**Lab 5 Report**

Name: 許哲維

Student ID: 111598066

Date: 2023.06.14

1. **Test Plan**
   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. **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. **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. **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. **Success criteria**

所有所選功能之設計案例必須全部通過。

1. **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 Corresponding Scenario (or Use Case)** | Create Post Test  (happy path) |
| **Descriptions of the Scenario (or Use Case)** | 1. 以管理員身分登錄  2. 選擇【Posts】並建立一則新貼文  3. 輸入新貼文內容(e.g. 標題)  4. 保存貼文  5. 顯示已成功建立新貼文 |
| **Pre-condition or Pre-Test Setup (Given)** | 1. KeystoneJS正在執行  2. 在Chrome瀏覽器開啟KeystoneJS |
| **Test Steps or Actions taken by User (When)** | 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  3. 選擇【Posts】  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 |
| **Input Data (for Performing the Actions)** | $SERVER = "http://127.0.0.1:3000/keystone/signin"  $DEMO EMAIL = "demo@keystonejs.com"  $DEMO PASSWORD = "demo"  $Test String = {Null, Blank Space, String(一般長度、最大長度、混和字符), Digit, Symbol} |
| **Expected Results and Acceptance Criteria to be Verified (Then)** | 建立一則具有正確標題的貼文。  所有不正確的輸入都會被Catch並顯示錯誤訊息。 |
| **Post Tear-Down (or Restoring Procedures)** | 1. 關閉Chrome瀏覽器 |

|  |  |
| --- | --- |
| **Test Case ID** | TC-003 |
| **Name of Corresponding Scenario (or Use Case)** | Edit Post Test  (happy path) |
| **Descriptions of the Scenario (or Use Case)** | 1. 以管理員身分登錄  2. 選擇【Posts】  3. 編輯貼文並保存  4. 顯示已成功保存貼文 |
| **Pre-condition or Pre-Test Setup (Given)** | 1. KeystoneJS正在執行  2. 在Chrome瀏覽器開啟KeystoneJS |
| **Test Steps or Actions taken by User (When)** | 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  3. 選擇【Posts】  4. 編輯貼文  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 |
| **Input Data (for Performing the Actions)** | $SERVER = "http://127.0.0.1:3000/keystone/signin"  $DEMO EMAIL = "demo@keystonejs.com"  $DEMO PASSWORD = "demo"  $Test String = {Null, Blank Space, String(一般長度、最大長度、混和字符), Digit, Symbol} |
| **Expected Results and Acceptance Criteria to be Verified (Then)** | 藉由正確的輸入編輯貼文。  所有不正確的輸入都會被Catch並顯示錯誤訊息。 |
| **Post Tear-Down (or Restoring Procedures)** | 1. 關閉Chrome瀏覽器 |

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

The details of the design are given below:

Lab5\_TestCase.xlsx

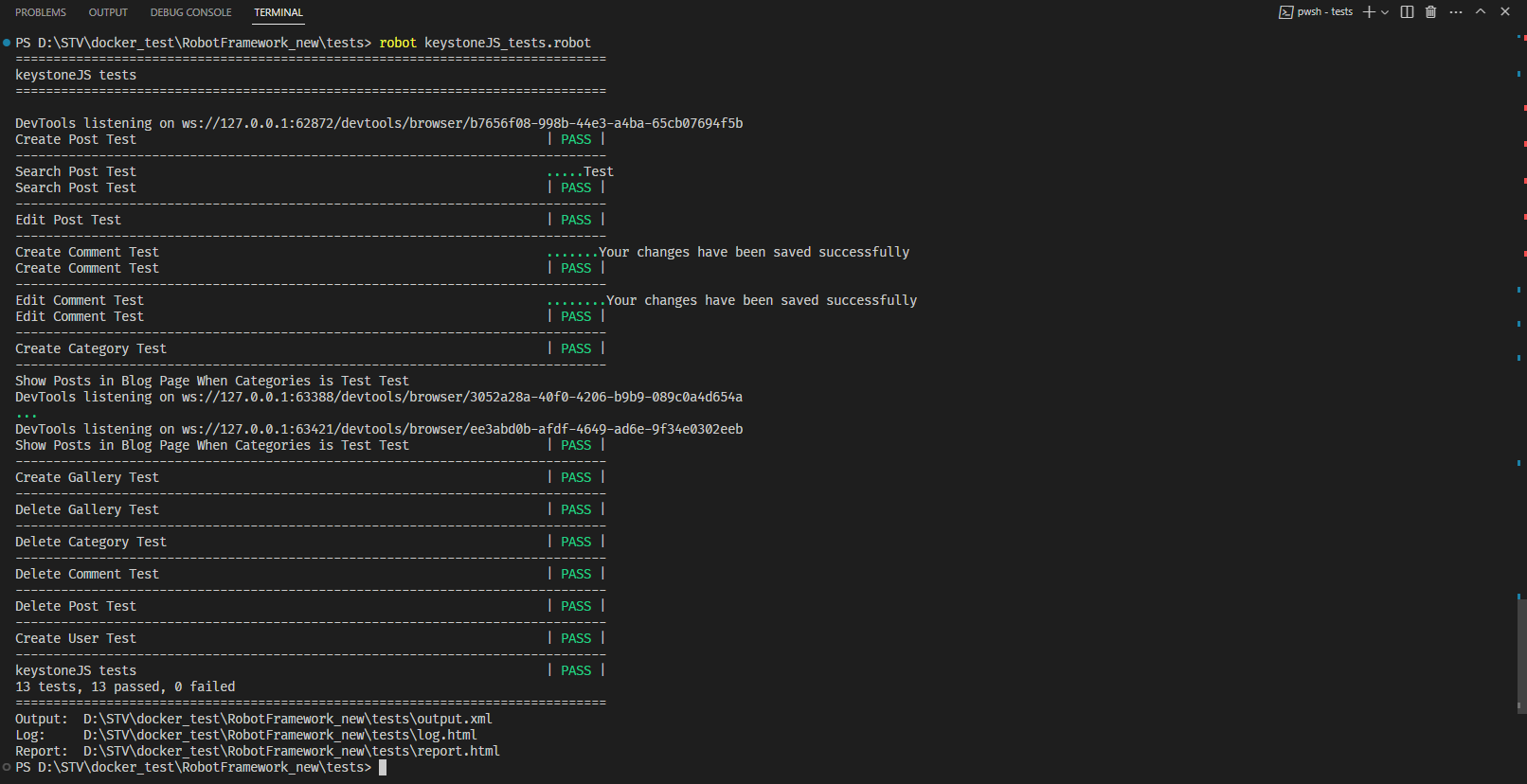
1. **Test Implementation**

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**](https://github.com/Daaata/STV)中找到。

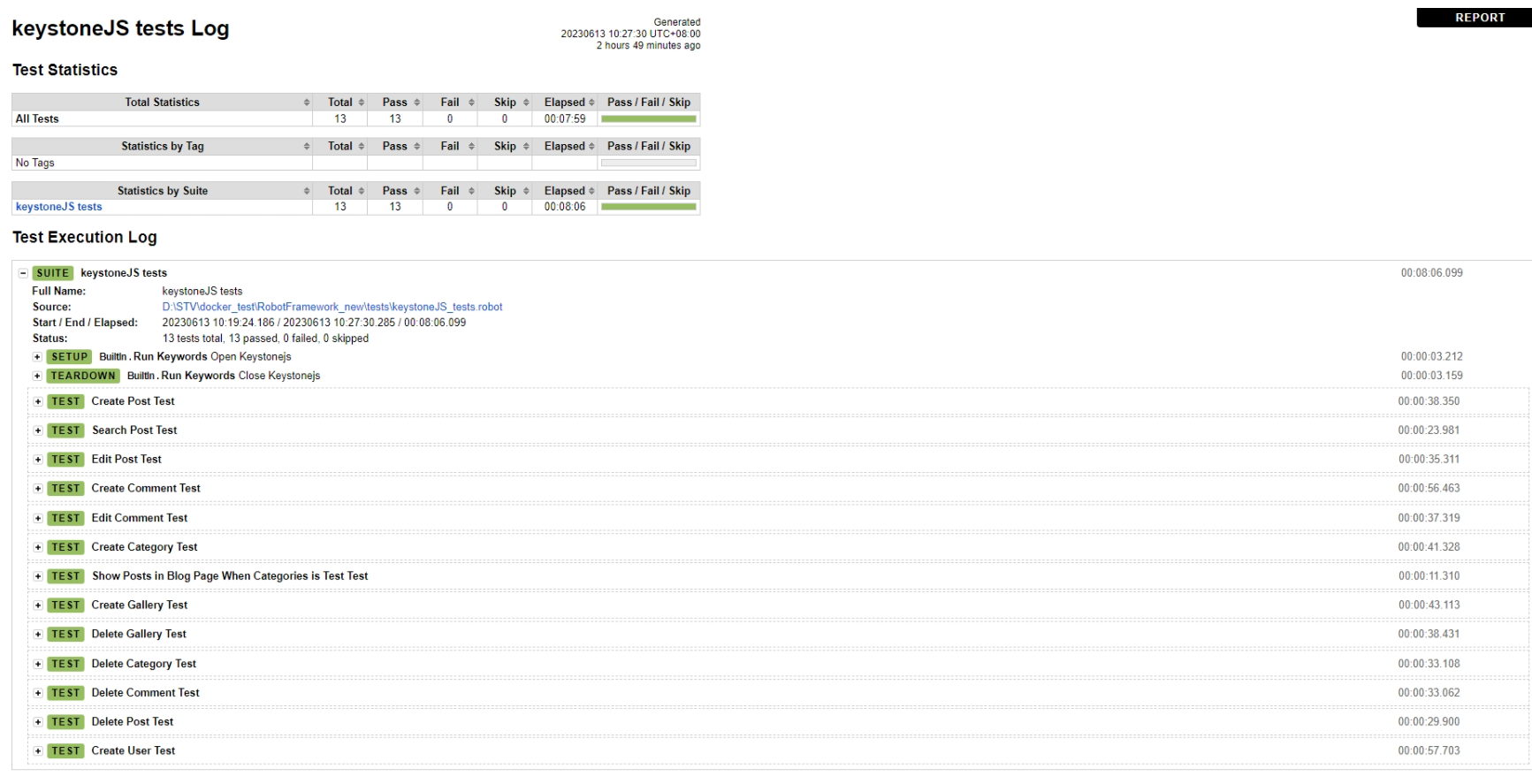
|  |  |  |
| --- | --- | --- |
| **Test Case ID.** | **Test method** | **Source test code** |
| TC-001 | Create Post Test |  |
| TC-003 | Edit Post Test |  |
| TC-011 | Delete Comment Test |  |

1. **Test Results**

|  |
| --- |
| **Environment & Infrastructure**  **windows 10**  **Robot Framework** |

****

****

****

1. **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！