|  |
| --- |
| BỘ GIÁO DỤC VÀ ĐÀO TẠO  Hình dạng**TRƯỜNG ĐẠI HỌC CMC**  Ảnh có chứa Phông chữ, biểu tượng, Đồ họa, văn bản  Mô tả được tạo tự động  **ĐỒ ÁN CHUYÊN NGÀNH**  **TRANG WEB CUNG CẤP DỊCH VỤ OCR( Tài liệu mô tả API)**  KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG  Giảng viên hướng dẫn :Ngô Hoàng Huy  Sinh viên thực hiện :-Nguyễn Đình Việt  -Phùng Quang Trà  -Phạm Đình Đức  Mã số sinh viên : BIT220169  Lớp : 22IT2  **Hà Nội, 2024** |
|  |

1. API USER

**1. Đăng ký người dùng**

**Endpoint:** POST /register

**Mô tả:** Đăng ký người dùng mới. Kiểm tra xem email đã tồn tại hay chưa. Nếu chưa, tạo người dùng mới và gửi email xác minh.

**Yêu cầu:**

* **Body:** { email: String, password: String }

**Phản hồi:**

* **Thành công:** 201 Created với thông điệp "Registration successful. Please check your email for verification."
* **Thất bại:** 400 Bad Request với thông điệp "Email already registered" nếu email đã tồn tại.

**2. Đăng nhập người dùng**

**Endpoint:** POST /login

**Mô tả:** Đăng nhập người dùng và cấp một token JWT.

**Yêu cầu:**

* **Body:** { email: String, password: String }

**Phản hồi:**

* **Thành công:** 200 OK với thông tin { token: String, userId: String, email: String, userRole: String }
* **Thất bại:** 401 Unauthorized với thông điệp "Invalid email or password"

**3. Lấy thông tin người dùng**

**Endpoint:** GET /user

**Mô tả:** Lấy thông tin chi tiết của người dùng hiện tại. Yêu cầu token JWT phải hợp lệ.

**Phản hồi:**

* **Thành công:** 200 OK với thông tin người dùng.
* **Thất bại:** 404 Not Found nếu không tìm thấy người dùng hoặc 500 Internal Server Error nếu có lỗi.

**4. Lấy tất cả người dùng**

**Endpoint:** GET /users

**Mô tả:** Lấy danh sách tất cả người dùng.

**Phản hồi:**

* **Thành công:** 200 OK với danh sách người dùng.
* **Thất bại:** 404 Not Found nếu không có người dùng nào hoặc 500 Internal Server Error nếu có lỗi.

**5. Đếm số lượng người dùng**

**Endpoint:** GET /userscount

**Mô tả:** Đếm số lượng người dùng hiện tại.

**Phản hồi:**

* **Thành công:** 200 OK với số lượng người dùng.
* **Thất bại:** 404 Not Found nếu không có người dùng hoặc 500 Internal Server Error nếu có lỗi.

**6. Cập nhật thông tin người dùng**

**Endpoint:** PUT /users/:id

**Mô tả:** Cập nhật thông tin của người dùng theo ID.

**Yêu cầu:**

* **URL:** :id - ID của người dùng.
* **Body:** Dữ liệu cần cập nhật.

**Phản hồi:**

* **Thành công:** 200 OK với thông tin người dùng đã cập nhật.
* **Thất bại:** 404 Not Found nếu không tìm thấy người dùng hoặc 400 Bad Request nếu có lỗi.

**7. Thay đổi mật khẩu người dùng**

**Endpoint:** POST /users/change-password

**Mô tả:** Thay đổi mật khẩu cho người dùng.

**Yêu cầu:**

* **Body:** { userId: String, newPassword: String }

**Phản hồi:**

* **Thành công:** 200 OK với thông điệp "Password changed successfully"
* **Thất bại:** 404 Not Found nếu không tìm thấy người dùng hoặc 400 Bad Request nếu có lỗi.

**8. Tìm kiếm người dùng theo email**

**Endpoint:** GET /users/search

**Mô tả:** Tìm kiếm người dùng dựa trên email (không phân biệt chữ hoa chữ thường).

**Yêu cầu:**

* **Query:** email - Email để tìm kiếm.

**Phản hồi:**

* **Thành công:** 200 OK với danh sách người dùng khớp với tìm kiếm.
* **Thất bại:** 500 Internal Server Error nếu có lỗi.

1. API DOCUMENT

**1. Tải lên tài liệu**

**Endpoint:** POST /uploadDocument

**Mô tả:** Tải lên tài liệu từ người dùng. Tài liệu được lưu dưới dạng hình ảnh và mã hóa base64, sau đó lưu vào cơ sở dữ liệu.

**Yêu cầu:**

* **Body:** { userId: String } (Thông tin người dùng)
* **File:** Tệp tin hình ảnh (Chỉ hỗ trợ image/jpeg, image/png, image/gif)

**Phản hồi:**

* **Thành công:** 201 Created với thông tin tài liệu và documentId.
* **Thất bại:**
  + 400 Bad Request nếu không có tệp tin, userId bị thiếu, hoặc loại tệp không hợp lệ.
  + 500 Internal Server Error nếu có lỗi trong quá trình tải lên hoặc lưu tài liệu.

**2. Xử lý tài liệu**

**Endpoint:** POST /processDocument/:documentId

**Mô tả:** Xử lý tài liệu để nhận diện văn bản từ ảnh sử dụng Tesseract.js. Kiểm tra quyền sử dụng của người dùng trước khi thực hiện OCR.

**Yêu cầu:**

* **URL:** :documentId - ID của tài liệu cần xử lý.

**Phản hồi:**

* **Thành công:** 200 OK với kết quả nhận diện văn bản và thông tin kết quả.
* **Thất bại:**
  + 400 Bad Request nếu không có dữ liệu hình ảnh.
  + 403 Forbidden nếu đăng ký đã hết hạn hoặc không còn lượt sử dụng.
  + 404 Not Found nếu tài liệu hoặc người dùng không tồn tại.
  + 500 Internal Server Error nếu có lỗi trong quá trình xử lý tài liệu hoặc nhận diện văn bản.

**3. Lấy tài liệu của người dùng**

**Endpoint:** GET /getUserDocuments/:userId

**Mô tả:** Lấy danh sách tất cả tài liệu của một người dùng cụ thể.

**Yêu cầu:**

* **URL:** :userId - ID của người dùng.

**Phản hồi:**

* **Thành công:** 200 OK với danh sách tài liệu.
* **Thất bại:**
  + 400 Bad Request nếu userId không hợp lệ.
  + 404 Not Found nếu không tìm thấy tài liệu cho người dùng.
  + 500 Internal Server Error nếu có lỗi trong quá trình truy xuất dữ liệu.

**4. Xóa tài liệu**

**Endpoint:** DELETE /deleteDocument/:documentId

**Mô tả:** Xóa tài liệu theo ID và xóa tệp tin tương ứng khỏi hệ thống tệp.

**Yêu cầu:**

* **URL:** :documentId - ID của tài liệu cần xóa.

**Phản hồi:**

* **Thành công:** 200 OK với thông điệp "Document deleted successfully".
* **Thất bại:**
  + 404 Not Found nếu không tìm thấy tài liệu.
  + 500 Internal Server Error nếu có lỗi trong quá trình xóa tài liệu hoặc tệp tin.

1. API RESULT

**1. Tạo kết quả**

**Endpoint:** POST /results

**Mô tả:** Tạo một kết quả mới và lưu vào cơ sở dữ liệu.

**Yêu cầu:**

* **Body:** { documentId: String, ocrText: String, language: String }

**Phản hồi:**

* **Thành công:** 201 Created với thông điệp và thông tin kết quả.
* **Thất bại:** 500 Internal Server Error nếu có lỗi trong quá trình lưu kết quả.

**2. Đếm số lượng kết quả**

**Endpoint:** GET /resultscount

**Mô tả:** Đếm số lượng kết quả trong cơ sở dữ liệu.

**Phản hồi:**

* **Thành công:** 200 OK với số lượng kết quả.
* **Thất bại:** 500 Internal Server Error nếu có lỗi trong quá trình đếm kết quả.

**3. Lấy kết quả của người dùng**

**Endpoint:** GET /getUserResults/:userId

**Mô tả:** Lấy tất cả kết quả thuộc về người dùng cụ thể.

**Yêu cầu:**

* **URL:** :userId - ID của người dùng.

**Phản hồi:**

* **Thành công:** 200 OK với danh sách kết quả.
* **Thất bại:**
  + 400 Bad Request nếu userId không hợp lệ.
  + 204 No Content nếu không tìm thấy kết quả cho người dùng.
  + 500 Internal Server Error nếu có lỗi trong quá trình truy xuất kết quả.

**4. Lấy thông tin thanh toán của người dùng**

**Endpoint:** GET /payments/user/:userId

**Mô tả:** Lấy thông tin thanh toán của người dùng.

**Yêu cầu:**

* **URL:** :userId - ID của người dùng.

**Phản hồi:**

* **Thành công:** 200 OK với danh sách thanh toán.
* **Thất bại:**
  + 204 No Content nếu không tìm thấy thanh toán cho người dùng.
  + 500 Internal Server Error nếu có lỗi trong quá trình truy xuất thông tin thanh toán.

**5. Tải kết quả dưới dạng TXT**

**Endpoint:** GET /downloadResult/txt/:documentId

**Mô tả:** Tải kết quả dưới dạng tệp TXT.

**Yêu cầu:**

* **URL:** :documentId - ID của tài liệu cần tải kết quả.

**Phản hồi:**

* **Thành công:** 200 OK với tệp TXT chứa kết quả OCR.
* **Thất bại:**
  + 404 Not Found nếu không tìm thấy kết quả.
  + 500 Internal Server Error nếu có lỗi trong quá trình tạo hoặc gửi tệp TXT.

**6. Tải kết quả dưới dạng PDF**

**Endpoint:** GET /downloadResult/pdf/:documentId

**Mô tả:** Tải kết quả dưới dạng tệp PDF.

**Yêu cầu:**

* **URL:** :documentId - ID của tài liệu cần tải kết quả.

**Phản hồi:**

* **Thành công:** 200 OK với tệp PDF chứa kết quả OCR.
* **Thất bại:**
  + 404 Not Found nếu không tìm thấy kết quả.
  + 500 Internal Server Error nếu có lỗi trong quá trình tạo hoặc gửi tệp PDF.

**7. Tải kết quả dưới dạng DOCX**

**Endpoint:** GET /downloadResult/docx/:documentId

**Mô tả:** Tải kết quả dưới dạng tệp DOCX.

**Yêu cầu:**

* **URL:** :documentId - ID của tài liệu cần tải kết quả.

**Phản hồi:**

* **Thành công:** 200 OK với tệp DOCX chứa kết quả OCR.
* **Thất bại:**
  + 404 Not Found nếu không tìm thấy kết quả.
  + 500 Internal Server Error nếu có lỗi trong quá trình tạo hoặc gửi tệp DOCX.

1. API PAYMENT

 **Tạo thanh toán mới** (POST /payments):

* **Mô tả**: Tạo một bản ghi thanh toán mới và cập nhật thông tin đăng ký của người dùng.
* **Yêu cầu**: userId, amount, paymentMethod, subscriptionType, status.
* **Trả về**:
  + Nếu thành công: 201 Created với thông báo và thông tin thanh toán.
  + Nếu thất bại: 400 Bad Request hoặc 404 Not Found với thông báo lỗi tương ứng.

 **Lấy thông tin thanh toán theo ID** (GET /payments/:id):

* **Mô tả**: Lấy thông tin thanh toán dựa trên ID.
* **Trả về**:
  + Nếu thành công: 200 OK với thông tin thanh toán.
  + Nếu thất bại: 404 Not Found nếu không tìm thấy thanh toán hoặc 500 Internal Server Error nếu có lỗi.

 **Lấy danh sách tất cả thanh toán** (GET /payments):

* **Mô tả**: Lấy danh sách tất cả các thanh toán.
* **Trả về**:
  + Nếu thành công: 200 OK với danh sách thanh toán.
  + Nếu thất bại: 500 Internal Server Error nếu có lỗi.

 **Lấy tổng số lượng thanh toán** (GET /paymentscount):

* **Mô tả**: Lấy tổng số lượng thanh toán.
* **Trả về**:
  + Nếu thành công: 200 OK với số lượng thanh toán.
  + Nếu thất bại: 500 Internal Server Error nếu có lỗi.

 **Lấy tổng số tiền thanh toán** (GET /totalpayment):

* **Mô tả**: Tính tổng số tiền thanh toán đã thực hiện.
* **Trả về**:
  + Nếu thành công: 200 OK với tổng số tiền thanh toán.
  + Nếu thất bại: 500 Internal Server Error nếu có lỗi.

 **Lấy thanh toán theo khoảng thời gian** (GET /payment/date-range):

* **Mô tả**: Lấy các thanh toán trong một khoảng thời gian cụ thể và tổng số tiền kiếm được trong khoảng thời gian đó.
* **Yêu cầu**: startDate, endDate (trong URL query).
* **Trả về**:
  + Nếu thành công: 200 OK với danh sách thanh toán và tổng số tiền kiếm được.
  + Nếu thất bại: 500 Internal Server Error nếu có lỗi.

 **Lấy tổng số tiền thanh toán theo khoảng thời gian** (GET /totalpayment/date-range):

* **Mô tả**: Tính tổng số tiền thanh toán trong một khoảng thời gian cụ thể.
* **Yêu cầu**: startDate, endDate (trong URL query).
* **Trả về**:
  + Nếu thành công: 200 OK với tổng số tiền thanh toán trong khoảng thời gian.
  + Nếu thất bại: 400 Bad Request nếu dữ liệu ngày không hợp lệ hoặc 500 Internal Server Error nếu có lỗi.

 **Lấy thanh toán theo ID người dùng** (GET /payments/user/:userId):

* **Mô tả**: Lấy danh sách các thanh toán của một người dùng cụ thể dựa trên userId.
* **Trả về**:
  + Nếu thành công: 200 OK với danh sách thanh toán.
  + Nếu không tìm thấy thanh toán: 204 No Content.
  + Nếu thất bại: 500 Internal Server Error nếu có lỗi.

1. API FEEDBACK

 **Tạo phản hồi mới** (POST /feedback):

* **Mô tả**: Tạo một phản hồi mới từ người dùng.
* **Yêu cầu**: userId, content.
* **Trả về**:
  + Nếu thành công: 201 Created với thông báo và thông tin phản hồi mới.
  + Nếu thất bại: 400 Bad Request nếu thiếu userId hoặc content, hoặc 500 Internal Server Error nếu có lỗi.

 **Lấy danh sách tất cả phản hồi** (GET /feedback):

* **Mô tả**: Lấy danh sách tất cả các phản hồi, sắp xếp theo thời gian tải lên từ mới nhất đến cũ nhất.
* **Trả về**:
  + Nếu thành công: 200 OK với danh sách phản hồi.
  + Nếu thất bại: 500 Internal Server Error nếu có lỗi.

 **Cập nhật phản hồi theo ID** (PUT /feedback/:id):

* **Mô tả**: Cập nhật nội dung của phản hồi dựa trên ID.
* **Yêu cầu**: id (ID của phản hồi), content (nội dung mới).
* **Trả về**:
  + Nếu thành công: 200 OK với thông báo và thông tin phản hồi đã được cập nhật.
  + Nếu không tìm thấy phản hồi: 404 Not Found.
  + Nếu thất bại: 500 Internal Server Error nếu có lỗi.

 **Xóa phản hồi theo ID** (DELETE /feedback/:id):

* **Mô tả**: Xóa phản hồi dựa trên ID.
* **Yêu cầu**: id (ID của phản hồi).
* **Trả về**:
  + Nếu thành công: 200 OK với thông báo xác nhận phản hồi đã bị xóa.
  + Nếu không tìm thấy phản hồi: 404 Not Found.
  + Nếu thất bại: 500 Internal Server Error nếu có lỗi.

 **Lấy phản hồi theo khoảng thời gian** (GET /feedback/time-range):

* **Mô tả**: Lấy danh sách phản hồi trong một khoảng thời gian cụ thể.
* **Yêu cầu**: startDate, endDate (trong URL query).
* **Trả về**:
  + Nếu thành công: 200 OK với danh sách phản hồi trong khoảng thời gian đã chỉ định.
  + Nếu thất bại: 500 Internal Server Error nếu có lỗi.

1. Cách cài đặt
   * + Mở folder ocrapp trên visual studio code
     + Chạy lệnh npm install trên power shell
     + Chạy lện npm start
     + Mở một powershell khác
     + Chạy lệnh cd api để di chuyển thư mục
     + Chạy lệnh npm install
     + Chạy lệnh npm start