

# Tóm tắt: Đơn giản hóa Upload Ánh Bìa và PDF cho Số Báo

## Vấn đề

Trước đây, chức năng tạo số báo mới gặp lỗi khi upload ảnh bìa và file PDF:

- URL không đúng
- Không tìm thấy file
- Quy trình upload phức tạp với nhiều bước

## Giải pháp Đã Triển Khai

Đã đơn giản hóa hoàn toàn quy trình upload bằng cách **tích hợp trực tiếp upload file vào API tạo/ sửa số báo.**

### 1. Cập nhật Backend API

#### API POST /api/issues (Tạo mới)

- Hỗ trợ multipart/form-data để nhận file trực tiếp
- Validation file type và size:
  - Ảnh bìa: JPEG, PNG, JPG, WebP (tối đa 5MB)
  - PDF: application/pdf (tối đa 50MB)
- Upload trực tiếp lên S3 với key có cấu trúc:

```
issues/covers/{year}-{number}-{timestamp}-{filename}
```
- Lưu cloud storage path vào database
- Văn tương thích ngược với JSON format

File: `/home/ubuntu/tapchi-hcqs/nextjs_space/app/api/issues/route.ts`

```

// Phát hiện content type
const contentType = request.headers.get('content-type') || '';

if (contentType.includes('multipart/form-data')) {
  const formData = await request.formData();

  // Lấy các trường form
  volumeNo = parseInt(formData.get('volumeNo') as string);
  number = parseInt(formData.get('number') as string);
  // ...

  // Upload ảnh bìa nếu có
  const coverImageFile = formData.get('coverImage') as File | null;
  if (coverImageFile && coverImageFile.size > 0) {
    // Validation
    // Upload to S3
    const buffer = Buffer.from(await coverImageFile.arrayBuffer());
    const key = `issues/covers/${year}-${number}-${timestamp}-${coverImageFile.name}`;
    coverImagePath = await uploadFile(buffer, key, coverImageFile.type);
  }

  // Upload PDF nếu có
  const pdfFile = formData.get('pdfFile') as File | null;
  // Tương tự...
}

```

## API PUT /api/issues/[id] (Cập nhật)

- ✅ Tương tự như POST API
- ✅ Hỗ trợ cả FormData và JSON
- ✅ Chỉ upload file mới khi người dùng chọn file mới
- ✅ Giữ nguyên file cũ nếu không có file mới

File: /home/ubuntu/tapchi-hcqs/nextjs\_space/app/api/issues/[id]/route.ts

## 2. Cập nhật Frontend Form

### Đơn giản hóa IssueForm Component

#### Trước đây:

- Sử dụng `react-hook-form` + `zod validation` phức tạp
- Upload file riêng biệt qua `/api/files/upload`
- Lưu URL sau khi upload xong
- Nhiều state và logic xử lý

#### Bây giờ:

- ✅ Loại bỏ `react-hook-form` và `zod` - đơn giản hơn
- ✅ Sử dụng controlled inputs với React state thông thường
- ✅ File được lưu trong component state
- ✅ Submit tất cả cùng một lúc với FormData
- ✅ Preview ảnh trực tiếp trước khi upload
- ✅ Hiển thị thông tin file PDF đã chọn (tên, kích thước)

File: /home/ubuntu/tapchi-hcqs/nextjs\_space/components/dashboard/issue-form.tsx

## Các Tính Năng Mới trong Form:

### 1. Chọn Ảnh Bìa:

```
```typescript
const handleCoverImageChange = (e: React.ChangeEvent) => {
  const file = e.target.files?.[0];
  // Validation
  setCoverImageFile(file);

  // Tạo preview ngay lập tức
  const reader = new FileReader();
  reader.onloadend = () => {
    setPreviewUrl(reader.result as string);
  };
  reader.readAsDataURL(file);
}
```
```

```

### 2. Chọn PDF:

```
typescript
const handlePdfChange = (e: React.ChangeEvent<HTMLInputElement>) => {
  const file = e.target.files?.[0];
  // Validation
  setPdfFile(file);
}
```

```

### 3. Submit Form:

```
```typescript
const handleSubmit = async (e: React.FormEvent) => {
  e.preventDefault();

  const formData = new FormData();
  formData.append('volumeNo', volumeNo);
  formData.append('number', number);
  // ... các trường khác

  // Thêm file nếu có
  if (coverImageFile) {
    formData.append('coverImage', coverImageFile);
  }
  if (pdfFile) {
    formData.append('pdfFile', pdfFile);
  }

  // Submit
  const response = await fetch(url, {
    method,
    body: formData // Không cần set Content-Type header
  });
}
```
```

```

### 3. UI/UX Cải Tiến

#### Ảnh Bìa:

- Preview ảnh ngay sau khi chọn (không cần upload riêng)
- Hiển thị tên file đã chọn
- Nút “Xóa” để bỏ chọn ảnh
- Hướng dẫn rõ ràng: “JPEG, PNG, WebP - Tối đa 5MB”
- Kích thước đề nghị: 800x1200px

#### PDF:

- Hiển thị thông tin file: tên + kích thước (MB)
- Icon PDF màu đỏ đặc trưng
- Nút “Xóa” để bỏ chọn file
- Hướng dẫn: “Tối đa 50MB”

#### Validation:

- Kiểm tra file type ngay khi chọn
- Kiểm tra kích thước file
- Thông báo lỗi rõ ràng bằng toast notification
- Các trường bắt buộc được đánh dấu với dấu sao đỏ (\*)

## Lợi Ích

---

#### Cho Người Dùng:

1. **Đơn giản hơn:** Chỉ cần chọn file và submit form
2. **Trực quan hơn:** Thấy preview ảnh và thông tin file ngay lập tức
3. **Nhanh hơn:** Không phải đợi upload riêng từng file
4. **Rõ ràng hơn:** Hướng dẫn và validation cụ thể

#### Cho Hệ Thống:

1. **Code đơn giản hơn:** Ít logic phức tạp, dễ maintain
2. **Ít request hơn:** Chỉ 1 request thay vì 3 (2 uploads + 1 create)
3. **Atomic operation:** Tạo số báo và upload file cùng lúc
4. **Backward compatible:** Vẫn hỗ trợ JSON format cũ

## Files Đã Sửa Đổi

---

#### Backend:

1. **app/api/issues/route.ts**
  - Added: `uploadFile` import from `@/lib/s3`
  - Modified: `POST` handler để hỗ trợ `FormData`
  - Added: File validation và upload logic
2. **app/api/issues/[id]/route.ts**
  - Added: `uploadFile` import from `@/lib/s3`
  - Modified: `PUT` handler để hỗ trợ `FormData`
  - Added: File validation và upload logic

## Frontend:

1. `components/dashboard/issue-form.tsx`
  - Removed: react-hook-form, zod dependencies
  - Removed: Complex validation schema
  - Removed: Separate file upload functions
  - Added: Simple controlled inputs
  - Added: File state management
  - Added: Image preview với FileReader
  - Added: PDF info display
  - Simplified: Submit logic với FormData

## Testing

---

- TypeScript compilation: **PASSED**
- Next.js build: **SUCCESSFUL**
- Dev server: **RUNNING**
- Checkpoint saved: “**Simplified issue cover/PDF upload**”

## Hướng Dẫn Sử Dụng

---

### Tạo Số Báo Mới:

1. Đăng nhập với tài khoản Admin/Tổng Biên Tập
2. Vào **Dashboard > Admin > Quản lý Số báo**
3. Click “**Thêm Số mới**”
4. Điền thông tin:
  - Tập (Volume) \* - bắt buộc
  - Số (Issue Number) \* - bắt buộc
  - Năm \* - bắt buộc
  - Tiêu đề (tùy chọn)
  - Mô tả (tùy chọn)
5. **Chọn Ảnh Bìa:**
  - Click nút “Chọn ảnh bìa”
  - Chọn file ảnh (JPEG/PNG/WebP, tối đa 5MB)
  - Xem preview ngay lập tức
6. **Chọn PDF:**
  - Click nút “Chọn file PDF”
  - Chọn file PDF (tối đa 50MB)
  - Xem thông tin file đã chọn
7. Nhập DOI (nếu có)
8. Chọn Ngày phát hành
9. Chọn Trạng thái: Nhập / Đã xuất bản
10. Click “**Tạo mới**”

### Sửa Số Báo:

1. Trong danh sách số báo, click nút **Sửa** (icon bút chì)
2. Form sẽ hiển thị thông tin hiện tại

3. Có thể:
  - Giữ nguyên ảnh bìa/PDF cũ
  - Hoặc chọn file mới để thay thế
4. Click “**Cập nhật**”

## Lưu Ý Kỹ Thuật

---

### File Storage:

- Tất cả file được upload lên AWS S3
- Cloud storage path được lưu trong database
- Format key: `issues/covers/{year}-{number}-{timestamp}-{filename}`
- Timestamp để tránh conflict file name

### Validation:

- **Ảnh bìa:**
  - Types: `image/jpeg`, `image/png`, `image/jpg`, `image/webp`
  - Max size: 5MB
- **PDF:**
  - Type: `application/pdf`
  - Max size: 50MB

### Backward Compatibility:

- API vẫn hỗ trợ JSON format cũ
- Nếu request có `Content-Type: application/json`, sẽ xử lý như trước
- Nếu có `multipart/form-data`, sẽ xử lý file upload

## Kết Luận

---

Đã hoàn thành thành công việc đơn giản hóa chức năng upload ảnh bìa và PDF cho số báo. Hệ thống giờ đây:

- Đơn giản và trực quan hơn cho người dùng
- Ít lỗi hơn (atomic operation)
- Dễ maintain hơn (less code, clearer logic)
- Hiệu suất tốt hơn (fewer requests)
- Validation đầy đủ (file type, size)
- Preview trực tiếp (better UX)

**Checkpoint đã được lưu:** “Simplified issue cover/PDF upload”

Ứng dụng đã sẵn sàng để triển khai!