

軟體測試期末報告

內容 (what- function)

- 我製作的是自動化研討會活動建置與管理雲端系統後端 API 服務，我這次測試的功能主要是使用者**登入**、**註冊**和研討會基本資訊的**增**、**刪**、**改**、**查**。

設計 (how- design)

- 登入功能
 - 使用者利用帳號與密碼進行登入。
 - 帳號與密碼驗證正確，便會回傳一串 Token.
- 註冊功能
 - 帳號：帳號必須要是電郵格式。（必填）
 - 名稱：名稱的長度必須在 2 至 30 個字母、數字。（必填）
 - 密碼：密碼的長度必須在 6 至 30 個字母或是數字。（必填）
- 新增研討會活動專案
 - 專案名稱：名稱的長度必須在 2 至 50 個字母。（必填）
- 研討會基本資訊管理功能
 - 研討會基本資訊的增、刪、改、查時，需要符合以下規則：
 - 登入狀態：必須要為真，且需要有用戶的登入 Token.（必填）
 - 專案的代號：專案編號。（必填）
 - 開始日期：日期格式年-月-日。開始日不能比今天小。
 - 結束日期：結束日期不會超過 7 天。
 - 舉辦者：舉辦者的名稱必須是 2 至 20 個字母。
 - 地點：地點名稱必須是 2 至 20 個字母。
 - 活動名稱：活動名稱必須是 2 至 20 個字母。
 - 詳細資訊：詳細資訊必須 2 至 1000 個字母。
- 刪除研討會活動專案
 - 專案名稱：名稱的長度必須在 2 至 50 個字母。
 - 專案的代號：專案編號。

程式模組說明 (how - code)

源代碼:

| 文件路徑 | 解釋 |
|---------------------------------------|-------------------------------|
| ./server/app.js | 伺服器的進入點。 |
| ./server/config/firebase.config.js | 資料庫的設定檔案。 |
| ./server/routes/loginRouter.js | 使用者登入伺服器的路由。 |
| ./server/routes/projectRouter.js | 活動專案的路由。 |
| ./server/routes/signupRouter.js | 註冊帳號的路由。 |
| ./server/utls/classes/dataInfo.js | 活動基本資訊的類別。 |
| ./server/utls/betweenTwoDays.js | 計算開始與結束的隔幾天。 |
| ./server/utls/checkEmailValidation.js | 檢查註冊電郵是否已註冊。 |
| ./server/utls/findUser.js | 查詢使用者是否有效。 |
| ./server/utls/schemas.js | 定義 request body 的結構。 |
| ./server/utls/vaildRequest.js | 檢查 user ID 與 Project ID 是否有誤。 |
| ./server/utls/validation.js | 檢查 request body 的結構與定義的是否相同。 |
| ./server/utls/verifyToken.js | 檢查 Json Web Token 是否有效。 |

測試代碼:

Path: ./server/___test___/

a) sample.test.js

1) 這裡測試無效的 API，用戶的登入和註冊功能。

| 描述 | 解釋 |
|------------------|-------------------------|
| Login test 1 | 測試登入功能。已註冊的帳號。 |
| Login test 2 | 測試登入功能。未註冊的帳號。 |
| Login test 3 | 測試登入功能。正確的帳號，錯誤的帳號。 |
| Login test 4 | 測試登入功能。沒有傳值的狀態下。 |
| Sign Up test 1 | 測試註冊功能。 |
| Sign Up test 2 | 測試註冊功能。在已註冊的帳號之下可否重複註冊。 |
| Sign Up test 3 | 測試註冊功能。錯誤的格式。 |
| Sign Up test 4 | 測試註冊功能。在沒有傳值的狀態。 |
| Get Unknow Route | 測試無效的 API. |

b) project.test.js

1) 這裡測試活動基本資訊的增、刪、改、查。

| 描述 | 解釋 |
|---------------------------|-------------------------|
| Post create new project 1 | 測試新增活動專案。 |
| Post create new project 2 | 測試無 Token，可否新增專案。 |
| Post create new project 3 | 測試活動名稱過長。 |
| Post get all project | 測試使用者新增活動專案之後時候可以讀取。 |
| Get dashboard data 1 | 測試是否可以讀取專案的資料。 |
| Get dashboard data 2 | 測試是否可以讀取專案的資料，在錯誤的架構之下。 |
| Get dashboard data 3 | 測試無 Token，可否讀取資料。 |

| | |
|---------------------------|--|
| Get Participants 1 | 測試參加研討會活動的總人數。 |
| Get Participants 2 | 測試 無 Token. |
| Get Participants 3 | 測試在 無效 的 Token. |
| Get Participants 4 | 測試在 無效 的專案名稱。 |
| Create Basic Event Info | 測試 新增 基本活動資訊。 |
| Update Basic Event Info 1 | 測試 更改 基本活動資訊。 |
| Update Basic Event Info 2 | 測試活動開始日期與結束日期相隔 大於 7 天。 |
| Update Basic Event Info 3 | 測試活動結束日期是否符合格式和是否有不再日曆上的日期。 (如：32日) |
| Update Basic Event Info 4 | 測試活動開始日期是否符合格式和是否有不再日曆上的日期。 (如：0日) |
| Update Basic Event Info 5 | 測試 沒有 Token 可否更新活動資訊。 |
| Update Basic Event Info 6 | 測試 沒有 專案編號是否可以更新活動。 |
| Update Basic Event Info 7 | 測試開始與結束日期都是 今天 。 |
| Update Basic Event Info 8 | 測試 無效 的 Token 可否更新活動資訊。 |
| Remove Project 1 | 測試 刪除 專案的功能。 |
| Remove Project 2 | 測試在 無效 的 Token ，可否刪除專案。 |
| Remove Project 3 | 測試在 沒有 Token ，可否刪除專案。 |
| Remove Project 4 | 測試在 沒有 專案編號，可否刪除專案。 |
| Remove Project 5 | 測試在 沒有 專案名稱，可否刪除專案。 |

成果 (what- result)，包含畫面的說明

- 測試碼（因為程式碼太長，所以以連結的方式呈現）
 - 測試碼的部分，分成三部分
 - 第一部分是登入：
https://github.com/D0561910/AMW_Server/blob/master/server/___test___/login.test.js
 - 第二部分是註冊：
https://github.com/D0561910/AMW_Server/blob/master/server/___test___/signup.test.js
 - 第三部分是專案：
https://github.com/D0561910/AMW_Server/blob/master/server/___test___/project.test.js
- 結果分成三部分
 - a) 登入結果：

```
PASS server/___test___/login.test.js
login test 1
  ✓ Login API with true email and password (1100ms)
login test 2
  ✓ Login API with wrong email and password (178ms)
login test 3
  ✓ Login API with true email, but wrong password (246ms)
login test 4
  ✓ Login API with empty body (13ms)
login test 5
  ✓ Login API with error email (8ms)
```

b) 註冊結果：

```
PASS server/___test___/signup.test.js
Sign Up test 1
  ✓ Sign Up API with new email, username and password (1139ms)
Sign Up test 2
  ✓ Sign Up API with same email, username and password (175ms)
Sign Up test 3
  ✓ Sign Up API with error type (10ms)
Sign Up test 4
  ✓ Sign Up API with empty body (9ms)
Get Unknow Route
  ✓ Get Unknow Route (10ms)
```

c) 專案結果：

```
PASS server/__test__/project.test.js (12.181s)
  Post Create New Project 1
    ✓ Create New Event Project (18ms)
  Post Create New Project 2
    ✓ Create New Event Project without Token (9ms)
  Post Create New Project 3
    ✓ Create New Event Project with longest name (9ms)
  Post get all project
    ✓ Request user project (174ms)
  Get Dashboard Data 1
    ✓ Request Basic Event Infomation with project ID and user token (336ms)
  Get Dashboard Data 2
    ✓ Request Basic Event Infomation with project ID and error user token (578ms)
  Get Dashboard Data 3
    ✓ Request Basic Event Infomation with project ID only (7ms)
  API Get Participants 1
    ✓ Request Total Num of Participants (336ms)
  API Get Participants 2
    ✓ Request Total Num of Participants with project ID only (7ms)
  API Get Participants 3
    ✓ Request Total Num of Participants with project ID and Invaild Token (171ms)
  API Get Participants 4
    ✓ Request Total Num of Participants (182ms)
  Create Basic Event Info
    ✓ Create Basic Event Information (341ms)
  Update Basic Event Info 1
    ✓ Update Basic Event Information (335ms)
  Update Basic Event Info 2
    ✓ Start Date and End Date are greater than 7 days (10ms)
  Update Basic Event Info 3
    ✓ Update Invaild days (8ms)
  Update Basic Event Info 4
    ✓ Update Invaild days (8ms)
  Update Basic Event Info 5
    ✓ Update with no token (7ms)
  Update Basic Event Info 6
    ✓ Update with no project id (8ms)
  Update Basic Event Info 7
    ✓ Update start date and end date is today (9ms)
  Update Basic Event Info 8
    ✓ Update Basic Event Information with Invaild Token (170ms)
  Remove Project 1
    ✓ Remove Project with correct token, project ID and project name (170ms)
  Remove Project 2
    ✓ Remove Project with correct project ID and project name but without token (7ms)
  Remove Project 3
    ✓ Remove Project with correct project ID and project name but invaild token (429ms)
  Remove Project 4
    ✓ Remove Project with correct token and project name (412ms)
  Remove Project 5
    ✓ Remove Project with correct token and project ID (173ms)
```

- Jest Test Coverage

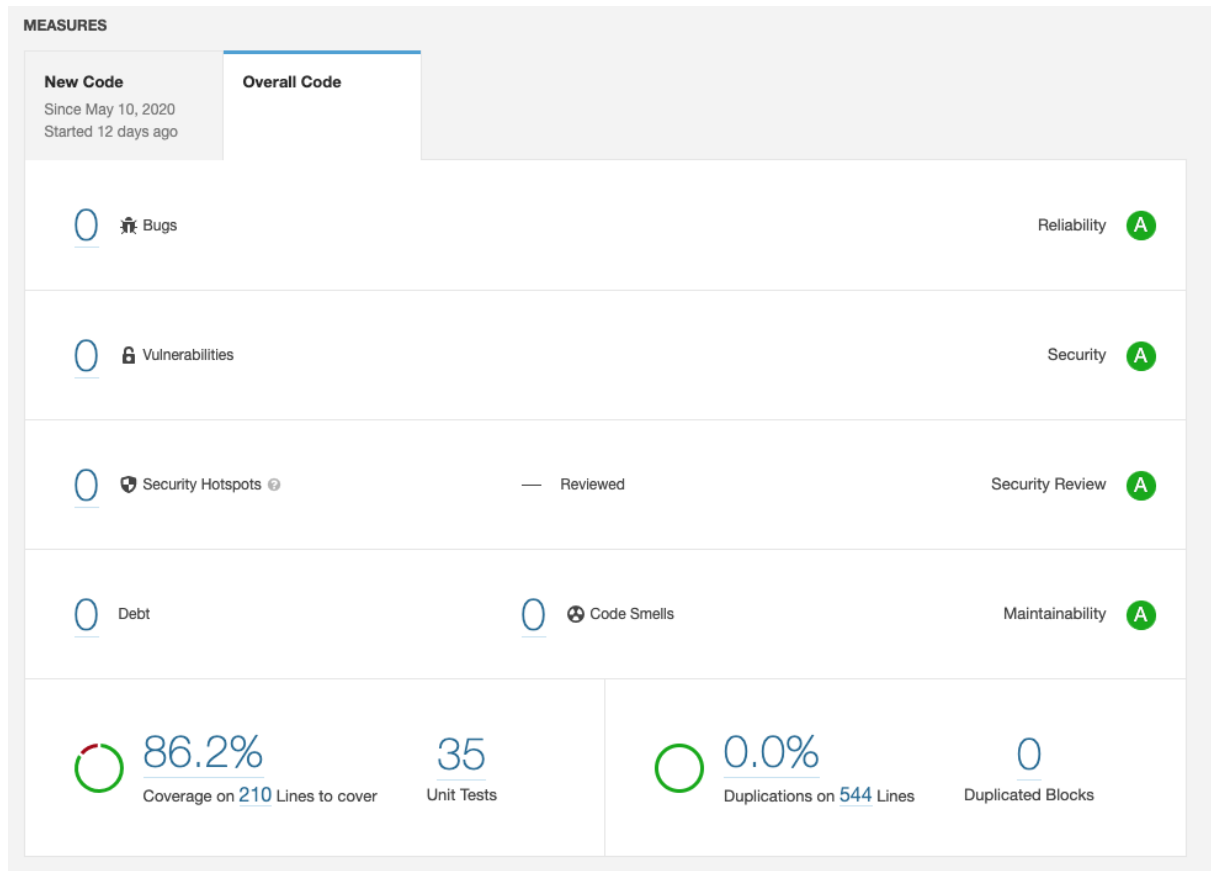
All files

98.93% Statements 185/187 86.36% Branches 38/44 97.3% Functions 36/37 98.91% Lines 181/183

Press n or j to go to the next uncovered block, b, p or k for the previous block.

| File | | Statements | | Branches | | Functions | | Lines | |
|----------------------|------------------------|------------|---------|----------|-------|-----------|-------|--------|---------|
| server | <div><div></div></div> | 84.62% | 11/13 | 0% | 0/6 | 0% | 0/1 | 84.62% | 11/13 |
| server/config | <div><div></div></div> | 100% | 2/2 | 100% | 0/0 | 100% | 0/0 | 100% | 2/2 |
| server/routes | <div><div></div></div> | 100% | 105/105 | 100% | 23/23 | 100% | 19/19 | 100% | 103/103 |
| server/utlis | <div><div></div></div> | 100% | 60/60 | 100% | 15/15 | 100% | 16/16 | 100% | 58/58 |
| server/utlis/classes | <div><div></div></div> | 100% | 7/7 | 100% | 0/0 | 100% | 1/1 | 100% | 7/7 |

- sonarQube MEASURES



測試報告 (test)

- 測試方法

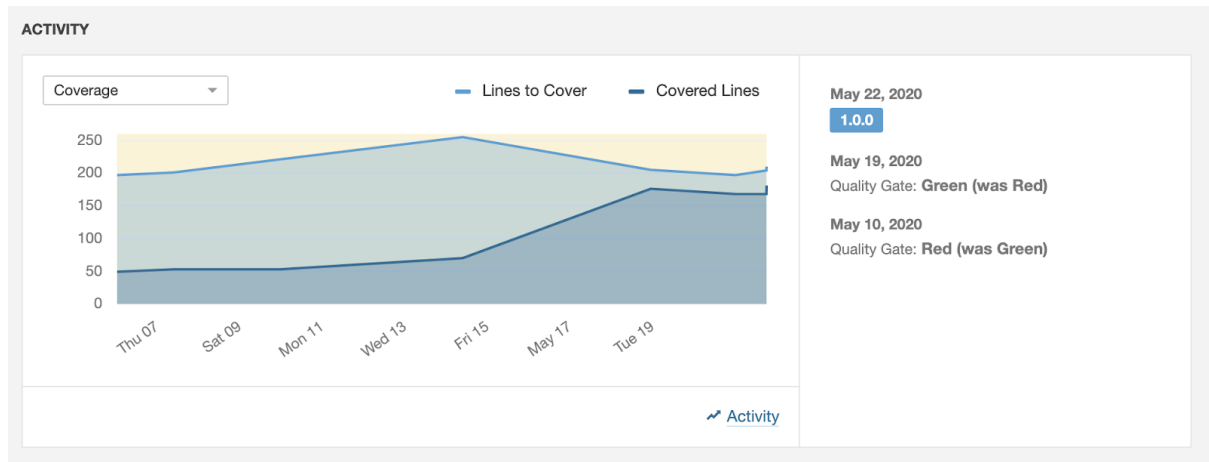
- 黑箱測試：在測試設計與案例說明內有標記每項測試是使用那種黑箱測試測試方法。
- 白箱測試：敘述涵蓋度98.93%，所有分支涵蓋度86.36%，功能涵蓋度97.3%和程式碼涵蓋度98.91%。

- 測試設計與案例說明

- [軟體測試設計與案例說明連結](#)

- 測試結果

- sonarQube 的 Activity Coverage



自評：

1. 功能的完整性 (包含程式碼行數)

- 登入系統的部分：

對於登入系統的部分，個人覺得蠻完整的。因為我的登入系統有檢測帳號與密碼。當使用者登入系統後會得到一串金鑰，用於確認使用者身分。

- 註冊帳號的部分：

對於註冊帳號的部分，個人覺得也是有信心的。因為系統會檢查帳號是否會重複註冊和密碼是使用秘文的方式儲存在資料庫內。

- 研討會管理功能部分：

對於內部功能的部分，在這的完整性有點不是那麼完整，因為我只是實作了一小部分的管理功能而已。例如：新增每日活動功能、活動圖片管理、活動地點、接駁車的資訊等等。

- 終結：

整個系統的架構個人覺得還可以接受，但是應該還可以更好。

2. 測試的完整性 (包含測試案例數，涵蓋度等量化指標)

- 敘述涵蓋度98.93%，所有分支涵蓋度86.36%，功能涵蓋度97.3%和程式碼涵蓋度98.91%
- 測試案例的設計沒有達到最好。
- 應該可以做得更好，例如：壓力測試

3. 測試的自動化程度 (包含測試碼行數)

- 我的自動化測試只是在 Github 上進行 Unit Test 測試而已。因為，沒有伺服器的原因。對於 sonarQube 的程式碼品質與安全是在本地端進行測試。
- 每次我把專案 push 上去 Github 時，Github 的 Action 會自動幫我 build 和 test。當出現 build failed 或是 test failed 時，Github 會以電郵的方式通知我。
- 我的 Commit 有很多以下的截圖是我最近的 push action。如果要看全部可以連結到以下網站：https://github.com/D0561910/AMW_Server/actions

| Event ▾ | Status ▾ | Branch ▾ | Actor ▾ |
|---|----------|----------|----------------------|
| ✓ test: add new test case reason: because follow full pair... | | master | 1 hour ago 1m 31s |
| Node.js CI #77: Commit 309fcc9 pushed by D0561910 | | | |
| ✓ remove console.log | | master | yesterday 1m 19s |
| Node.js CI #76: Commit 1a6fad7 pushed by D0561910 | | | |
| ✓ update variable name | | master | yesterday 1m 34s |
| Node.js CI #75: Commit 647c875 pushed by D0561910 | | | |
| ✓ test: add test case. reason: get member and update ba... | | master | yesterday 1m 55s |
| Node.js CI #74: Commit f5abc8f pushed by D0561910 | | | |
| ✓ fix: update basic event info schema. reason: basic eve... | | master | yesterday 1m 45s |
| Node.js CI #73: Commit c3bab17 pushed by D0561910 | | | |
| ✓ feat: add vaildRequest function. reason: vaildRequest f... | | master | yesterday 1m 26s |
| Node.js CI #72: Commit 0af0f8c pushed by D0561910 | | | |
| ✓ refactor: sample.test.js -> login.test.js & signup.test.js... | | master | 2 days ago 1m 33s |
| Node.js CI #71: Commit 85c7c72 pushed by D0561910 | | | |
| ✓ feat: update package.json reason: when run babel com... | | master | 2 days ago 2m 7s |
| Node.js CI #70: Commit 1b24836 pushed by D0561910 | | | |
| ✓ test: add project test case. reason: add without projec... | | master | 2 days ago 1m 30s |
| Node.js CI #69: Commit 7ad2735 pushed by D0561910 | | | |
| ✓ test: add remove project test case reason: forget test t... | | master | 2 days ago 1m 34s |
| Node.js CI #68: Commit 5376b9e pushed by D0561910 | | | |
| ✓ test: add get total Participants test case, without toke... | | master | 2 days ago 1m 23s |
| Node.js CI #67: Commit 88bd09e pushed by D0561910 | | | |
| ✓ test: add create new project test case. reason: add tes... | | master | 2 days ago 1m 46s |
| Node.js CI #66: Commit 9cb2557 pushed by D0561910 | | | |