# [登入]

## 先需登入方能使用權限。

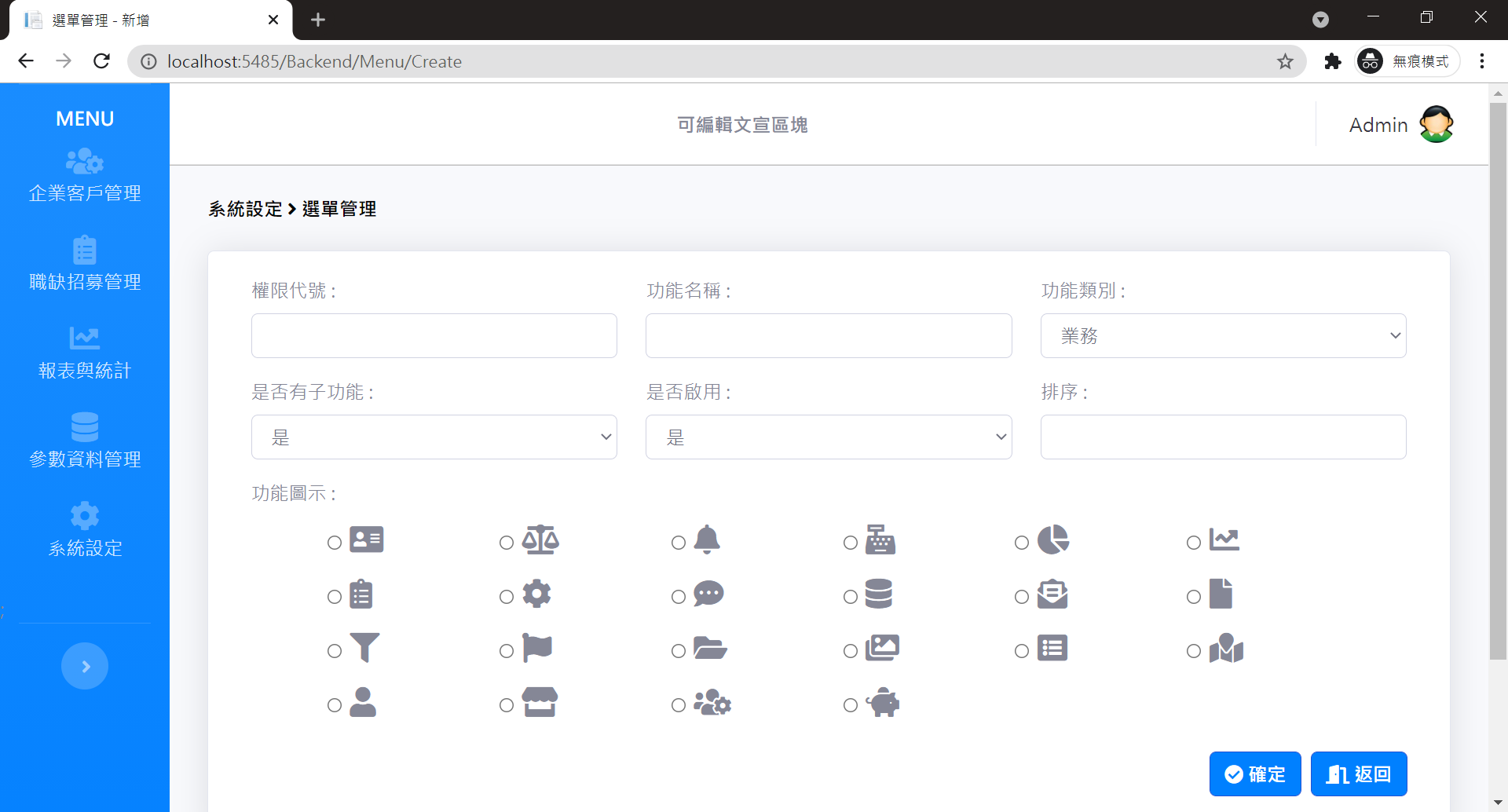
前台: <http://localhost:5485/Home/LoginChoice>

後台: <http://localhost:5485/Backend/Home/AdminLogin>

# [選單管理]

## 新增主目錄

第一層設定功能圖示、功能類別，若功能圖示沒有符合的可以到參數設定(1.4節補充)做新增圖示。



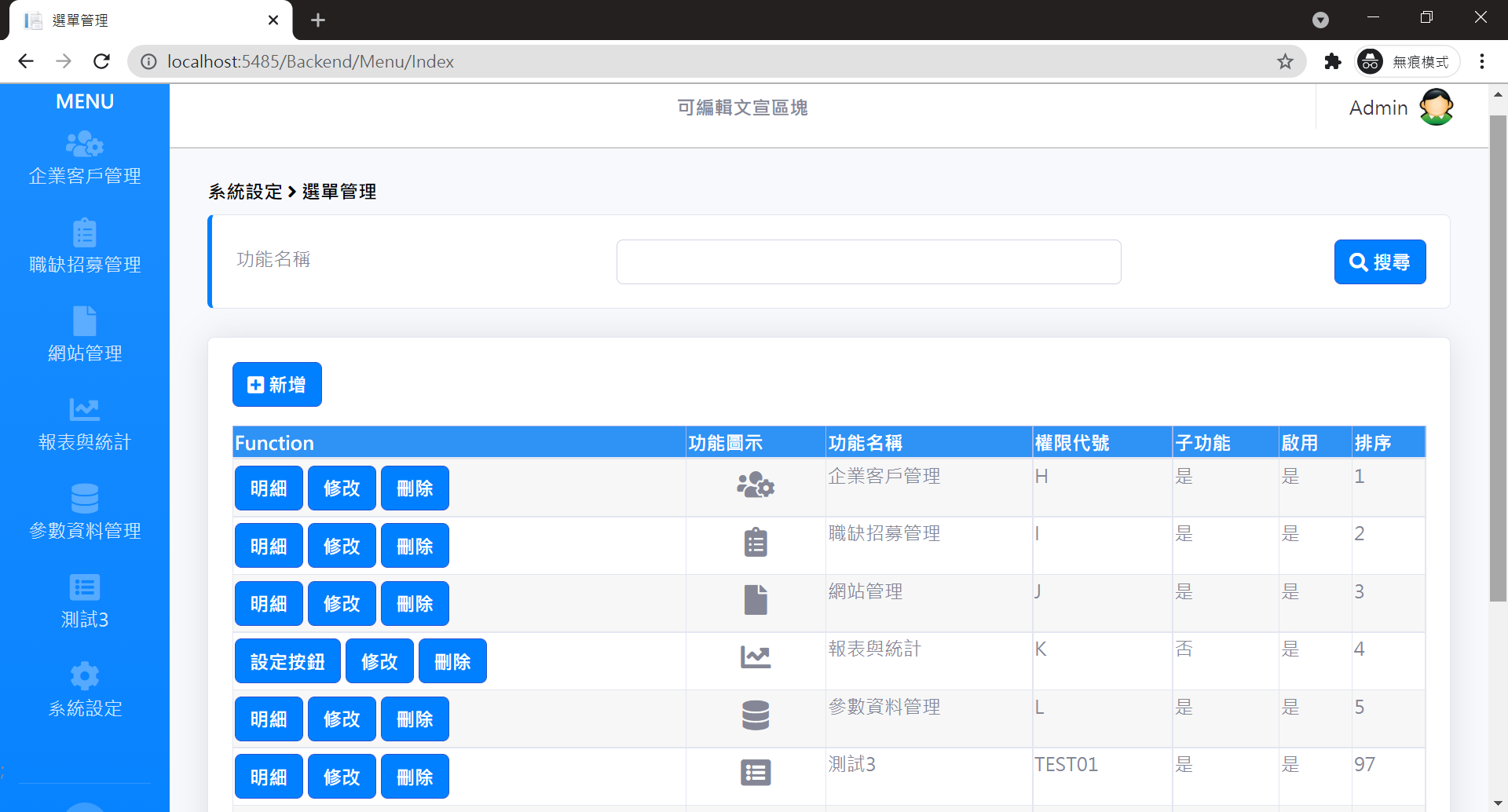
若有子功能，則可繼續往下一層新增(至多三層)；若無，則須設定Controller及Action。



## 設定第二層及第三層

點擊明細可進入第二層或第三層，同上述步驟，但不需設定功能圖示、功能類別。

注意，若這裡有需求是顯示在目錄上但是為同一支controller，請在Action後面加上Index，參考參數資料管理。

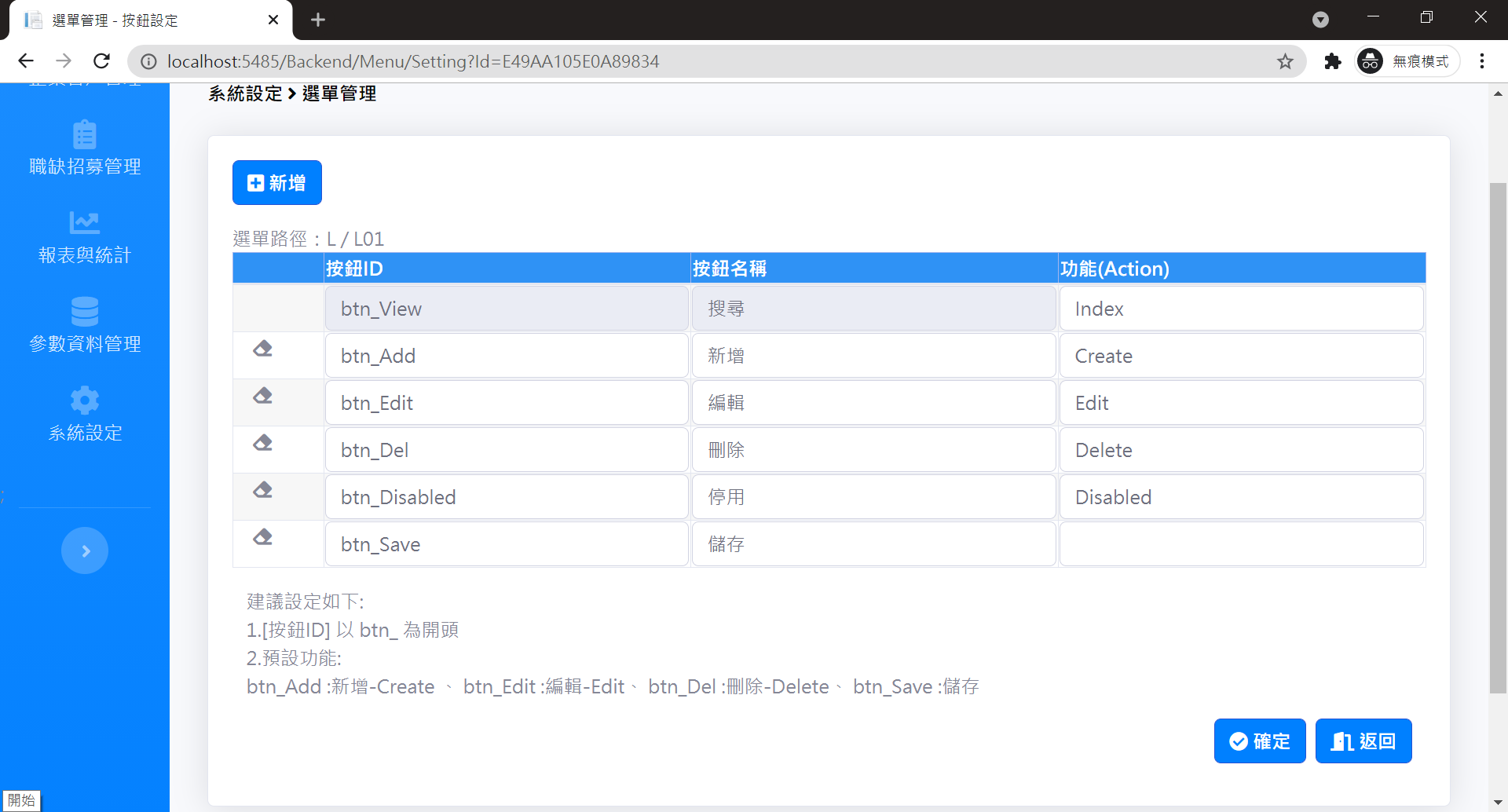


## 設定按鈕

\*按鈕控制：[此專案無該需求，僅需使用預設設定到btn\_View部分即可，無需設定以下功能。]\*

若無子功能，即可設定按鈕，進入畫面後預先設定**btn\_View**的功能，Action預設**Index**，這裡注意，**若無此功能按鈕則可能導致於權限設定中無法存取該頁面的狀況**。

另外，為求統一，建議設定按鈕ID及功能如下面說明。



## Button設定

在已控制權限的Controller下的View，按鈕需加上name為上面設定的對應按鈕ID，並加上authbtn屬性，例如name="btn\_Add" authbtn、name="btn\_Edit" authbtn、name="btn\_Del" authbtn等。

設定完成後應會看到若在第2.1節中無打勾的按鈕則不會顯示。



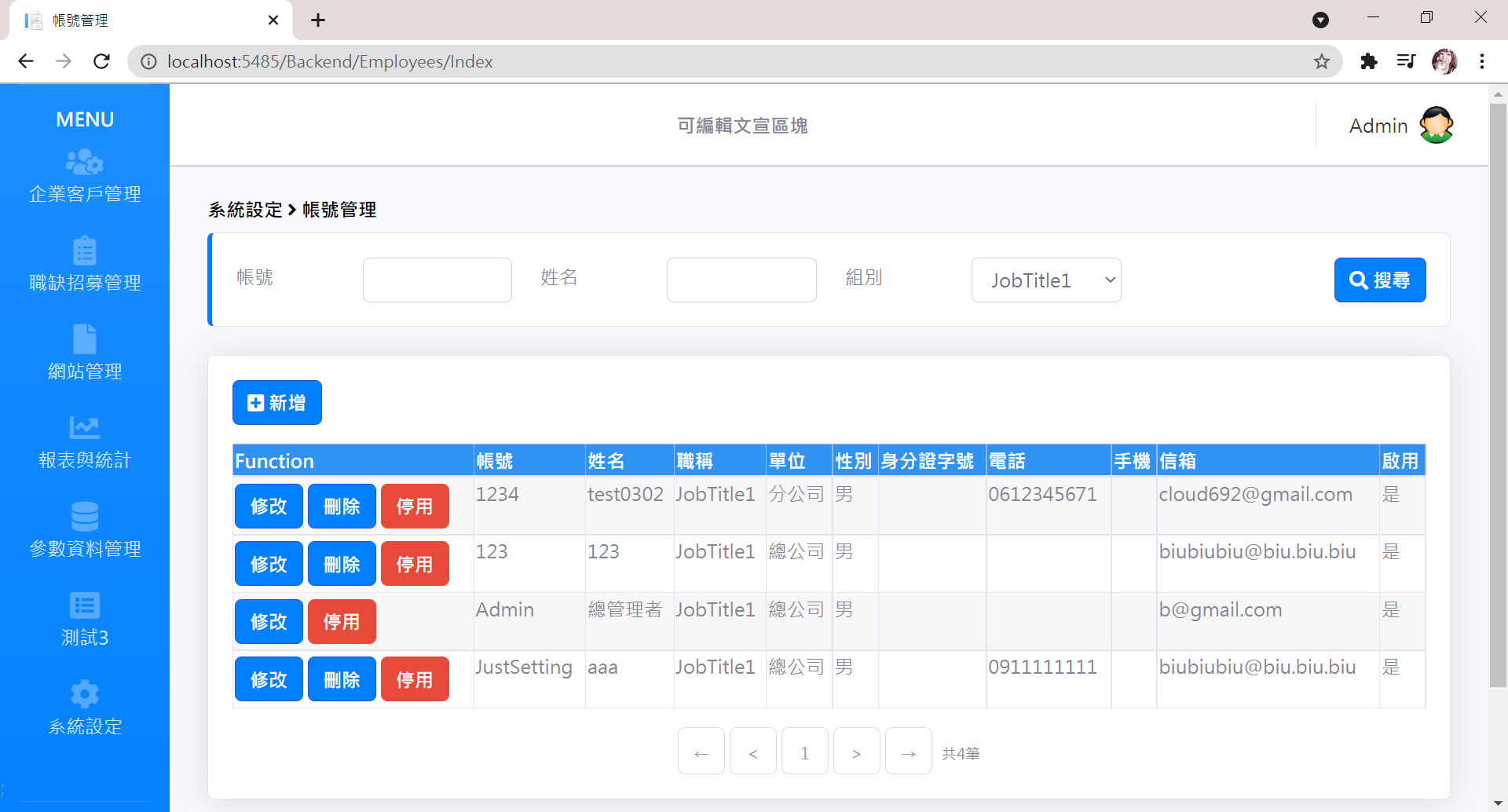
## 按鈕及頁面控管

\*[此專案無該需求，僅範例] \*

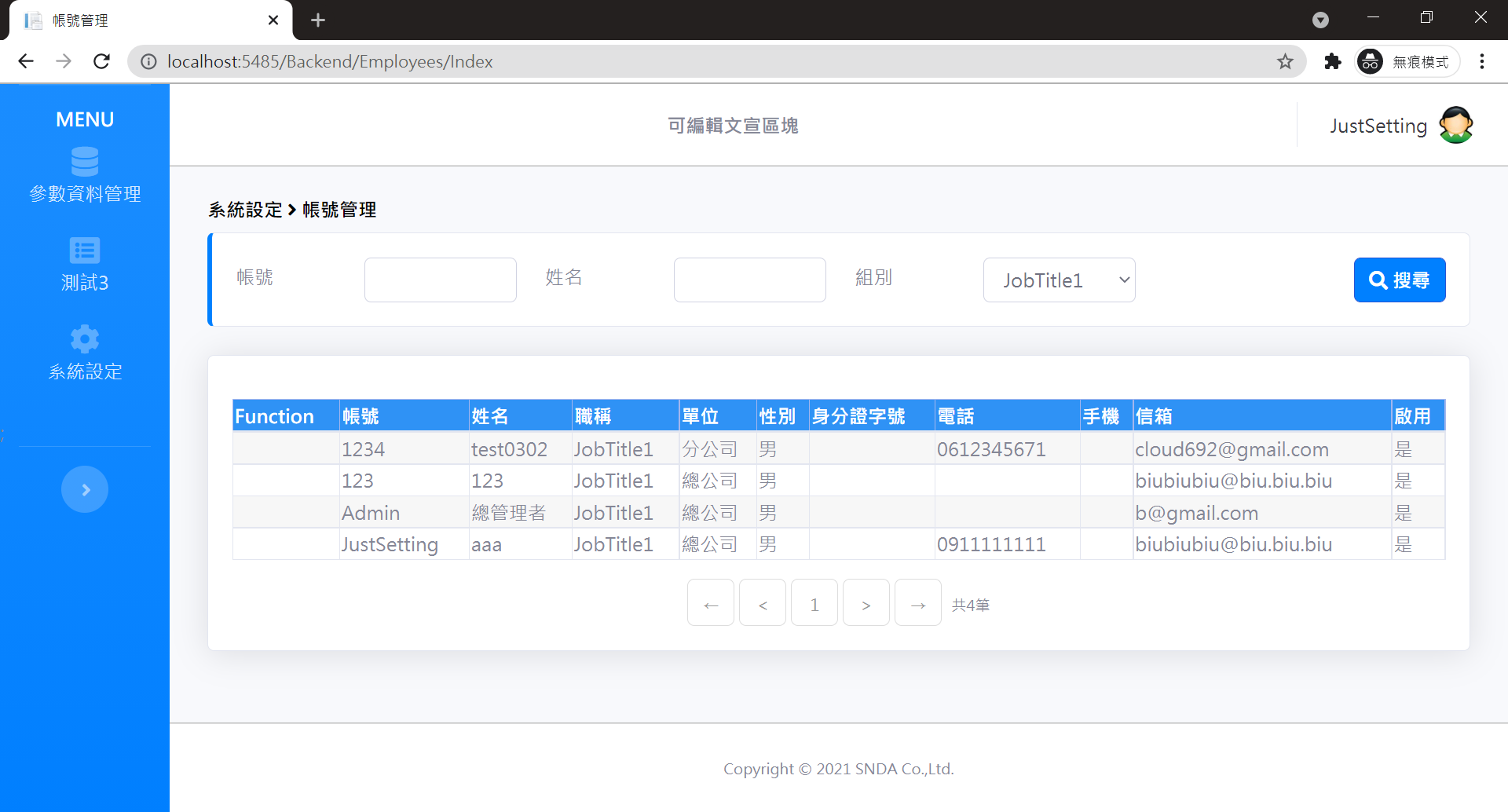
若要進行按鈕及頁面控管，需要對按鈕加上1.4節的View設定(僅隱藏)，及下面3.2節的參數controlAction，若強制進入頁面則會導回前台登入。

可參考系統設定內任一支功能項，以下為範例：

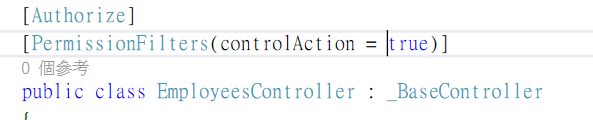
有按鈕權限:



無按鈕權限:

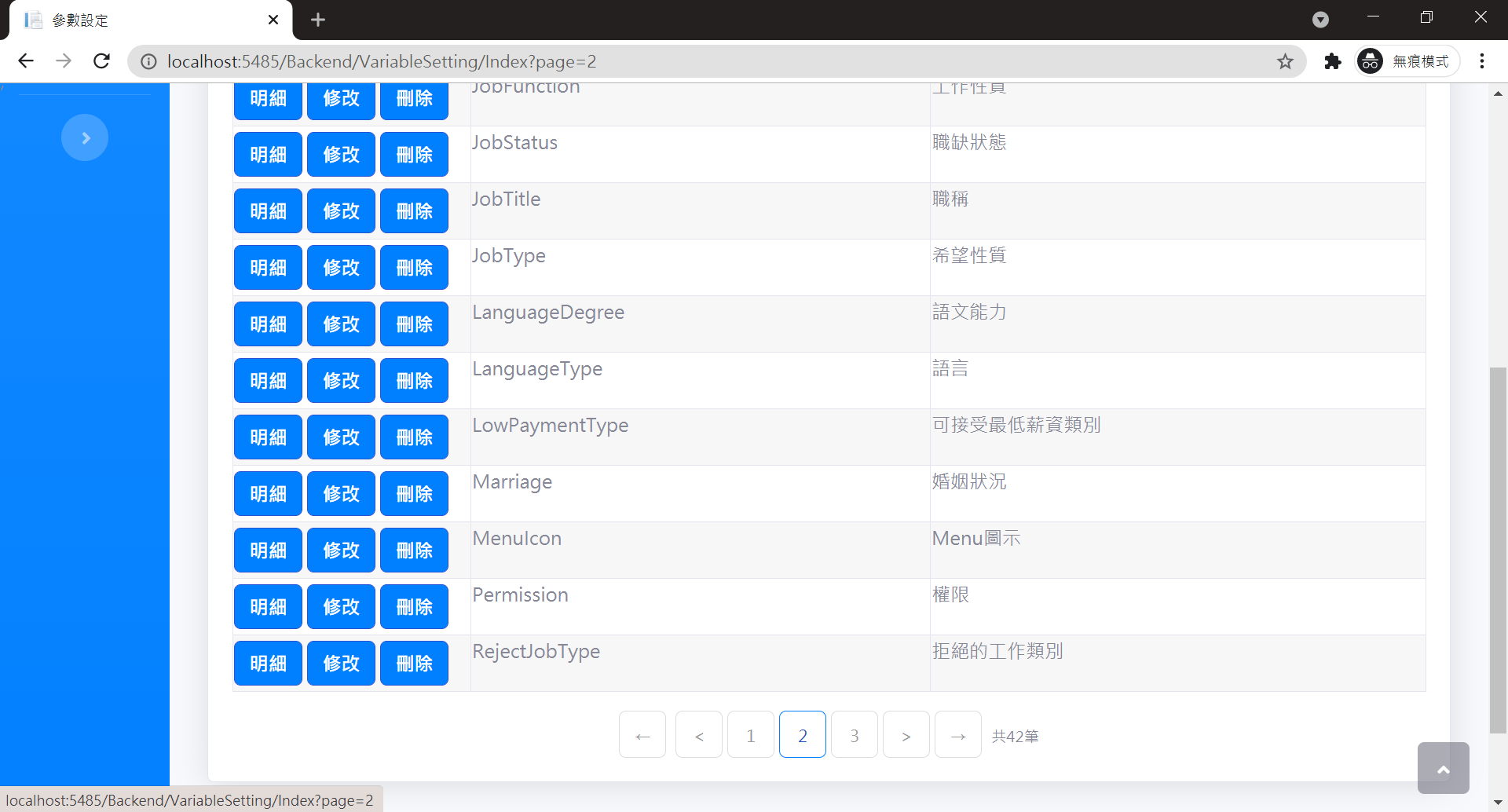


有做控鈕控制必需加上controlAction加上true 參數。



## 新增目錄圖示

在系統設定/參數設定中找到: MenuIcon- Menu圖示，點擊明細做新刪修。



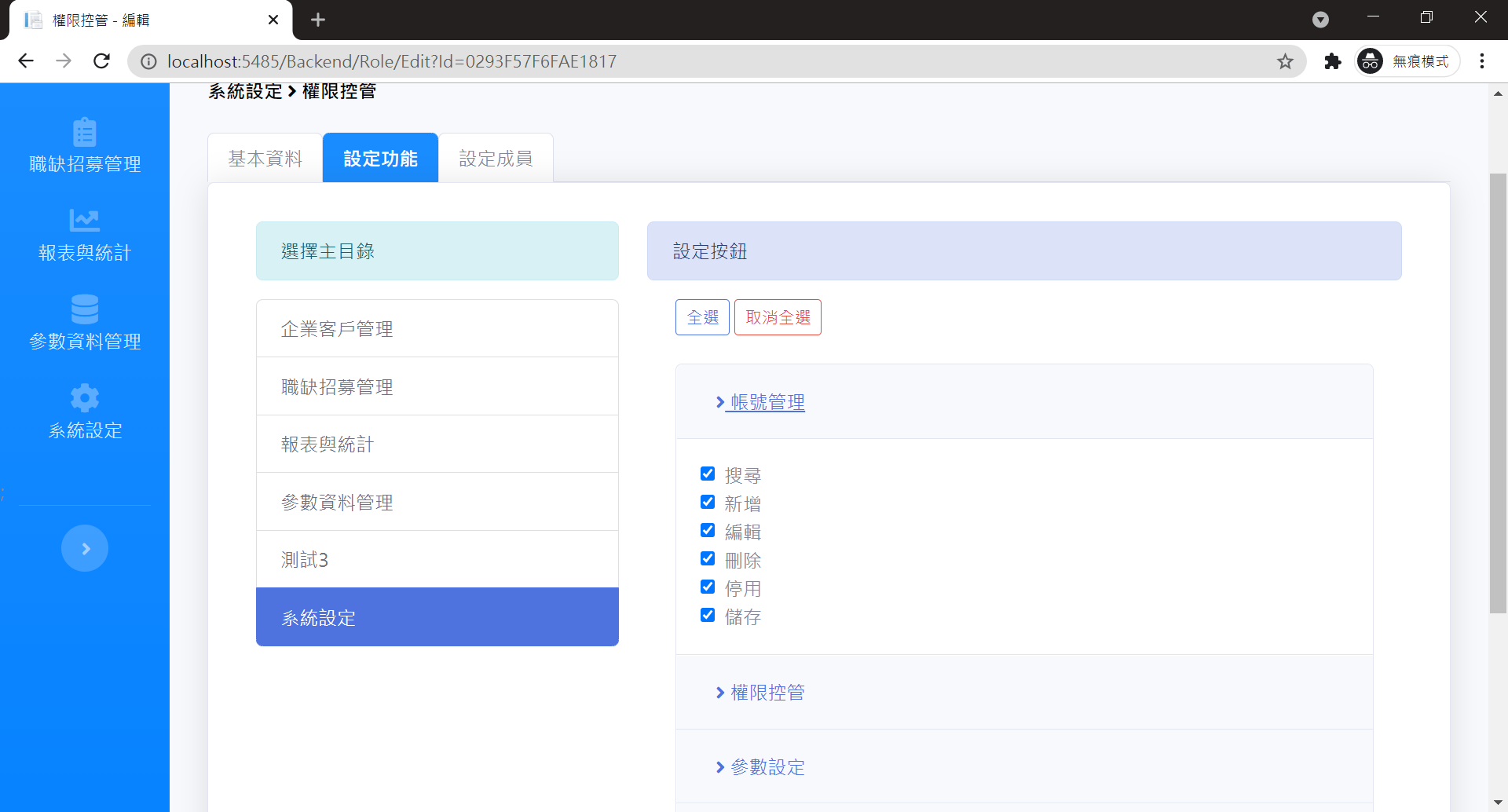
取至 [Font Awesome](https://fontawesome.com/v5.15/icons?d=gallery&p=2&m=free) ，選擇想要加入的圖示並複製html Code到說明1。



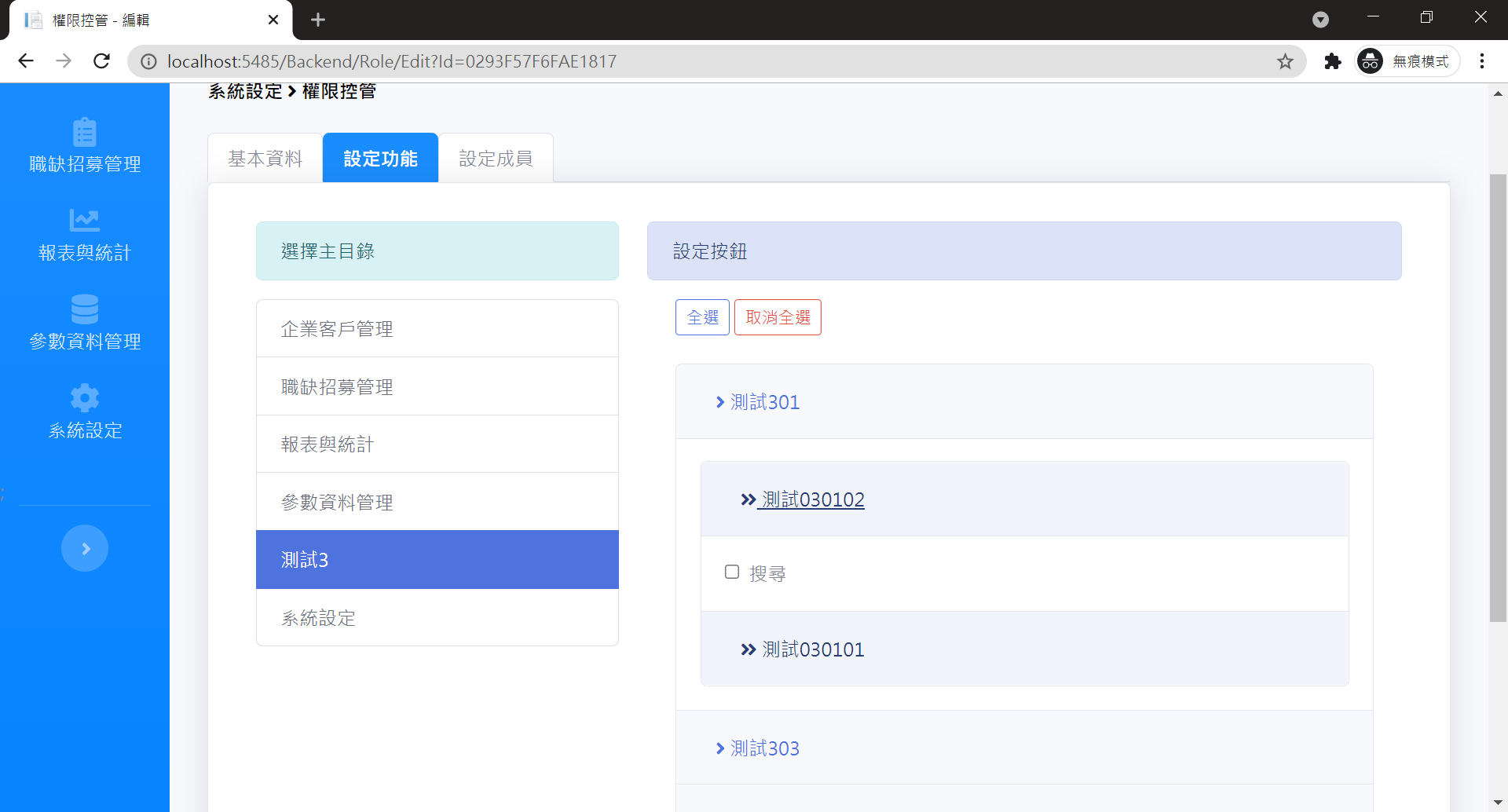
# [權限控管]

## 權限設定

進入權限控管後選擇要新增或編輯的角色，進入編輯設定可以在此畫面做權限的勾選，先選定主目錄，產生第二層/第三層的按鈕選項。

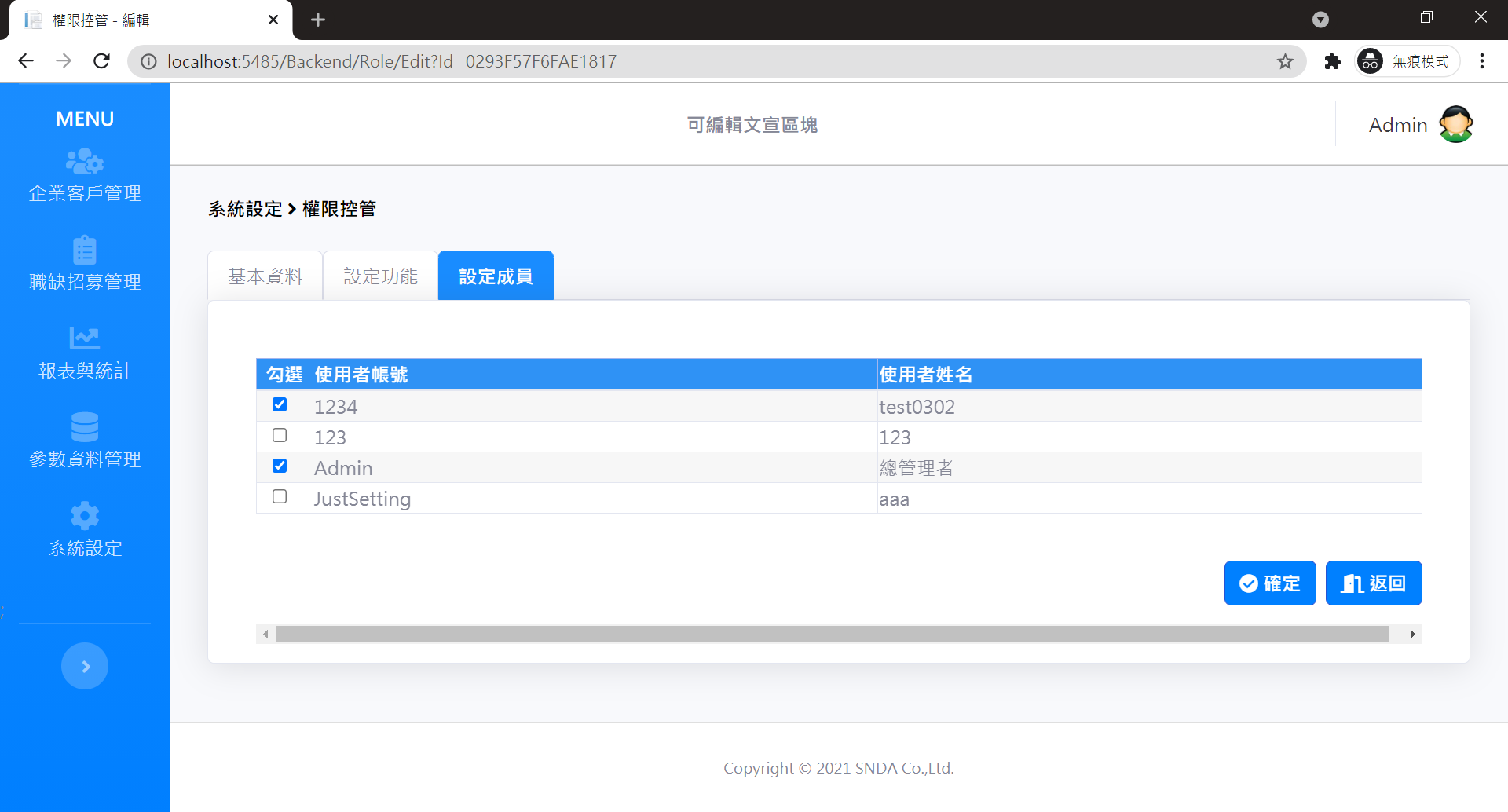


**若是沒有主目錄下沒有任何次目錄選擇”搜尋”權限，則不會在目錄中顯示此項目。**



## 設定成員

點擊設定成員頁籤可選擇能使用該權限的使用者。



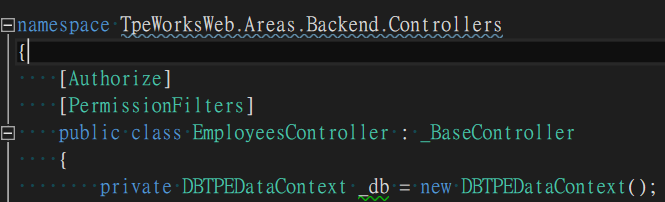
# [Controller設定]

## 登入判斷及權限控管

在需要做權限控管的public Controller上補上，注意順序

[Authorize]

[PermissionFilters]



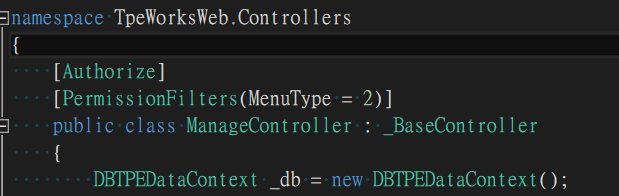
## 參數設定

在預設的情況中，不需另外設定，但以此專案為例，

因為有三種角色: 0:後台登入帳號 1:企業帳號 2:會員帳號 3:推薦人

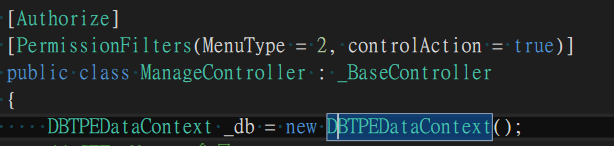
所以在後台以外角色上需要加上參數MenuType

[PermissionFilters(MenuType = 2)]



以及若是需要控制到頁面的權限，則在參數controlAction加上true

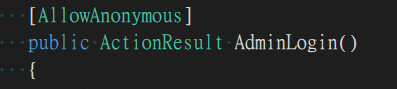
\*[此專案無該需求，前台只需設定到MenuType， 下圖為多參數範例] \*



## 允許所有權限使用

若要在最外層已加上[Authorize]的情況下有其中一個Action是不受登入限制的時候，加上

[AllowAnonymous]



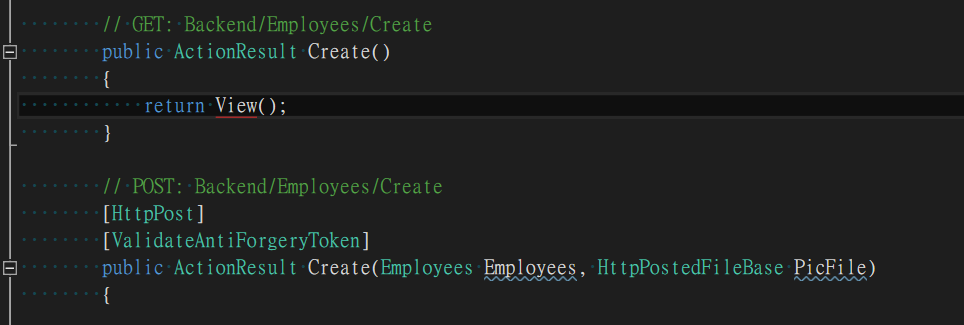
# [其他設定]

## POST

以新增和修改為例，通常此功能項都需要兩個Action，為GET、POST，這裡要注意的是，需要在POST的Action上加上

[HttpPost]

[ValidateAntiForgeryToken]



## 資料庫物件、全域變數

統一將資料庫物件及各全域變數加上private



## Index設定分頁

正常狀況下，Index都需加上參數，讓分頁運作，以下範例：

public ActionResult Index(int? page, FormCollection fc)

{

GetCatcheRoutes(page, fc);

int currentPageIndex = getCurrentPage(page, fc);

var CasesList = getPageData(currentPageIndex, DefaultPageSize, Cases.AsQueryable());

}

並在View補上：

$(document).ready(function () {

pagePlugin("pages", '@ViewBag.TotalCount', '@ViewBag.nowPage', '@ViewBag.TotalPage');

});

## 新增、刪除、修改Function

新增用SaveCreate(\_db.Table, table);

編輯用SaveUpdate(\_db.Table, table);

刪除用SaveDelete(\_db.Table, table);

若編輯時遇到錯誤，請在call資料表前加上：(參考EmployeesController/Disabled)

\_db. Table.Context.DeferredLoadingEnabled = false;

## Get參數加密傳遞

在View上使用 @Cryption.ParaEncryption(item.Id) 加密

到Action時解密 int sid = Cryption.ParaDecryption(id);

## 各支View需打上Title

除了Index外，格式為: XXXX - Action

範例:

ViewBag.Title = "選單管理";

ViewBag.Title = "選單管理 - 新增";

ViewBag.Title = "選單管理 - 編輯";