Lab 5 Report

Name: 許哲維

Student ID: 111598066

Date: 2023.06.14

1 Test Plan

1.1 Test requirements

在 Lab 5 中,要測試的系統是 KeystoneJS,這是個開源的框架,用於在 Node.js 中開發 Database 導向的網站、應用程式與 API。KeystoneJS 使創建 複雜的網站和應用程式變得容易,並帶有一個漂亮的自動生成的管理 UI。

本次實驗只需要測試要求之功能,從中設計至少 10 個測試案例並為每個特性設計一個或多個測試案例。

1.2 Test Strategy

Post Features

Create a post on the Admin UI page

Edit a post on the Admin UI page

Delete a post on the Admin UI page

Search posts by keyword on the Admin UI page

Comment Features

Create a comment on Admin UI page

Edit a comment on Admin UI page

Delete a comment on Admin UI page

Category Features

Create a category on Admin UI page

Show posts of the specific category by pressing category name on the "Blog" page

Enquirie Feature

Create an enquiry on the "Contact" page

Delete an enquiry on Admin UI page

User Feature

Create a new user on Admin UI page (Name, Email, Phone, and Password must be set when creating the new user)

1.3 Test activities

To implement the proposed strategy, the following activities are planned to perform.

No.	Activity Name	Plan hours	Schedule Date
1	Study KeystoneJS and Docker	2	2023.06.01
2	Learn Robot Framework	8	2023.06.03
3	Design test cases for the selected methods	5	2023.06.08
4	Implement test cases	6	2023.06.11
5	Complete Lab5 report	3	2023.06.14

1.4 Design Approach

使用ISP為每個待測試功能設計案例。測試工具將使用Robot Framework來達成測試需求。

1.5 Success criteria

所有所選功能之設計案例必須全部通過。

2 Test Design

To fulfill the test requirements listed in section 1.1, the following methods are selected and corresponding test cases are designed.

Test Case ID	TC-001
Name of	Create Post Test
Corresponding	(happy path)
Scenario (or Use Case)	
	1. 以管理員身分登錄
Descriptions of the	2. 選擇【Posts】並建立一則新貼文
Descriptions of the	3. 輸入新貼文內容(e.g. 標題)
Scenario (or Use Case)	4. 保存貼文
	5. 顯示已成功建立新貼文
Pre-condition or Pre- 1. KeystoneJS 正在執行	
Test Setup (Given)	2. 在 Chrome 瀏覽器開啟 KeystoneJS
	1. 用 URL = \$SERVER 跳轉到登錄畫面
	2. 用 \$DEMO EMAIL 與 \$DEMO PASSWORD
	以管理員身分登錄
Test Steps or Actions	3. 選擇【Posts】
taken by User (When)	4. 建立一則新貼文
	5. 用 \$Test String 輸入新貼文標題
	6. 建立貼文
	7. 驗證是否成功創建新貼文

	8. 登出
	\$SERVER = "http://127.0.0.1:3000/keystone/signin"
Immy Data (for	\$DEMO EMAIL = "demo@keystonejs.com"
Input Data (for Performing the	\$DEMO PASSWORD = "demo"
Actions)	\$Test String = {Null, Blank Space, String(一般長
	度、最大長度、混和字符), Digit, Symbol}
Expected Results and	建立一則具有正確標題的貼文。
Acceptance Criteria to	所有不正確的輸入都會被 Catch 並顯示錯誤訊
be Verified (Then)	息。
Post Tear-Down (or	1. 關閉 Chrome 瀏覽器
Restoring Procedures)	

Test Case ID	TC-003
Name of	Edit Post Test
Corresponding	(happy path)
Scenario (or Use Case)	
	1. 以管理員身分登錄
Descriptions of the	2. 選擇【Posts】
Scenario (or Use Case)	3. 編輯貼文並保存
	4. 顯示已成功保存貼文
Pre-condition or Pre-	1. KeystoneJS 正在執行
Test Setup (Given)	2. 在 Chrome 瀏覽器開啟 KeystoneJS
	1. 用 URL = \$SERVER 跳轉到登錄畫面
	2. 用 \$DEMO EMAIL 與 \$DEMO PASSWORD
	以管理員身分登錄
Test Steps or Actions	3. 選擇【Posts】
•	4. 編輯貼文
taken by User (When)	5. 用 \$Test String 編輯貼文內容
	6. 儲存貼文
	7. 驗證是否編輯成功
	8. 登出
	\$SERVER = "http://127.0.0.1:3000/keystone/signin"
Input Data (for	\$DEMO EMAIL = "demo@keystonejs.com"
1	\$DEMO PASSWORD = "demo"
8	
Actions)	\$Test String = {Null, Blank Space, String(一般長
	度、最大長度、混和字符), Digit, Symbol}

Expected Results and	藉由正確的輸入編輯貼文。
Acceptance Criteria to	所有不正確的輸入都會被 Catch 並顯示錯誤訊
be Verified (Then)	息。
Post Tear-Down (or	1. 關閉 Chrome 瀏覽器
Restoring Procedures)	

Test Case ID	TC-011
Name of	Delete Comment Test
Corresponding	(happy path)
Scenario (or Use Case)	
	1. 以管理員身分登錄
Descriptions of the	2. 選擇【Posts】的【Comments】
Scenario (or Use Case)	3. 刪除評論
	4. 在【Comments】顯示為空
Pre-condition or Pre-	1. KeystoneJS 正在執行
Test Setup (Given)	2. 在 Chrome 瀏覽器開啟 KeystoneJS
	1. 用 URL = \$SERVER 跳轉到登錄畫面
	2. 用 \$DEMO EMAIL 與 \$DEMO PASSWORD
	以管理員身分登錄
Test Steps or Actions	3. 選擇【Posts】
taken by User (When)	4. 選擇【Comments】
	5. 刪除評論
	6. 驗證確認是否成功刪除評論
	7. 登出
Input Data (for	\$SERVER = "http://127.0.0.1:3000/"
Performing the	\$DEMO EMAIL = "demo@keystonejs.com"
Actions)	\$DEMO PASSWORD = "demo"
Expected Results and	刪除評論。
Acceptance Criteria to	所有不正確的結果都會被 Catch 並顯示錯誤訊
be Verified (Then)	息。
Post Tear-Down (or	1. 關閉 Chrome 瀏覽器
Restoring Procedures)	

The details of the design are given below:

Lab5_TestCase.xlsx

3 Test Implementation

Section 2 中指定的測試用例設計是使用 Robot Framework 實現的。下面給出了 3 個選定測試案例的測試腳本。 其餘測試腳本實現可以在連結中找到。

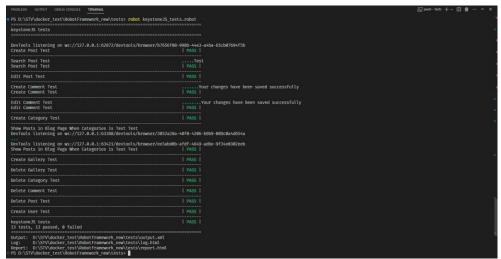
Test Case ID.	Test method	Source test code	
TC-001	Create Post Test	Create Post Test Signin Keystonejs Click to Post Page Create Action \${Result} Get Text class=css-ctpeu Should Contain \${Result} \${isSuccess} Signout Keystonejs	
TC-003	Edit Post Test	Edit Post Test Signin Keystonejs Click to Post Page Click Link link=Test Sleep 1.2s Input Action \${Result} Get Text class=css-ctpeu Should Contain \${Result} \${isSuccess} Signout Keystonejs	
TC-011	Delete Comment Test	Delete Comment Test Signin Keystonejs Click to Post Page Click to Comment Page Delete Action \${Result} Get Text class=css-pbviij Should Contain \${Result} \${CommentIsEmpty} Signout Keystonejs	

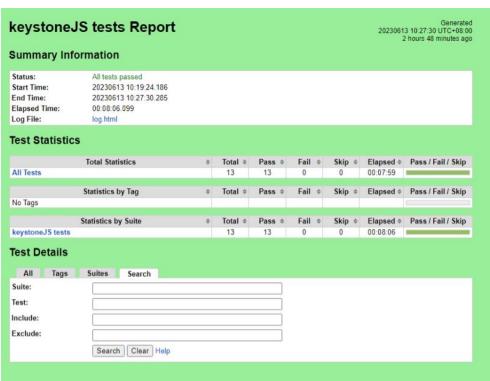
4 Test Results

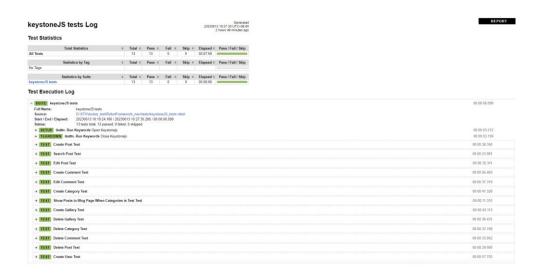
Environment & Infrastructure

windows 10

Robot Framework







5 Summary

在 Lab 5 中,使用 Robot Framework 與 ISP 設計並實現了 13 個測試案 例,且執行結果皆全部通過,因此滿足 Section 1 中的要求。

而在 Lab 5 的練習中,先透過 Robot Framework 的範例程式來學習如何操作、撰寫測試腳本,而藉由設定關鍵字來減少行數,在使用前原有 200 多行,使用後僅剩 100 多行,使得測試腳本更精簡、更有可讀性,在這一次的 Lab 令我學到很多!