**1. Tạo Action để Hiển Thị File**: Tạo một action trong controller để hiển thị tệp từ thư mục cụ thể. Dưới đây là một ví dụ về cách thực hiện điều này:

|  |
| --- |
| [HttpGet("{filename}")]  public IActionResult GetImage(string filename)  {  // Đường dẫn đầy đủ đến thư mục chứa ảnh  var imagePath = Path.Combine(\_webHostEnvironment.ContentRootPath, "Uploads", "Images", filename);  if (System.IO.File.Exists(imagePath))  {  // Đọc nội dung của tệp và trả về dưới dạng Stream  var imageStream = System.IO.File.OpenRead(imagePath);  return File(imageStream, "image/jpeg"); // Thay đổi kiểu mime type tùy theo loại tệp  }  else  {  return NotFound(); // Trả về 404 nếu không tìm thấy tệp  }  } |

Trong ví dụ này, chúng ta sử dụng **System.IO.File** để đọc nội dung của tệp từ thư mục **Uploads/Images** và trả về nó dưới dạng dữ liệu hình ảnh (hoặc loại mime type tương ứng). Nếu tệp không tồn tại, chúng ta trả về mã trạng thái 404.

**2. Thiết lập Thư mục Tải Lên**: Đảm bảo rằng bạn đã thiết lập đúng thư mục để tải lên tệp ảnh trong ứng dụng của bạn. Trong đoạn code ở bước 1, chúng ta giả định rằng tệp ảnh được lưu trong thư mục **Uploads/Images**.

**3. Kiểm tra Authentication và Authorization**: Nếu cần thiết, bạn có thể kiểm tra quyền truy cập vào các tệp ảnh và thực hiện kiểm tra xác thực và ủy quyền trước khi cho phép người dùng truy cập vào chúng.

Khi bạn đã hoàn thành các bước trên, bạn có thể truy cập các tệp ảnh trong thư mục được chỉ định thông qua API của bạn bằng cách sử dụng URL tương ứng, ví dụ: **https://localhost:5001/api/images/yourimage.jpg**. Chắc chắn rằng bạn đã điều chỉnh đường dẫn URL để phản ánh thư mục và tên tệp ảnh cụ thể bạn muốn hiển thị.