屋畫面下
4. 成功事後狀態：刊登成功
5. 失敗事後狀態：刊登失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 輸入要刊登的房屋資料(房屋名稱、地址、設備、租金、特色、介紹) | 1. 驗證資料 |
|  | 1. 將資料更新至資料庫 |
|  | 1. 系統更改該房屋的狀態為已刊登，回傳刊登成功訊息 |
|  | 1. 跳回房屋管理頁面 |

1. 活動圖

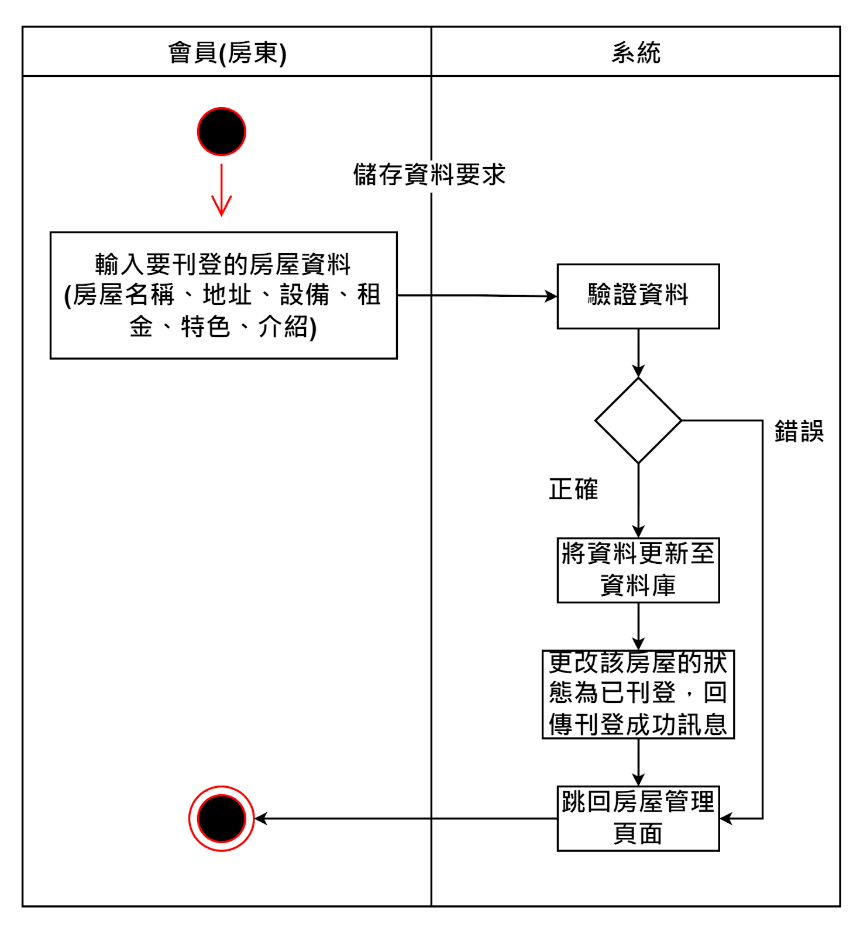


圖 34 會員(房東)刊登待出租房屋

1. 路徑設計
2. 刊登要求
3. URL：owners/locations/houses/publish/{publish}/edit
4. Request method：get
5. Name：owner.publish.edit
6. Request Data：狀態status為”刊登”
7. Model：House
8. Controller and its method：LocationHouseController@publish\_edit

publish\_edit (House $house)

{

//回傳刊登房屋表單頁面

}

1. View ：owner.publish.edit
2. 儲存刊登要求
3. URL：owners/locations/houses/publish/{publish}
4. Request method：patch
5. Name：owner.publish.update
6. Request Data：none
7. Model：House
8. Controller and its method：LocationHouseController@publish\_update

publish\_update (Request $request,House $house)

{

//將房屋中的是否刊登欄位更新為已刊登狀態

跳轉到房東租屋頁面。

}

1. Redirect to：owners.home.index

### 會員(房東)取消刊登房屋

1. 簡述：房東取消刊登房屋
2. 參與者：房東
3. 前提：已刊登房屋
4. 成功事後狀態：取消成功
5. 失敗事後狀態：取消失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 房東點擊取消刊登按鈕 | 1. 系統更改資料庫資訊 |
|  | 1. 系統更改該房屋的狀態為未刊登，回傳取消成功訊息 |
|  | 1. 跳回房屋管理頁面 |

1. 活動圖

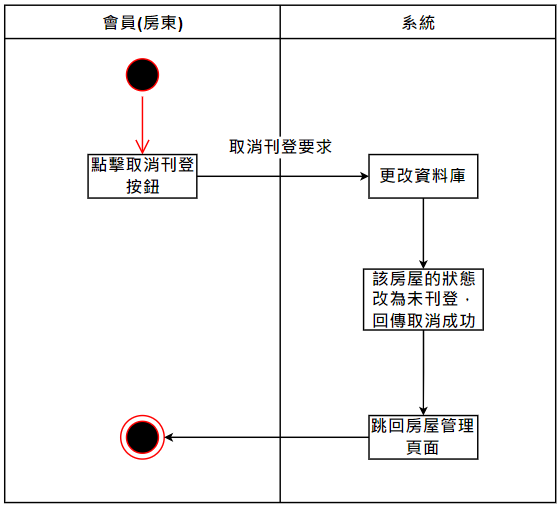


圖 35 會員(房東)取消刊登房屋

1. 路徑設計
2. 取消刊登要求
3. URL：locations/{location}/houses/{house}/states/update
4. Request method：patch
5. Name：locations.houses.states.unpublish
6. Request Data：狀態為出租或待定
7. Model：Location, House
8. Controller and its method：LocationHouseController@unpublish\_destroy

unpublish\_destroy (House $house)

{

//將房屋中的狀態欄位更新為未刊登狀態，跳轉到房屋管理頁面。

}

1. Redirect to：owners.home.index

### \*會員(房東)篩選某個狀態所有房屋資訊

1. 簡述：房東查看某狀態房屋資訊
2. 參與者：房東
3. 前提：已在房東租屋頁面，且選擇狀態篩選按鈕(已刊登 未刊登 已出租)
4. 成功事後狀態：查看成功
5. 失敗事後狀態：顯示錯誤訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 會員(房東)選擇狀態篩選按鈕 | 1. 系統顯示符合篩選的房屋頁面 |

1. 活動圖

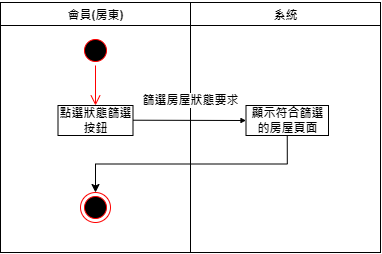


圖 36 會員(房東)查看某狀態房屋資訊

1. 路徑設計
2. 查看某狀態房屋資訊頁面要求
3. URL：houses/by\_status/{status}
4. Request method：get
5. Name：houses.by\_status
6. Request Data：none
7. Model：House
8. Controller and its method：HouseController@by\_status

by\_status($status)

{

//顯示被選擇的狀態房屋資訊頁面，選擇出租中就會顯示所有出租中的房屋

}

1. View ：houses.by\_status

### 會員(房東)查看單一房屋資訊

1. 簡述：房東查看單一房屋資訊
2. 參與者：房東
3. 前提：已在房東管理頁面，點選單個房屋
4. 成功事後狀態：查看成功
5. 失敗事後狀態：顯示錯誤訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選單一房屋 | 1. 系統顯示單一房屋資訊頁面 |

1. 活動圖

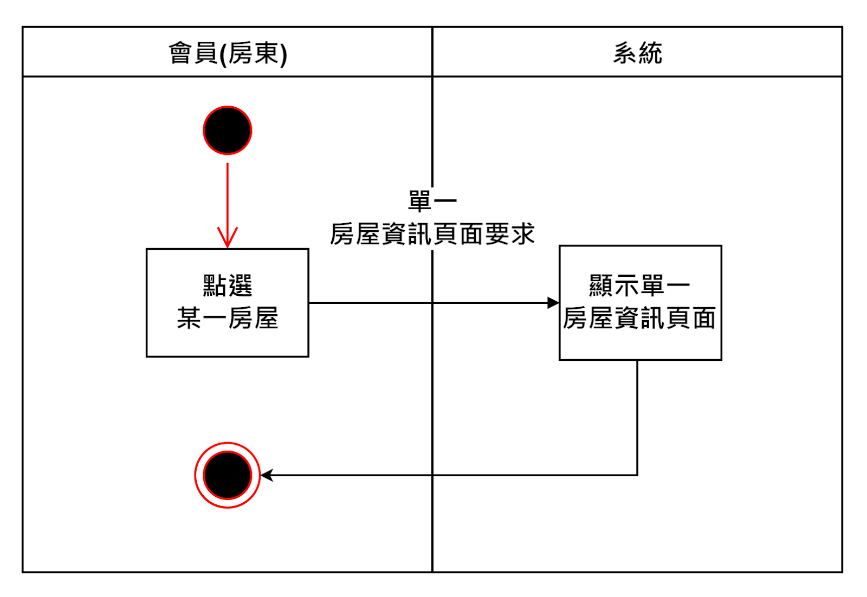


圖 37 會員(房東)查看刊登中房屋資訊

1. 路徑設計
2. 查看單一房屋資訊要求
3. URL：owners/houses/{house}
4. Request method：get
5. Name：owners.houses.show
6. Request Data：none
7. Model：House
8. Controller and its method：OwnerHouseController@show

show(House $house)

{

//取得目前登入的房東資料，取得名下得單一房屋資料，將資料傳入owners.houses.show

}

1. Redirect to：owners.houses.show

### 會員(房東)新增合約書

1. 簡述：房東新增房屋合約書
2. 參與者：房東
3. 前提：已進入房屋管理頁面
4. 成功事後狀態：新增成功
5. 失敗事後狀態：新增失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選新增合約 | 1. 回傳新增合約畫面 |
| 1. 輸入合約內容 | 1. 驗證資料 |
|  | 1. 新增至資料庫儲存 |
|  | 1. 跳回房屋管理畫面 |

1. 活動圖

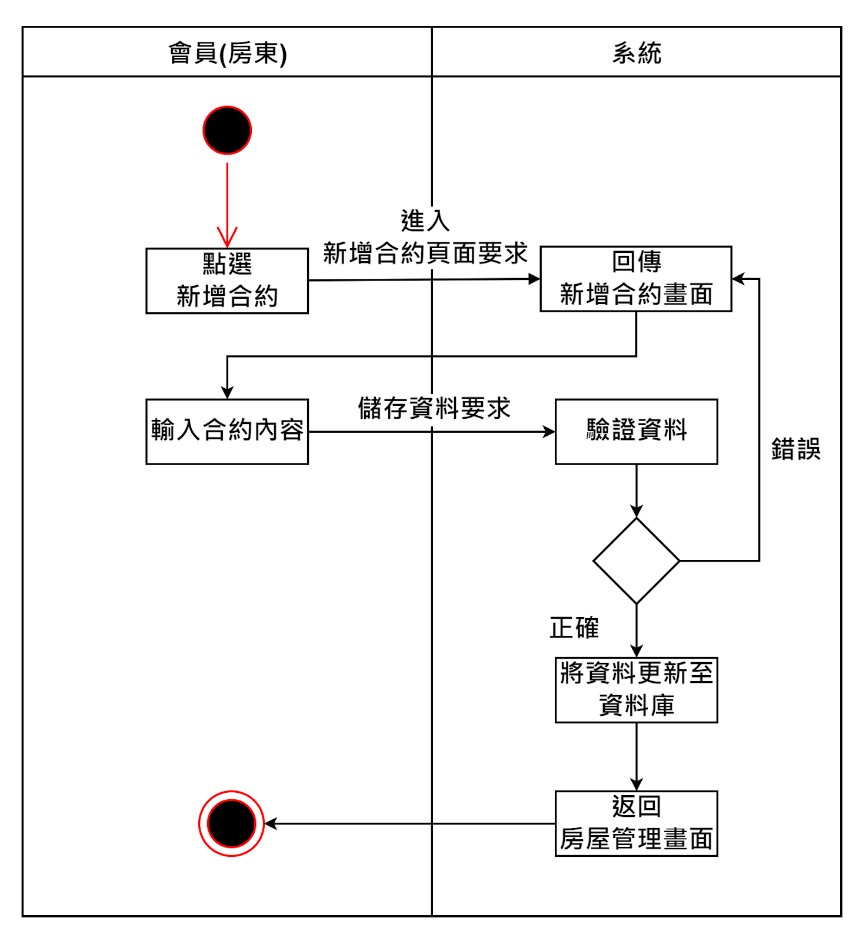


圖 38 會員(房東)新增合約書

1. 路徑設計
2. 進入新增合約頁面要求
3. URL：owners/houses/{house}/contracts/create
4. Request method：get
5. Name：houses.contracts.create
6. Request Data：none
7. Model：House, Contract
8. Controller and its method：HouseContractController@create

create( )

{

//取得目前登入的房東資料，取得名下得單一房屋資料，進行新增合約書

}

1. View：houses.contracts.create
2. 儲存合約資料要求
3. URL：owners/houses/{house}/contracts
4. Request method：post
5. Name：houses.contracts.store
6. Request Data：表單上新的合約資料
7. Model：House, Contract
8. Controller and its method：HouseContractController@store

contract\_store(Request $request)

{

//將要儲存的資料回傳到House儲存

}

1. Redirect To：houses.contracts.index

### 會員(房東)刪除合約書

1. 簡述：房東刪除房屋合約書
2. 參與者：房東
3. 前提：已進入房屋管理頁面
4. 成功事後狀態：刪除成功
5. 失敗事後狀態：刪除失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選刪除合約 | 1. 跳出確認訊息 |
| 1. 點選確認 | 1. 系統刪除該合約書 |
|  | 1. 跳回房屋管理畫面 |

1. 活動圖

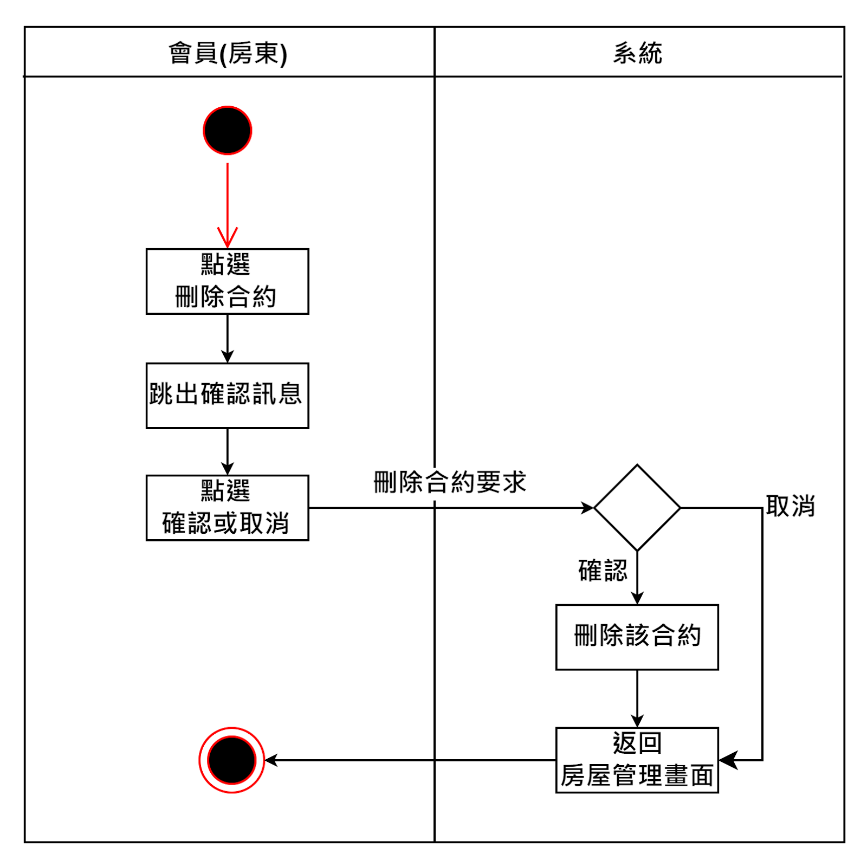


圖 39 會員(房東)刪除合約書

1. 路徑設計
2. 刪除合約書要求
3. URL：owners/houses/contracts/{contract}
4. Request method：delete
5. Name：houses.contracts.destroy
6. Request Data：none
7. Model：House, Contract
8. Controller and its method：HouseContractController@destroy

destroy (Contract $contract)

{

//刪除目前所選的合約

}

1. Redirect to：houses.contracts.index

### 會員(房東)查看合約書

1. 簡述：房東查看合約書
2. 參與者：房東
3. 前提：已進入房屋資訊頁面
4. 成功事後狀態：查看成功
5. 失敗事後狀態：查看失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選合約按鈕 | 1. 顯示房屋合約 |

1. 活動圖



圖 40 會員(房東)查看合約書

1. 路徑設計
2. 查看合約書要求
3. URL：owners/houses/contracts/{contract}
4. Request method：get
5. Name：houses.contracts.show
6. Request Data：none
7. Model：House, Contract
8. Controller and its method：HouseContractController@show

comtract\_show (House $house, Contract $contract)

{

//取得目前登入的房東資料，取得名下得單一房屋的合約書內容

將資料顯示在合約畫面

}

1. View：houses.contracts.show

### \*會員(房東)查看費用資訊

1. 簡述：房東新增該房屋費用資訊(水電費等)
2. 參與者：房東
3. 前提：已進入房東管理頁面中的某間房屋
4. 成功事後狀態：新增成功
5. 失敗事後狀態：新增失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 房東點選新增按鈕 | 1. 系統回傳新增費用資訊頁面 |
| 1. 房東輸入費用資料 | 1. 驗證資料 |
|  | 1. 將資料新增至資料庫 |
|  | 1. 系統跳回費用資訊頁面 |

1. 活動圖

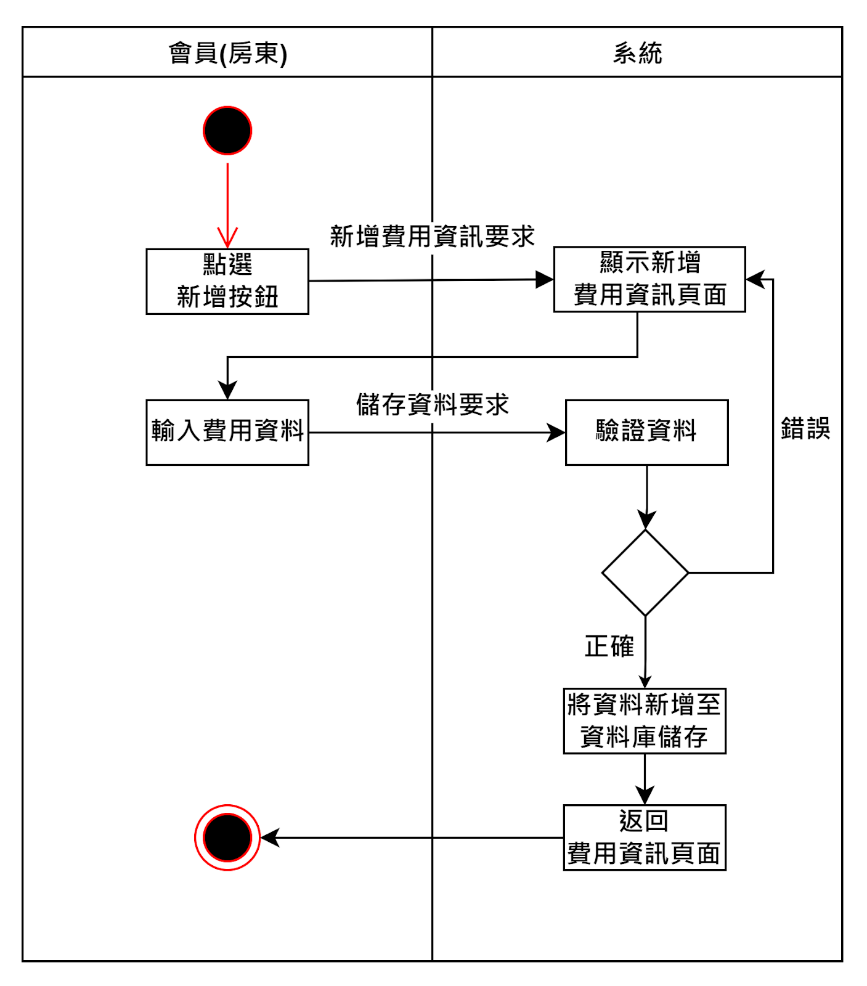


圖 41 會員(房東)新增費用資訊

1. 路徑設計
2. 新增費用資訊要求
3. URL：owners/houses/{house}/costs/create
4. Request method：get
5. Name：houses.costs.create
6. Request Data：none
7. Model：House, Cost
8. Controller and its method：HouseCostController@create

create()

{

//取得目前登入的房東資料，取得名下特定房屋，

新增費用資訊

}

1. View：houses.costs.create
2. 儲存資料要求
3. URL：owners/houses/{house}/costs
4. Request method：post
5. Name：houses.costs.store
6. Request Data：none
7. Model：House, Cost
8. Controller and its method：HouseCostController@store

store(Request $request)

{

//儲存費用資訊

}

1. Redirct to：houses.costs.index

### 會員(房東)新增費用資訊

1. 簡述：房東新增該房屋費用資訊(水電費等)
2. 參與者：房東
3. 前提：已進入房東管理頁面中的某間房屋
4. 成功事後狀態：新增成功
5. 失敗事後狀態：新增失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 房東點選新增按鈕 | 1. 系統回傳新增費用資訊頁面 |
| 1. 房東輸入費用資料 | 1. 驗證資料 |
|  | 1. 將資料新增至資料庫 |
|  | 1. 系統跳回費用資訊頁面 |

1. 活動圖

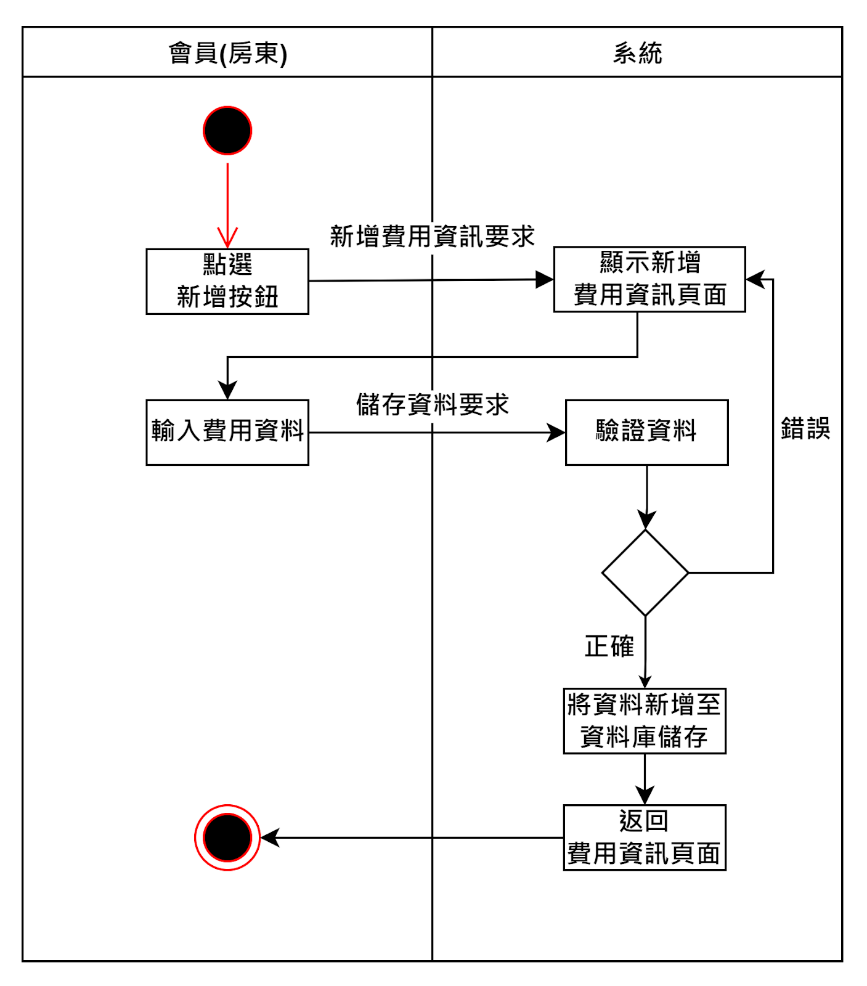


圖 41 會員(房東)新增費用資訊

1. 路徑設計
2. 新增費用資訊要求
3. URL：owners/houses/{house}/costs/create
4. Request method：get
5. Name：houses.costs.create
6. Request Data：none
7. Model：House, Cost
8. Controller and its method：HouseCostController@create

create()

{

//取得目前登入的房東資料，取得名下特定房屋，

新增費用資訊

}

1. View：houses.costs.create
2. 儲存資料要求
3. URL：owners/houses/{house}/costs
4. Request method：post
5. Name：houses.costs.store
6. Request Data：none
7. Model：House, Cost
8. Controller and its method：HouseCostController@store

store(Request $request)

{

//儲存費用資訊

}

1. Redirct to：houses.costs.index

### 會員(房東)修改費用資訊

1. 簡述：房東修改該房屋費用資訊(水電費等)
2. 參與者：房東
3. 前提：已進入房東管理頁面中的某間房屋
4. 成功事後狀態：修改成功
5. 失敗事後狀態：修改失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 房東點選修改按鈕 | 1. 系統回傳修改費用資訊頁面 |
| 1. 房東輸入要修改的費用資料 | 1. 驗證資料 |
|  | 1. 將資料更新至資料庫 |
|  | 1. 系統跳回費用資訊頁面 |

* 註解：驗證資料錯誤則需重新輸入費用資料。

1. 活動圖

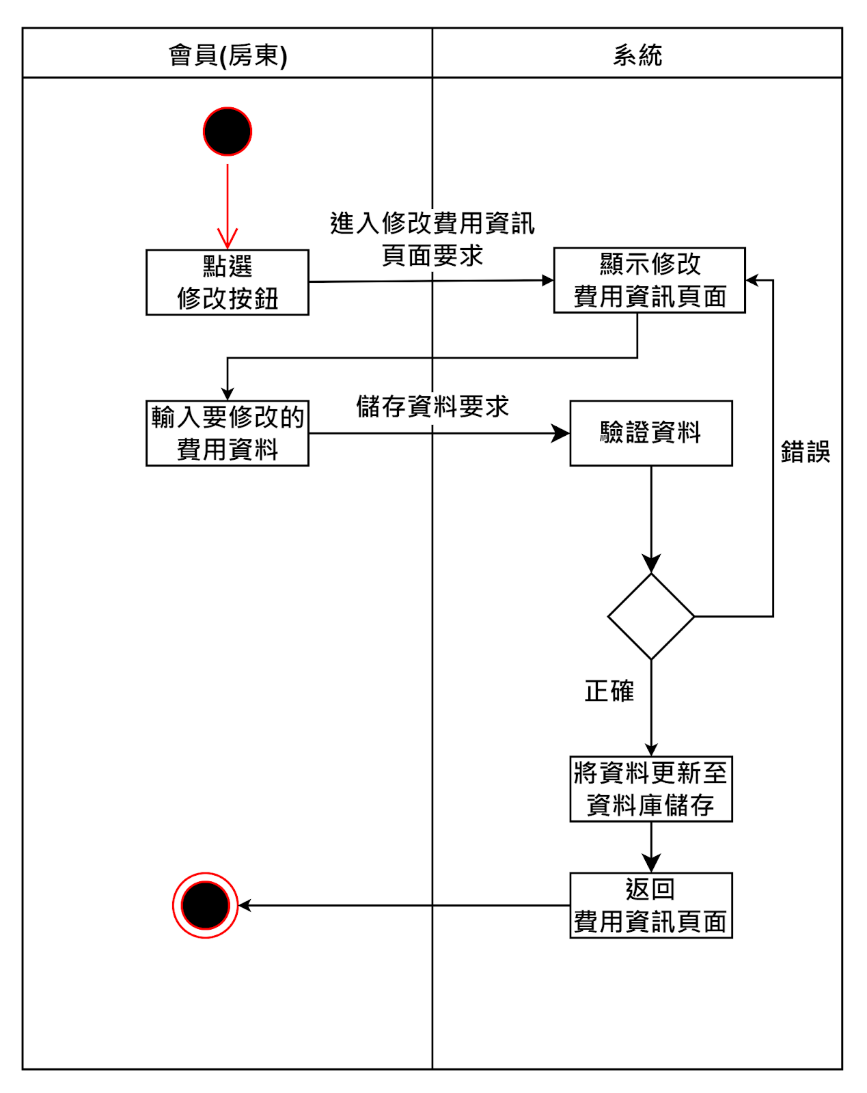


圖 42 會員(房東)修改費用資訊

1. 路徑設計
2. 進入修改費用資訊頁面要求
3. URL：owners/houses/costs/{cost}/edit
4. Request method：get
5. Name：houses.costs.edit
6. Request Data：none
7. Model：House, Cost
8. Controller and its method：HouseCostController@edit

edit(Cost $cost)

{

//取得目前登入的房東資料，取得名下特定房屋，修改費用

}

1. View：houses.costs.edit
2. 儲存資料要求
3. URL：owners/houses/costs/{cost}
4. Request method：patch
5. Name：houses.costs.update
6. Request Data：none
7. Model：House, Cost
8. Controller and its method：HouseCostController@update

update(Request $request,Cost $cost)

{

//儲存費用資訊

}

1. Redirect To：houses.costs.index

### 會員(房東)刪除費用資訊

1. 簡述：房東刪除該房屋費用資訊(水電費等)
2. 參與者：房東
3. 前提：已進入房東管理頁面中的某間房屋
4. 成功事後狀態：刪除成功
5. 失敗事後狀態：刪除失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選刪除按鈕 | 1. 跳出視窗，詢問確定刪除該費用資訊嗎? |
| 1. 點選確認 | 1. 刪除該費用資訊 |
|  | 1. 跳回費用資訊頁面 |

1. 活動圖

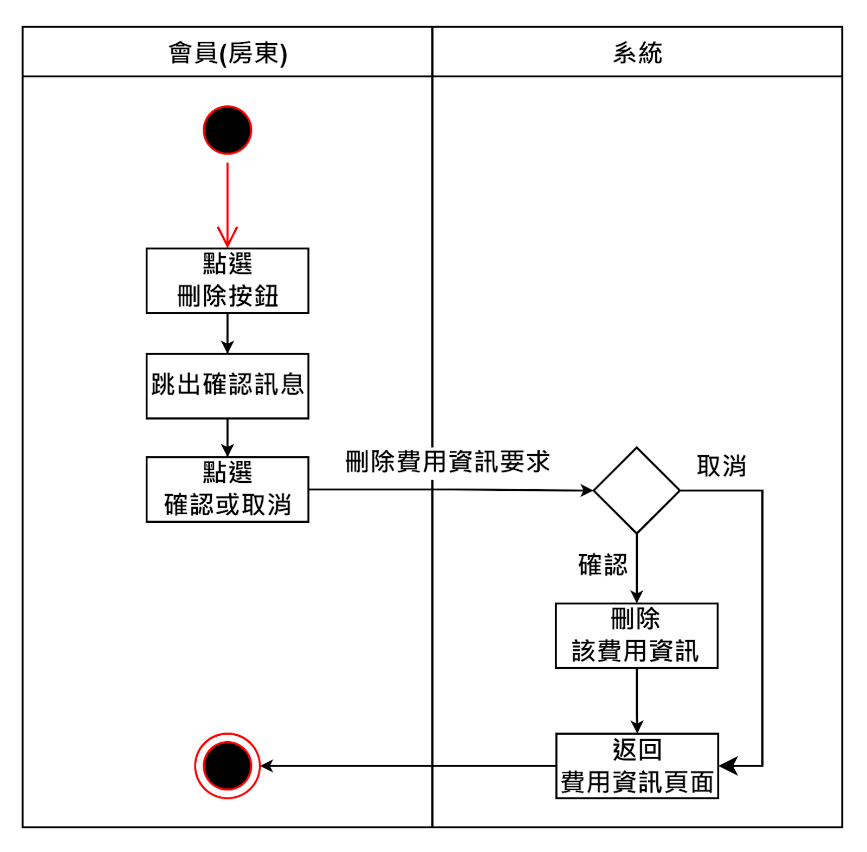


圖 43 會員(房東)刪除費用資訊

1. 路徑設計
2. 刪除費用資訊要求
3. URL：owners/houses/costs/{cost}
4. Request method：delete
5. Name：houses.costs.destroy
6. Request Data：none
7. Model：House, Cost
8. Controller and its method：HouseCostController@destroy cost\_destroy(Cost $cost)

{

//取得目前登入的房東資料，取得名下特定房屋，

取得該費用進行刪除費用

}

1. Redirect To：houses.costs.index

### 會員(房東)查看報修訊息

1. 簡述：房東查看報修訊息
2. 參與者：房東
3. 前提：已進入房屋管理頁面
4. 成功事後狀態：查看成功
5. 失敗事後狀態：查看失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 房東點選查看報修訊息 | 1. 系統回傳報修訊息畫面 |

* 註解：如沒有報修訊息，會顯示沒有報修訊息。

1. 活動圖

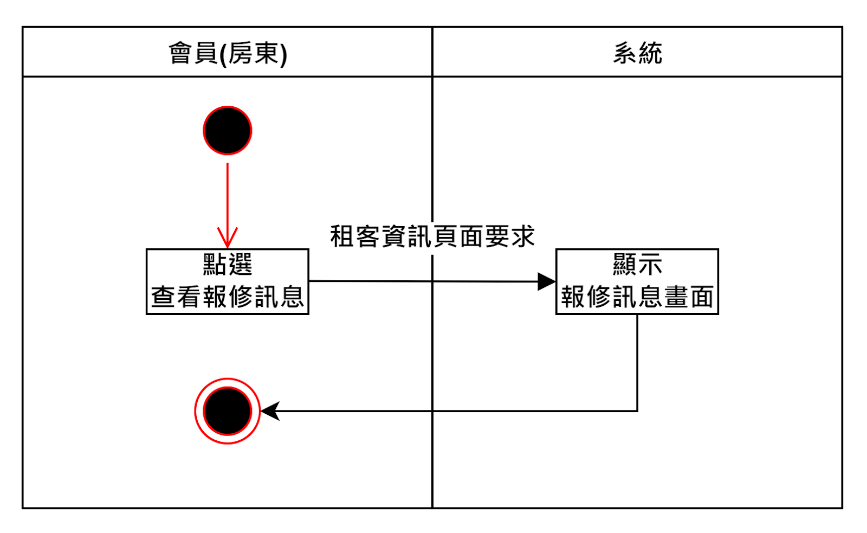


圖 44 會員(房東)查看報修訊息

1. 路徑設計
2. 顯示報修訊息要求
3. URL：owners/houses/{house}/repairs
4. Request method：get
5. Name：houses.repairs.index
6. Request Data：none
7. Model：House, Repair
8. Controller and its method：HouseRepairController@index

index ()

{

//取得目前登入的房東資料，取得名下特定房屋，顯示報修資料

}

1. View：houses.repairs.index

### 會員(房東)查看詳細報修訊息

1. 簡述：房東查看詳細報修訊息
2. 參與者：房東
3. 前提：已進入房屋管理的管理報修訊息畫面下
4. 成功事後狀態：查看成功
5. 失敗事後狀態：查看失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 房東點選單一報修訊息 | 1. 系統回傳詳細報修訊息畫面 |

1. 活動圖

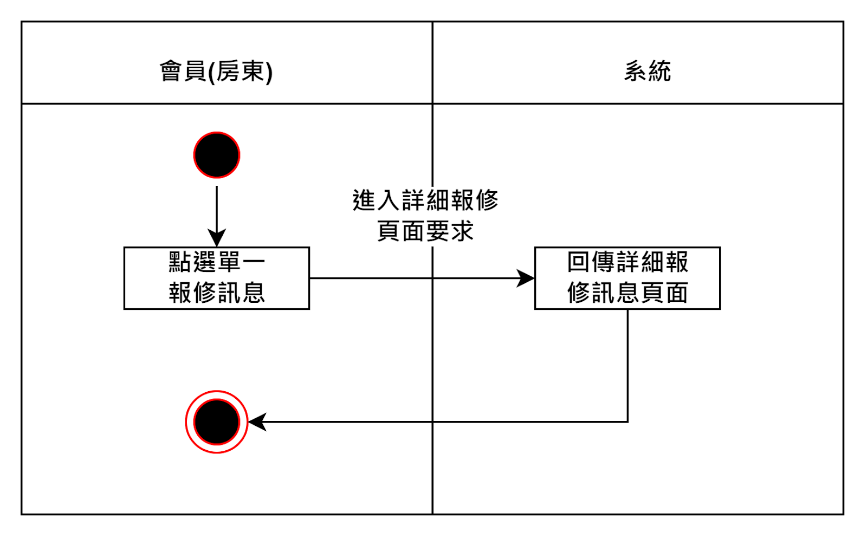


圖 45 會員(房東)查看詳細報修訊息

1. 路徑設計
2. 顯示詳細報修訊息要求
3. URL：owners/houses/repairs/{repair}
4. Request method：get
5. Name：houses.repairs.show
6. Request Data：none
7. Model：House
8. Controller and its method：HouseRepairController@show

show (Repair $repair)

{

//取得目前登入的房東資料，取得名下特定房屋，取得單一報修，顯詳細報修資料

}

1. View：houses.repairs.show

### 會員(房東)更新報修狀態

1. 簡述：更新報修物品狀態(未維修、已維修、已換新)
2. 參與者：房東
3. 前提：已進入房東管理頁面
4. 成功事後狀態：更新狀態成功
5. 失敗事後狀態：更新狀態失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 房東點選更新報修狀態 | 1. 系統跳出更新報修狀態頁面 |
| 1. 房東修改報修狀態 | 1. 驗證資料 |
|  | 1. 系統跳回報修訊息畫面 |

1. 活動圖

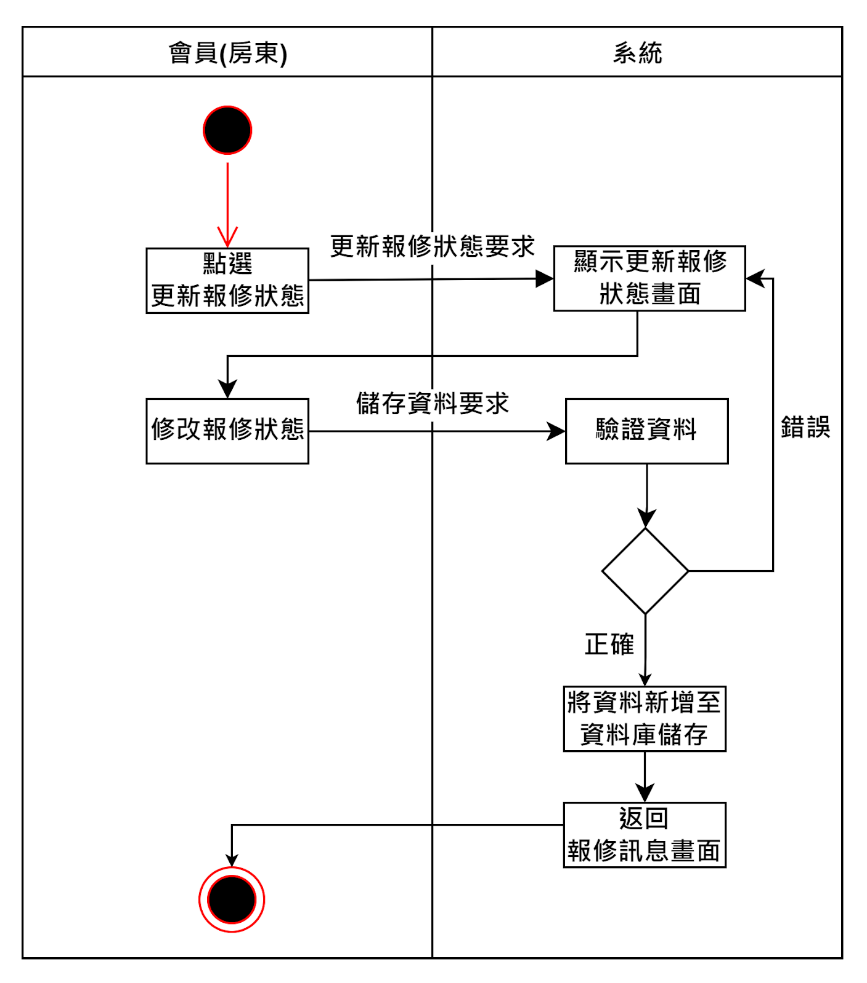


圖 46 會員(房東)更新報修狀態

1. 路徑設計
2. 儲存資料要求
3. URL：owners/houses/repairs/{repair}
4. Request method：patch
5. Name：houses.repairs.update
6. Request Data：none
7. Model：House, Repair
8. Controller and its method：HouseRepairController@update

update(Request $request,Repair $repair)

{

//儲存報修資料

}

1. Redirect To：houses.repairs.index

一張含有 桌 的圖片

自動產生的描述

## 3-10 會員(租客)相關使用案例分析與設計

### 會員(租客)查看公告

1. 簡述：租客查看公告
2. 參與者：租客
3. 前提：房東有上傳公告
4. 成功事後狀態：顯示公告
5. 失敗事後狀態：無公告內容
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 租客點選公告按鈕 | 1. 系統顯示公告頁面 |

1. 活動圖

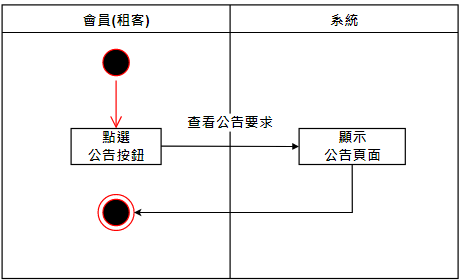


圖 47 會員(租客)查看公告

1. 路徑設計
2. 查看公告要求
3. URL：houses/posts
4. Request method：get
5. Name：houses.posts.index
6. Request Data：none
7. Model：House
8. Controller and its method：HousePostController@index

index(Post $post)

{

//取得目前登入的租客資料，顯示公告頁面

}

1. View：houses.posts.index

一張含有 文字 的圖片

自動產生的描述

### 會員(租客)查看合約

1. 簡述：租客查看合約
2. 參與者：租客
3. 前提：房東有上傳合約
4. 成功事後狀態：顯示合約
5. 失敗事後狀態：無合約內容
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 租客點選合約按鈕 | 1. 系統顯示合約頁面 |

1. 活動圖

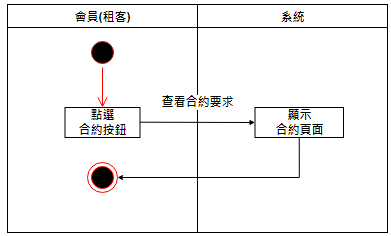


圖 48 會員(租客)查看合約

1. 路徑設計
2. 查看合約要求
3. URL：houses/contracts/{contract}
4. Request method：get
5. Name：houses.contracts.show
6. Request Data：none
7. Model：House, Contract
8. Controller and its method：HouseContractController@show

show(Contract $contract)

{

//取得目前登入的租客資料，顯示合約頁面

}

1. View：houses.contract.show

### 會員(租客)查看費用資訊

1. 簡述：租客查看費用資訊，包含水費、電費、保證金、租金、管理費等
2. 參與者：租客
3. 前提：租客已簽約進入房屋
4. 成功事後狀態：顯示費用資訊
5. 失敗事後狀態：顯示失敗訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選查看費用 | 1. 系統顯示費用總覽畫面 |

1. 活動圖

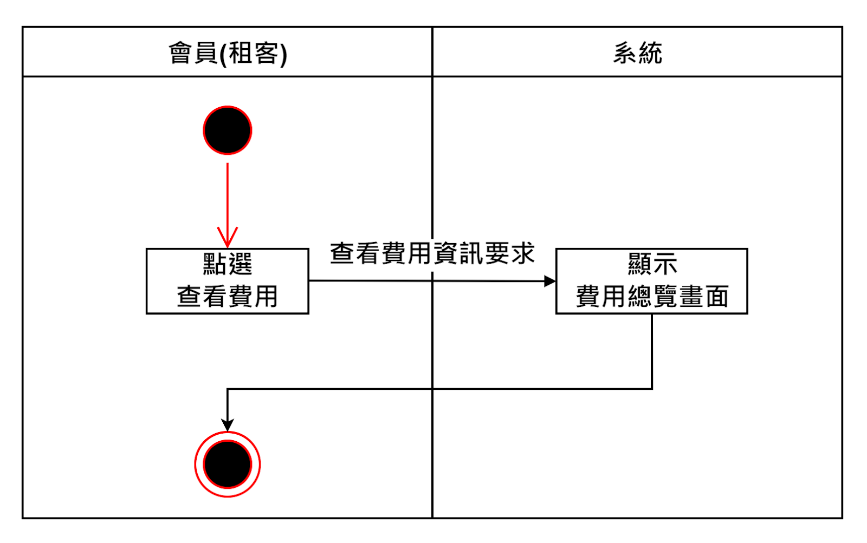


圖 49 會員(租客)查看費用資訊

1. 路徑設計
2. 顯示進階搜尋表單要求
3. URL：renters/costs
4. Request method：get
5. Name：renters.costs.index
6. Request Data：none
7. Model：House, Cost
8. Controller and its method：HouseCostController@index

index(Cost $cost)

{

//顯示費用資訊頁面

}

1. View：renters.costs.index

一張含有 桌 的圖片

自動產生的描述

### 會員(租客)進行繳費

1. 簡述：租客進行繳費
2. 參與者：租客
3. 前提：有未繳費用
4. 成功事後狀態：繳費成功訊息
5. 失敗事後狀態：繳費失敗訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選繳費按鈕 | 1. 顯示繳費資訊頁面 |
| 1. 填寫繳費資訊 | 1. 驗證資料 |
|  | 1. 顯示繳費成功訊息 |
|  | 1. 回到費用總覽畫面 |

1. 活動圖

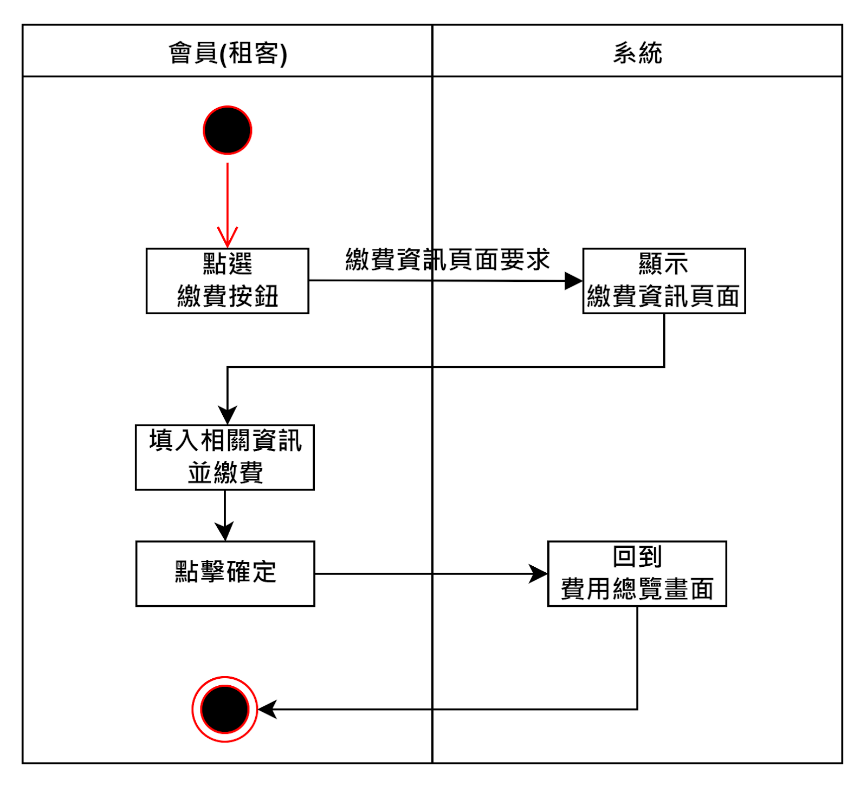


圖 50 會員(租客)進行繳費

1. 路徑設計
2. 繳費要求
3. URL：pays/create
4. Request method：get
5. Name：pays.create
6. Request Data：none
7. Model：Cost
8. Controller and its method：CostController@create

create()

{

//跳出付款相關資訊的頁面，可選擇、填寫付款方式

}

1. View：pays.create
2. 提交繳費
3. URL：pays
4. Request method：post
5. Name：pays.store
6. Request Data：繳費資料
7. Model：Cost , CostDetail
8. Controller and its method：CostController@store

store(Request $request)

{

//繳費完成，儲存繳費資料

}

1. Redirect to：pays.show

### 會員(租客)查看報修訊息

1. 簡述：租客查看報修訊息
2. 參與者：租客
3. 前提：已進入租客管理頁面，點選報修按鈕
4. 成功事後狀態：查看成功
5. 失敗事後狀態：查看失敗
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選報修按鈕 | 1. 系統回傳報修訊息畫面 |

1. 活動圖

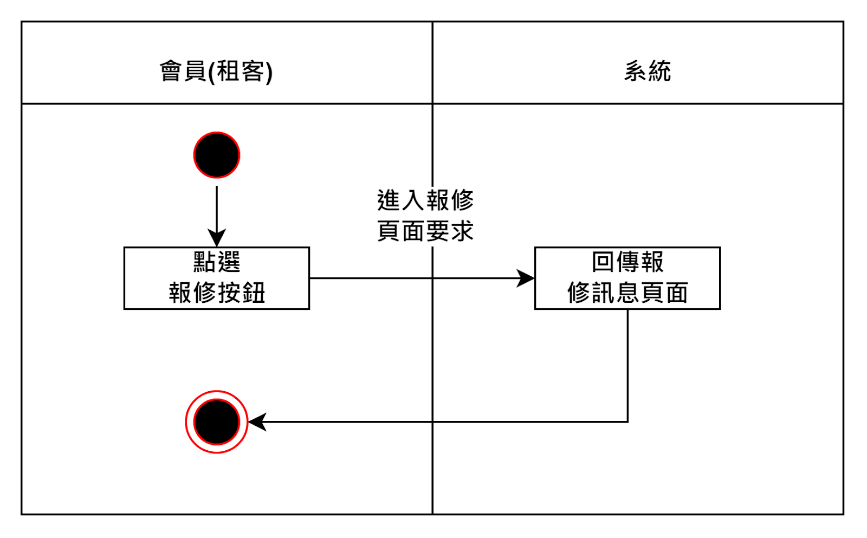


圖 51 會員(租客)查看報修訊息

1. 路徑設計
2. 報修要求
3. URL：renters/repairs
4. Request method：get
5. Name：renters.houses.repairs.index
6. Request Data：none
7. Model：House, Repair
8. Controller and its method：HouseRepairController@index

index (Repair $repair)

{

//取得目前登入的房東資料，取得名下特定房屋，取得單一報修，顯示詳細報修資料

}

1. View：renters.repairs.index

### 會員(租客)新增報修訊息

1. 簡述：租客新增報修訊息
2. 參與者：租客
3. 前提：租客已簽約進入房屋
4. 成功事後狀態：新增成功訊息
5. 失敗事後狀態：新增失敗訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選新增按鈕 | 1. 回傳報修表單 |
| 1. 填寫報修內容並送出 | 1. 顯示儲存內容 |
| 1. 點選確認按鈕 | 1. 將資料新增至資料庫儲存，且顯示報修成功訊息 |
|  | 1. 返回租客管理頁面 |

1. 活動圖

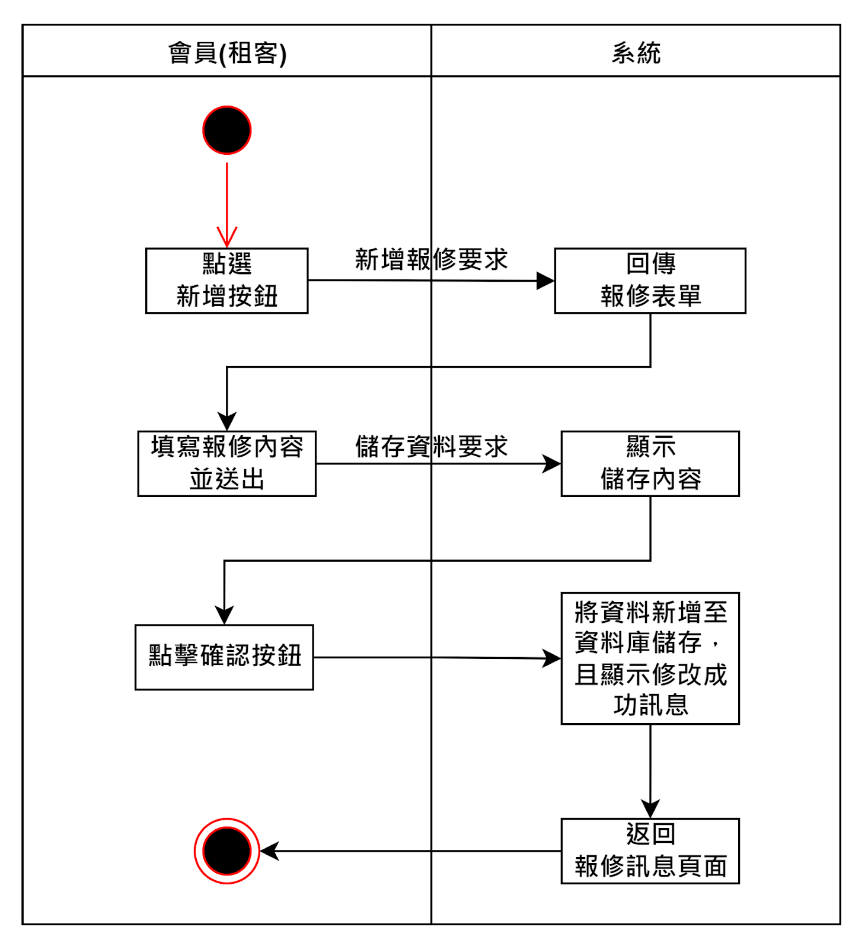


圖 52 會員(租客)新增報修訊息

1. 路徑設計
2. 新增報修要求
3. URL：renters/repairs/create
4. Request method：get
5. Name：renters.repairs.create
6. Request Data：none
7. Model：House, Repair
8. Controller and its method：HouseRepairController@create

create()

{

//顯示新增報修表單

}

1. View：renters.repairs.create
2. 儲存資料要求
3. URL：renters/repairs
4. Request method：post
5. Name：renters.repairs.store
6. Request Data：none
7. Model：House, Repair
8. Controller and its method：HouseRepairController@store

store(Request $request)

{

//將表單所輸入的資料儲存是資料庫

}

1. View：renters.repairs.index

### 會員(租客)修改報修訊息

1. 簡述：租客修改報修訊息
2. 參與者：租客
3. 前提：租客已簽約進入房屋
4. 成功事後狀態：修改成功訊息
5. 失敗事後狀態：修改失敗訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選編輯按鈕 | 1. 回傳修改報修表單 |
| 1. 填寫要修改的報修內容並送出 | 1. 顯示儲存內容 |
| 1. 點選確認按鈕 | 1. 將資料更新至資料庫儲存，且顯示修改成功訊息 |
|  | 1. 返回租客管理頁面 |

1. 活動圖

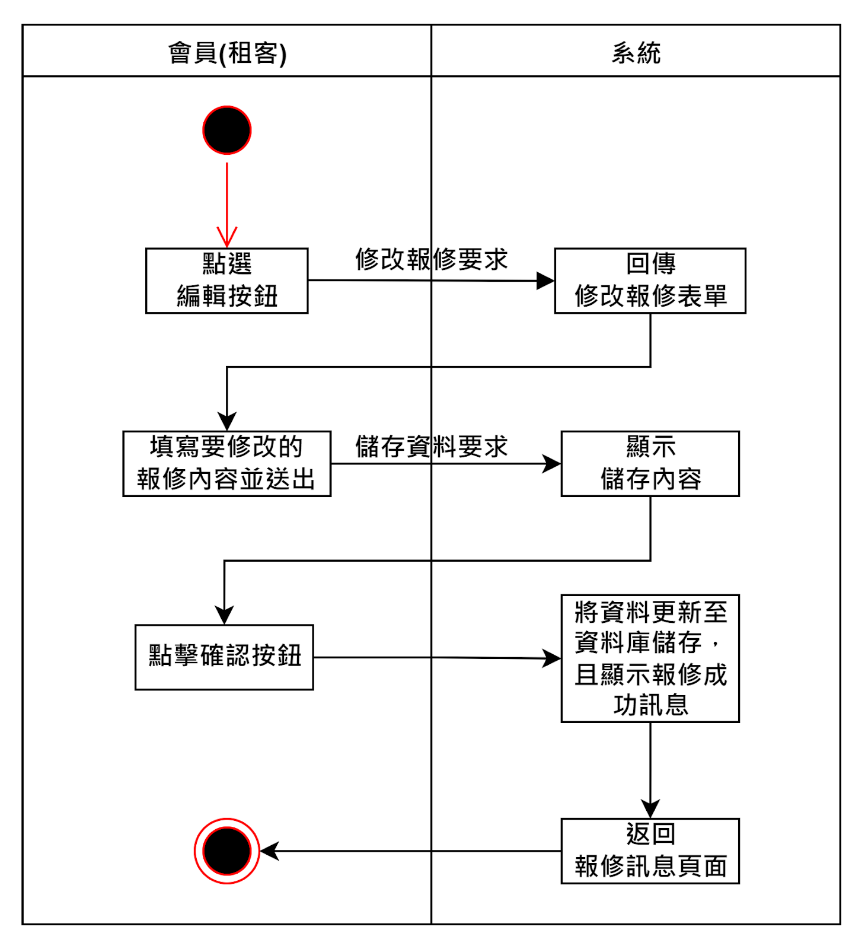


圖 53 會員(租客)修改報修訊息

1. 路徑設計
2. 修改報修要求
3. URL：renters/repairs/{repair}/edit
4. Request method：get
5. Name：renters.repairs.edit
6. Request Data：none
7. Model：House, Repair
8. Controller and its method：HouseRepairController@edit

edit(Repair $repair)

{

//顯示修改報修表單

}

1. View：renters.repairs.edit
2. 儲存資料要求
3. URL：renters/repairs/{repair}
4. Request method：patch
5. Name：renters.repairs.store
6. Request Data：none
7. Model：House, Repair
8. Controller and its method：HouseRepairController@update

update(Request $request,Repair $repair)

{

//更新要修改的報修表單，並跳回報修頁面

}

1. View：renters.repairs.index

### 會員(租客)刪除報修訊息

1. 簡述：租客刪除報修訊息
2. 參與者：租客
3. 前提：租客已簽約進入房屋
4. 成功事後狀態：刪除成功訊息
5. 失敗事後狀態：刪除失敗訊息
6. 事件路徑

基本路徑

|  |  |
| --- | --- |
| **使用者動作** | **系統回應** |
| 1. 點選刪除按鈕 | 1. 跳出視窗，詢問確定刪除該報修訊息嗎? |
| 1. 點選確認 | 1. 刪除該報修訊息 |
|  | 1. 返回租客管理頁面 |

1. 活動圖

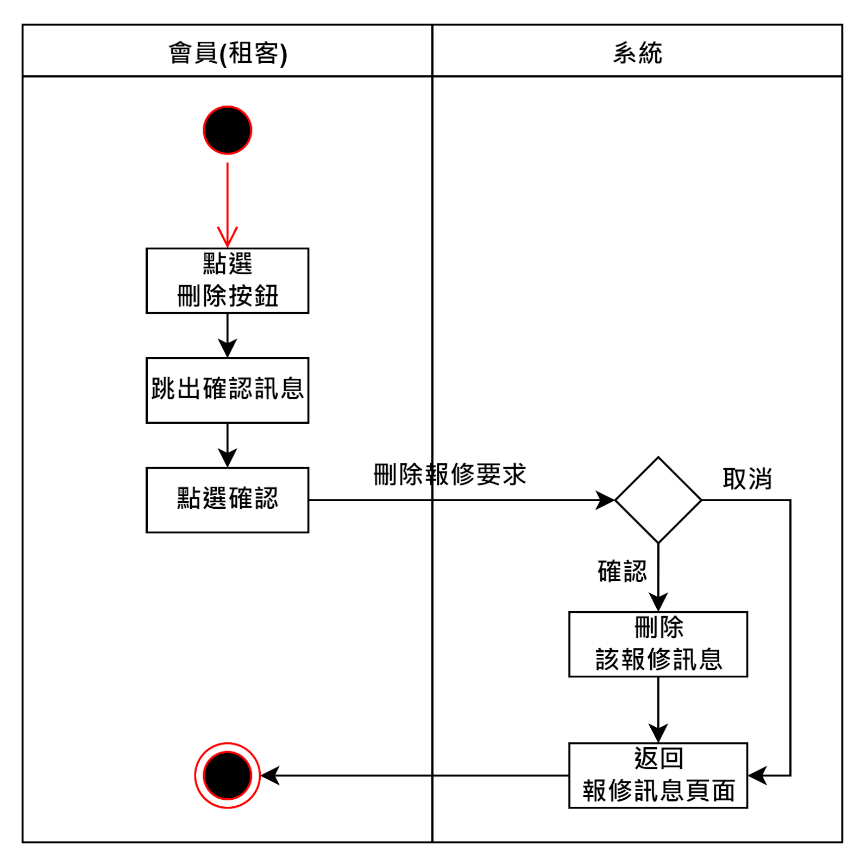


圖 54 會員(租客)刪除報修訊息

1. 路徑設計
2. 刪除報修要求
3. URL：renters/repairs/{repair}
4. Request method：delete
5. Name：renters.repairs.destroy
6. Request Data：none
7. Model：House, Repair
8. Controller and its method：HouseRepairController@destroy

destroy(Repair $repair)

{

//刪除所選的報修要求

}

1. Redirect to：renters.repairs.index

## 3-11 專案規劃

本專題將系統規劃成兩個Laravel專案，分別為二手物交易前台、二手物交易管理後台。

* 二手物交易前台
* 主要使用者：訪客、會員、賣家
* 主要功能：
* 訪客可辦理註冊成為本網站的會員，成為會員後可進行二手物交易，並查詢訂單資訊。
* 賣家可向平台方申請商品品質檢測，提高該二手物的價值，也可在賣家後台查詢營運狀態等資訊。
* 額外套用的套件或Bootstrap樣板：
* Cozastore模板
* 作用：本網站訪客、會員套入此模板。
* 來源：<https://themewagon.com/themes/free-html5-ecommerce-website-template/>
* 套用步驟：

Step1.下載該模板並解壓縮。

Step2.在Laravel專案中新增訪客會員版面的Layout檔。

Step3.將Step1中的模板程式碼引入Step2的Layout檔。

Step4.在所有訪客會員相關的View中繼承該Layout檔。

* SB Admin 2模板。
* 作用：本網站賣家套入此模板。
* 來源：https://startbootstrap.com/theme/sb-admin-2
* 套用步驟：

Step1.下載該模板並解壓縮。

Step2.在Laravel專案中新增賣家版面的Layout檔。

Step3.將Step1中的模板程式碼引入Step2的Layout檔。

Step4.在所有賣家相關的View中繼承該Layout檔。

* 二手物交易管理後台：
* 主要使用者：平台人員。
* 主要功能：可以進行審核賣家申請單和檢測人員排班以及進行檢測與二手物類別管理和品質檢測項目之動作。
* 額外套用的套件或Bootstrap樣板：
* DASHMIN模板。
* 作用：本網站平台人員頁面套入此模板。
* 來源：<https://themewagon.com/themes/dashmin-responsive-free-bootstrap-5-html5-admin-dashboard-template/>
* 套用步驟：

Step1.下載該模板並解壓縮。

Step2.在Laravel專案中新增平台人員版面的Layout檔。

Step3.將Step1中的模板程式碼引入Step2的Layout檔。

Step4.在所有平台人員相關的View中繼承該Layout檔。

## 3-12 系統運作環境

1. **使用者系統運作環境**

本專題將使用者分成訪客與會員、賣家、平台人員來說明使用者的系統運作環境。

**1. 訪客與會員**

* + 硬體：個人電腦一台。
  + 軟體：

1. Microsoft Windows 7以上之平台。
2. Browser：IE、Chrome等瀏覽器。

**2. 賣家**

* + 硬體：個人電腦一台，網路攝影機。
  + 軟體：

1. Microsoft Windows 7以上之平台。
2. Browser：IE、Chrome等瀏覽器。

**3. 平台人員**

* + 硬體：個人電腦一台。
  + 軟體：

1. Microsoft Windows 7以上之平台。
2. Browser：IE、Chrome等瀏覽器。

二、伺服器端系統運作環境

* 硬體：個人電腦，最好是伺服器等級的個人電腦。
* 軟體：

1. Microsoft Windows 7以上，最好是Microsoft Windows Server或是Linux。
2. Apache或Nginx。
3. MySQL 5.7以上。
4. PHP 7.3以上，且需安裝有適當擴充擴充功能(extensions)，可參考Laravel文件[6]。
5. 上述三項軟體，若在Windows環境可以直接安裝WAMP或WNMP，若在Llinx環境可以直接安裝LAMP或LNMP。

## 3-13 系統開發工具

本專題使用了Laravel、Wagon、SourceTree、GitHub、PhpStorm等工具來開發系統，以下分別說明其主要功用。

* Laravel：PHP程式語言中最多人使用的框架。
* Wagon：含UwAmp、Git及Composer等軟體。
  + - UwAMP：含Apache、MySQL和PHP等軟體。
    - Git：版本管理軟體。
    - Composer：PHP套件管理軟體。
* SourceTree：圖形化的版本管理軟體。
* GitHub：協助版本管理網站。
* PhpStorm：PHP/Laravel程式開發工具。

# 進度與工作分配

## 一、完成專題分配

1. 專題期中規劃書(系統分析與設計)部分:

* 第一章 緒論

1-1 專題背景與動機: 徐千素、賴妤典、曾翊晴、鄭羽庭、黃佳怡

1-2 專題目的: 徐千素、賴妤典、曾翊晴、鄭羽庭、黃佳怡

1-3 專題預期成效: 徐千素、賴妤典、曾翊晴、鄭羽庭、黃佳怡

* 第二章 技術探討
* 第三章 系統分析

3-1 需求描述(訪客、會員): 徐千素

3-1 需求描述(會員(房東/租客)): 賴妤典

3-1 需求描述(房東): 曾翊晴、鄭羽庭

3-1 需求描述(租客): 黃佳怡

3-2 事件分析(訪客、會員): 徐千素

3-2 事件分析(會員(房東/租客)): 賴妤典

3-2 事件分析(房東): 曾翊晴、鄭羽庭

3-2 事件分析(租客): 黃佳怡

3-3 系統使用案例分析(訪客、會員): 徐千素、賴妤典

3-3 系統使用案例分析(房東): 曾翊晴、鄭羽庭

3-3 系統使用案例分析(租客): 黃佳怡

3-4 實體關聯圖: 鄭羽庭

3-5 關聯式資料庫綱要圖: 鄭羽庭

3-6 資料庫欄位設計: 鄭羽庭

3-7 訪客、會員相關使用案例分析與設計: 徐千素

3-8 會員(房東/租客)相關使用案例分析與設計: 賴妤典

3-9 房東相關使用案例分析與設計: 曾翊晴、鄭羽庭

3-10 租客相關使用案例分析與設計: 黃佳怡

1. 系統開發部分(待調整):

* 房東新增租客簽約流程

1. 專題報告書及其他

## 二、專題進度規劃

2/20~5/31，共15 周

1. 資料庫&路由建置，訪客、租客、房東版面設計
2. 房東新增、修改、刪除地點
3. 房東新增、修改、刪除房屋
4. 房東查看、刊登、取消刊登、篩選房屋
5. 訪客/會員查詢(篩選)出租房屋
6. 房東新增、刪除、查看合約書；租客查看合約，會員新增、修改轉帳資訊
7. 房東新增、修改、刪除費用資訊；租客查看費用資訊
8. (可能會做比較久):租客進行繳費
9. 房東查看、新增、修改、刪除公告和信件代收；租客查看公告和信件
10. 租客查看、新增、修改、刪除報修訊息；房東查看詳細報修訊息、更新報修狀態
11. 房東查看、新增、刪除管理員
12. 會員修改會員資料、登出
13. 系統功能最終修改、報告書最終修改
14. 系統功能最終修改、報告書最終修改
15. 系統功能最終修改、報告書最終修改