

SAD Milestone4 測試報告

Group 10

測試目標說明

- 確認系統主要功能符合需求
- 驗證核心操作流程無錯誤
- 觀察使用者介面是否符合預期

測試一：OCR 辨識測試

測試目標：確認系統是否能成功接收範例文章的掃描檔，並正確進行 OCR 辨識，將辨識結果顯示於使用者界面。

測試範圍：

1. 測試上傳不同種類的掃描文章檔案，並驗證系統能否正確辨識。
2. 驗證 OCR 模型是否正確呼叫，並比較其辨識結果與原文。

測試步驟：

1. **檔案上傳：**上傳格式正常的範例文章掃描檔，有固定幾個測試用範文。各有清楚模糊等等的差異。
2. **OCR 辨識：**前端傳回後端處理並執行 OCR 辨識。
3. **顯示結果：**顯示 OCR 辨識後的文字結果，並與原文進行對比。

預期結果：

1. **系統成功接收檔案：**系統應能成功接收並儲存上傳的檔案，無任何錯誤或中斷。並顯示文章到使用者介面上。
2. **OCR 辨識文字正確顯示：**OCR 辨識後的文字應與原文內容相符，無明顯錯誤或遺漏。
3. **辨識時間符合預期：**上傳與辨識時間應該控制在合理範圍內，超過 15 秒的情況應被視為異常。

測試案例：

測試項目	測試結果
檔案格式是否支援	成功接收檔案
傳送非 pdf	成功出現錯誤訊息
上傳與辨識時間是否超過預期	超過預期，花了一分三十五秒
是否呼叫到正確的 OCR 模型	是，和本地端比對過結果相符

邏輯檢查點：

1. 在經過檢查後，超過預期的原因是電腦的差異，是否使用 GPU 來做平行的推論對 OCR 辨識的效率有著極高的影響力。因此使用此功能時必須確定架設時的設定情況。

測試二：文章修復與比對

測試目標：確認系統是否能正確在修復介面修改辨識錯誤文字，並保證修改後的文字即時更新，與原文章相符。

測試範圍：

1. 測試使用者是否能夠點擊並進行文字修改。
2. 測試多處錯誤文字的標記與修改功能。
3. 測試多次一鍵修改後，內容是否正確更新且不會偏離文章主題。

測試步驟：

1. **載入修復介面：**能在畫面中看到 OCR 辨識過後的原始文章。
2. **手動修改：**能否使用刪除和鍵盤輸入修改原始辨識結果。
3. **一鍵修復：**手動標記出多處需要修改的文字錯誤，並與修改後對比，檢查功能是否能達到目的。
4. **多次執行一鍵修改：**內容是否正確更新且不會偏離文章主題。

預期結果：

1. **點擊並能使用鍵盤輸入**
2. **標記多處錯誤並一鍵修改：**使用者應能夠一鍵修改多處錯誤，並且修改後的文字內容符合預期。
3. **多次點擊不會偏離主題：**每次點擊一鍵修改後，應該保持文章的正確性與主題一致。

測試案例：

測試項目	測試結果
點擊並能使用鍵盤輸入	成功
標記多處錯誤並進行一鍵修復	成功修改亂碼，以及填補標點符號
多次點擊不偏離主題	成功，功能無填補任何不相關內容

邏輯檢查點：

- 1. 在測試內容內，功能都正常，多次點擊一件修復最多是重新排版，對於自數與內容並沒有很大差距，修改過後在範文的辨識度超過 99% 的正確率，與原始的 92% 有很大的進步。
- 2. 若自行加入一些亂碼，功能可以有效分辨以及去除亂碼。

測試三：分項評分與扣分明細功能測試

測試目標：確認系統能夠對文章進行分項評分，並正確顯示扣分明細，讓使用者能夠清楚了解每個部分的得分情況，並能夠識別文章的優缺點。

測試範圍：

- 1. 測試系統能否依照設定的評分項目，對文章進行分項評分。
- 2. 測試系統是否能夠正確顯示每個評分項目的扣分說明。
- 3. 測試系統是否使用正確的評分準則和 Prompt 來計算分數。

測試步驟：

- 1. 提交文章進行評分：提交一篇測試文章，確保文章內容覆蓋不同的評分項目，如語法、邏輯結構、內容完整性等。
- 2. 確認回傳結果：檢查系統回傳的評分結果是否包含分項及扣分明細。
- 3. 分項評分與扣分明細：確認每個評分項目是否正確列出其得分與扣分原因，並且是否符合預期的評分準則。
- 4. 確認評分結果合理性：內容是否正確更新且不會偏離文章主題。

預期結果：

- 1. 回傳格式正確：系統應回傳正確格式的評分結果，包含每個分項的得分、扣分明細，以及最終分數。確保資料型態正確，且字數不會過多

2. **明確列出優缺點**：評分結果應該具體指出文章的優缺點，並附有清楚的扣分原因，讓使用者能夠理解哪些部分需要改進。
3. **評分結果合理**：評分結果應該與文章內容相符，並符合預定的評分準則，確保評分過程客觀公正。
4. **正確使用評分準則和 Prompt**：系統應該根據正確的評分準則對文章進行評分，並使用合適的 Prompt 來生成扣分明細。

測試案例：

測試項目	測試結果
回傳格式是否正確	成功
是否列出寫作文章的優缺點	成功，並有類似具體說明
評分結果是否合理	成功
是否呼叫正確的評分準則與 Prompt	成功

邏輯檢查點：

1. 在測試內容當中，評語皆再合適的長度，分數與多次對比後的結果不會相差過多，評語具體且回應的是相關的欄位內容。

測試四：分數穩定性測試

測試目標：確認同一篇文章反覆上傳系統後，系統的評分結果穩定，且歷史紀錄能夠正確保存每次上傳的結果。

測試範圍：

1. 測試同一篇文章反覆上傳多次後，評分結果是否穩定。
2. 測試每次上傳後，歷史紀錄是否正確保存。
3. 測試歷史紀錄中每次上傳的分數、原文、評分細項是否一致。

測試步驟：

1. **反覆上傳同一篇文章**：將同一篇文章反覆上傳系統大約 3 次，並比對批改分數結果。
2. **記錄每次評分結果**：每次上傳後，記錄系統返回的評分結果和扣分明細。

3. **比對歷史紀錄與上傳結果：**檢查歷史紀錄中的分數、原文與評分細項是否與上傳後的結果一致。

預期結果：

1. **分數穩定性：**系統的評分結果應該穩定，分數不應該有大幅波動（例如：波動範圍應該控制在 1 分以內）。另外，分數應該在合理範圍內，例如：不能低於 0，且不能超過滿分 25。
2. **歷史紀錄完整性：**歷史紀錄應該完整保存每次上傳的紀錄，包含時間戳、原文版本、評分結果與扣分細項。
3. **評分細項一致性：**每次上傳的分數和扣分細項應該保持一致，並且無明顯的差異。

測試案例：

測試項目	測試結果
分數是否穩定	是
歷史紀錄是否完整	是
每次上傳的結果與歷史紀錄一致	是
歷史紀錄排序是否正確	是

邏輯檢查點：

1. 已確保同一篇文章的分數在多次上傳中穩定不變，波動範圍應該控制在合理範圍內，並且相當內容的文章可以得到相似的分數，並且針對各自評論語法結構等等。