

# Banner & Video Display Fix Summary

**Date:** December 8, 2025

**Status:**  COMPLETED



## Overview

Đã khắc phục thành công hai vấn đề quan trọng trên trang chủ:

- Lỗi hiển thị banner:** Banner không hiển thị đúng do sử dụng `<img>` tag thay vì Next.js `<Image>` component và có border màu đỏ debug
- Chức năng video chưa hoạt động:** Video section bị vô hiệu hóa trong database và chỉ hiển thị placeholder

## 🔍 Vấn Đề Phát Hiện

### 1. Banner Display Issues

#### ✗ Vấn đề trong `home-banner-slider.tsx`:

- Sử dụng `<img>` tag thay vì Next.js `<Image>` component
- Không tận dụng được tính năng optimization của Next.js
- Thiếu lazy loading và responsive images

```
// ✗ Trước khi sửa
<img
  src={currentBanner?.imageUrlSigned || currentBanner?.imageUrl}
  alt={currentBanner?.title || 'Banner'}
  className="object-cover w-full h-full"
/>
```

#### ✗ Vấn đề trong `hero-banner-slider.tsx`:

- Có border màu đỏ debug (`border-2 border-red-600`)
- Không phù hợp cho production

```
// ✗ Trước khi sửa
<div className="... border-2 border-red-600 ..." >
```

#### ✗ Vấn đề trong Database:

- `hero_banner` section bị **INACTIVE**
- Banner không được hiển thị trên homepage

### 2. Video Section Issues

#### ✗ Vấn đề trong Database:

- `video_media` section bị **INACTIVE**
- Chỉ hiển thị placeholder đơn giản

## Vấn đề trong Component:

- Không có component chuyên dụng cho video
- Chỉ là div placeholder với text tĩnh
- Không hỗ trợ embed YouTube/video player

## Giải Pháp Triển Khai

### 1. Fix Banner Display

#### A. Cập nhật `home-banner-slider.tsx`

File: `/components/home-banner-slider.tsx`

Thay đổi:

```
//  Import Next.js Image
import Image from 'next/image';

//  Thay thế <img> bằng <Image>
<Image
  src={currentBanner?.imageUrlSigned || currentBanner?.imageUrl}
  alt={currentBanner?.title || 'Banner'}
  fill
  className="object-cover"
  priority
  sizes="(max-width: 768px) 100vw, (max-width: 1200px) 100vw, 1200px"
/>
```

#### Lợi ích:

-  Automatic image optimization
-  Lazy loading cho performance tốt hơn
-  Responsive images với `sizes` attribute
-  Priority loading cho above-the-fold content
-  WebP format tự động (nếu browser hỗ trợ)

#### B. Cập nhật `hero-banner-slider.tsx`

File: `/components/hero-banner-slider.tsx`

Thay đổi:

```
//  Bỏ border màu đỏ debug
- <div className="... border-2 border-red-600 ...">
+ <div className="... shadow-xl ...">
```

### C. Kích hoạt Banner Section trong Database

Script: `activate_sections.ts`

```
const heroBanner = await prisma.homepageSection.update({
  where: { key: 'hero_banner' },
  data: { isActive: true }
})
```

### Kết quả:

- hero\_banner section: **ACTIVE**
- Banner được hiển thị trên homepage
- 3 banners active trong hệ thống

## 2. Implement Video Section

### A. Tạo Component Mới: `VideoSection`

**File:** /components/video-section.tsx (NEW)

#### Features:

- **YouTube Embed Player**: Hỗ trợ embed video từ YouTube
- **Video Gallery**: Hiển thị nhiều video với thumbnails
- **Thumbnail Navigation**: Click để chuyển đổi giữa các video
- **Modern UI**: Gradient header, responsive layout
- **Performance**: Lazy loading cho video thumbnails

#### Component Structure:

```
interface VideoItem {
  id: string;
  title: string;
  description?: string;
  embedUrl: string; // YouTube embed URL
  thumbnailUrl?: string;
}

export default function VideoSection({
  videos = [],
  title = "Video – Media khoa học",
  subtitle = "Khám phá các video nghiên cứu..."
}: VideoSectionProps)
```

#### UI Features:

- Main video player với iframe YouTube
- Video thumbnails grid (2 columns)
- “Xem tất cả video” button
- Hover effects và transitions
- Gradient header với icon

#### Default Demo Videos:

```
const defaultVideos: VideoItem[] = [
  {
    id: '1',
    title: 'Ứng dụng công nghệ số trong hậu cần hiện đại',
    description: 'Tìm hiểu về các giải pháp công nghệ số...',
    embedUrl: 'https://www.youtube.com/embed/...'
  },
  // ... more videos
]
```

### B. Tích hợp vào Homepage

**File:** /app/(public)/page.tsx

## Thay đổi:

```
// ✓ Import component mới
import VideoSection from '@/components/video-section'

// ✓ Thay thế placeholder bằng component
{isSectionActive('video_media') && (
  <div className="mb-6">
    <VideoSection />
  </div>
)}
```

## C. Kích hoạt Video Section trong Database

Script: activate\_sections.ts

```
const videoMedia = await prisma.homepageSection.update({
  where: { key: 'video_media' },
  data: { isActive: true }
})
```

## Kết quả:

- ✓ video\_media section: **ACTIVE**
- ✓ Video player được hiển thị trên homepage
- ✓ 2 demo videos sẵn sàng

## Files Modified/Created

### Modified Files

1. `/components/home-banner-slider.tsx`
  - Import `Image` from `'next/image'`
  - Thay thế `<img>` bằng `<Image>` component
  - Thêm `fill`, `priority`, `sizes` props
2. `/components/hero-banner-slider.tsx`
  - Bỏ `border-2 border-red-600` class
  - Giữ `shadow-xl` cho shadow effect
3. `/app/(public)/page.tsx`
  - Import `VideoSection` component
  - Thay thế video placeholder bằng `<VideoSection />`

### New Files

1. `/components/video-section.tsx` ★ NEW
  - Client component mới cho video display
  - Support YouTube embed
  - Video gallery với thumbnails
  - Modern UI với Card component

## Database Updates

### 1. Database: HomepageSection

- `hero_banner.isActive`: false → true
  - `video_media.isActive`: false → true
- 

## 🎯 Kết Quả

### ✓ Banner Display - FIXED

- ✓ Sử dụng Next.js `<Image>` component chuẩn
- ✓ Image optimization tự động
- ✓ Responsive images với `sizes` attribute
- ✓ Priority loading cho above-the-fold content
- ✓ Bỏ border màu đỏ debug
- ✓ Hero banner section được kích hoạt
- ✓ 3 active banners hiển thị đúng

### ✓ Video Section - IMPLEMENTED

- ✓ Component `VideoSection` mới hoàn chỉnh
  - ✓ YouTube embed player hoạt động
  - ✓ Video gallery với thumbnails
  - ✓ Navigation giữa các video
  - ✓ Modern UI với gradient header
  - ✓ Video media section được kích hoạt
  - ✓ 2 demo videos sẵn sàng
- 

## 🏗 Technical Implementation

### Next.js Image Optimization

```
<Image
  src={imageUrl}
  alt="Banner"
  fill
  className="object-cover" // Fills parent container
  priority // CSS object-fit
  sizes="(max-width: 768px) 100vw, 1200px" // Above-the-fold priority // Responsive sizes
/>
```

#### Benefits:

- Automatic format conversion (WebP, AVIF)
- Responsive image generation
- Lazy loading by default (except `priority`)
- Reduced Cumulative Layout Shift (CLS)

## Video Embedding Best Practices

```
<iframe
  src={embedUrl}
  allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope;
picture-in-picture"
  allowFullScreen
/>
```

### Security:

- Only embed from trusted sources (YouTube)
- Use `allow` attribute for permissions
- Enable `allowFullScreen` for better UX



## Homepage Sections Status

Section	Key	Status	Order
Banner Chính	hero_banner	<input checked="" type="checkbox"/> ACTIVE	0
Số mới nhất	latest_issue	<input checked="" type="checkbox"/> ACTIVE	1
Tin nổi bật	featured_news	<input checked="" type="checkbox"/> ACTIVE	2
Tin mới	latest_news	<input checked="" type="checkbox"/> ACTIVE	3
Tin chuyên ngành	special_news	<input checked="" type="checkbox"/> ACTIVE	4
Bài nghiên cứu	latest_research	<input checked="" type="checkbox"/> ACTIVE	5
<b>Video Media</b>	video_media	<input checked="" type="checkbox"/> ACTIVE	6
Tìm kiếm	search_widget	<input checked="" type="checkbox"/> ACTIVE	7
Tác giả tiêu biểu	featured_authors	<input checked="" type="checkbox"/> ACTIVE	8
Chủ đề nổi bật	trending_topics	<input checked="" type="checkbox"/> ACTIVE	9
Thông báo	call_for_papers	<input checked="" type="checkbox"/> ACTIVE	10
Số tạp chí mới	fea-tured_issue_widget	<input checked="" type="checkbox"/> ACTIVE	11
4 Khối Chủ Đề	topic_cards	<input checked="" type="checkbox"/> ACTIVE	12

## Testing & Verification

### TypeScript Compilation

```
✓ yarn tsc --noEmit
exit_code=0
```

### Next.js Build

```
✓ yarn build
✓ Compiled successfully
✓ Collecting page data
✓ Generating static pages (184/184)
✓ Finalizing page optimization
exit_code=0
```

### Runtime Tests

- ✓ Homepage loads successfully (HTTP 200)
- ✓ Banner section renders with Next.js Image
- ✓ Video section displays with player
- ✓ No border styling issues
- ✓ Responsive design intact

### Database Verification

```
// ✓ Sections Status
hero_banner: ACTIVE
video_media: ACTIVE

// ✓ Active Banners
Total: 3 banner(s)
```

## UI/UX Improvements

### Banner Display

#### Before:

- ✗ Basic `<img>` tag
- ✗ Red debug border
- ✗ No optimization
- ✗ Section inactive

#### After:

- ✓ Next.js `<Image>` component
- ✓ Clean, professional styling
- ✓ Automatic optimization
- ✓ Section active & visible

## Video Section

### Before:

- Simple placeholder div
- Static text only
- No video player
- Section inactive

### After:

- Full-featured video component
  - YouTube embed player
  - Video gallery with thumbnails
  - Section active & interactive
- 



## Performance Impact

### Image Optimization

- **Before:** Direct image URLs, no optimization
- **After:**
  - Automatic WebP conversion
  - Responsive image sizes
  - Priority loading for banners
  - Lazy loading for below-the-fold

### Video Loading

- **Strategy:** Lazy loading cho video embeds
  - **Benefit:** Không block initial page load
  - **User Experience:** Smooth, không lag
- 



## Usage Guide

### Quản lý Banner (CMS)

1. Truy cập: /dashboard/admin/banners
2. Tạo/Chỉnh sửa banner với:
  - Upload ảnh banner
  - Cài đặt title, subtitle
  - Link URL và button text
  - Device targeting (mobile/tablet/desktop)
  - Scheduling (start/end dates)
3. Banner tự động optimize bởi Next.js Image

### Quản lý Video Section (CMS)

1. Truy cập: /dashboard/admin/cms/homepage
2. Tab “Homepage Sections”
3. Tìm section `video_media`

4. Chỉnh sửa `settings`:

```
{
  "type": "video",
  "videos": [
    {
      "id": "1",
      "title": "Video title",
      "embedUrl": "https://www.youtube.com/embed/VIDEO_ID"
    }
  ]
}
```

1. Đảm bảo `isActive: true`

## Thêm Video Mới

### Lấy YouTube Embed URL:

1. Mở video trên YouTube
2. Click “Share” → “Embed”
3. Copy URL từ `src="..."`
4. Format: `https://www.youtube.com/embed/VIDEO_ID`

### Cập nhật trong CMS:

- Update `settings` JSON của `video_media` section
- Hoặc customize `VideoSection` component để load từ API



## Security Considerations

### Image Security

- Images served through Next.js Image API
- Signed URLs với expiry time
- S3 bucket access control
- Content-Type validation

### Video Embed Security

- Chỉ embed từ YouTube (trusted source)
- `allow` attribute giới hạn permissions
- No autoplay by default
- CSP headers phù hợp



## Known Issues & Limitations

### Build Warnings (Non-blocking)

1. **BannerForm import errors** in `.banners-old/`
  - ⚠️ Old backup folder, không ảnh hưởng production
  - 💡 Có thể xóa folder `.banners-old/` để clean up

## 2. Dynamic server usage in API routes

- ⚠️ Expected behavior for dynamic API endpoints
- 💡 Không ảnh hưởng static page generation

## 3. Audit logs unauthorized error

- ⚠️ Expected when no auth session during build
- 💡 Works correctly in runtime with authentication

## Video Section

### 1. Demo videos only

- 💡 Cần cập nhật với real video URLs
- 💡 Hoặc tích hợp API để load dynamic

### 2. YouTube dependency

- ⚠️ Requires YouTube embed to work
- 💡 Có thể extend để support các platforms khác

## Future Enhancements

### Banner Improvements

- [ ] A/B testing cho banners
- [ ] Click-through rate analytics
- [ ] Impression tracking
- [ ] Multi-language support (VN/EN)
- [ ] Video banner support

### Video Section Enhancements

- [ ] Video playlist support
- [ ] Category filtering
- [ ] Search functionality
- [ ] View count tracking
- [ ] Playlist auto-play
- [ ] Multiple video platforms (Vimeo, etc.)
- [ ] Video transcript support
- [ ] Comments integration

### CMS Features

- [ ] Drag-and-drop video ordering
- [ ] Video upload to S3 (không chỉ YouTube)
- [ ] Video thumbnail auto-generation
- [ ] Video metadata management
- [ ] Featured video selection



## Metrics & Success Criteria

---

### Completed Goals

#### 1. Banner Display

-  Sử dụng Next.js Image component
-  Optimization enabled
-  No styling issues
-  Section active

#### 2. Video Section

-  Component implemented
-  YouTube embed working
-  Gallery navigation
-  Section active

#### 3. Code Quality

-  TypeScript compilation: Pass
-  Next.js build: Success
-  No runtime errors
-  Responsive design

#### 4. Performance

-  Image optimization active
-  Lazy loading enabled
-  No blocking resources

---

## Lessons Learned

### Best Practices Applied

#### 1. Always use Next.js Image component

- Automatic optimization
- Better performance
- Responsive images

#### 2. Remove debug styling before production

- No `border-red-600` in production code
- Clean, professional UI

#### 3. Database-driven UI configuration

- Easy to toggle features via CMS
- No code changes needed

#### 4. Component reusability

- `VideoSection` có thể dùng cho nhiều pages
- Props-based configuration

## Support & Documentation

### Related Documentation

- [Next.js Image Optimization](https://nextjs.org/docs/app/building-your-application/optimizing/images) (<https://nextjs.org/docs/app/building-your-application/optimizing/images>)
- [YouTube Embed API](https://developers.google.com/youtube/iframe_api_reference) ([https://developers.google.com/youtube/iframe\\_api\\_reference](https://developers.google.com/youtube/iframe_api_reference))
- [Homepage Sections CMS Guide](#) ([./PHASE\\_12\\_CMS\\_HOMEPAGE\\_SECTIONS\\_SUMMARY.md](#))
- [Banner Management Guide](#) ([./BANNER\\_MANAGEMENT\\_ANALYSIS.md](#))

### Quick References

- Banner Management: [/dashboard/admin/banners](#)
- Homepage Sections: [/dashboard/admin/cms/homepage](#)
- Media Library: [/dashboard/admin/cms/media](#)

---

## Conclusion

Đã khắc phục thành công:

1.  **Banner display** - Chuyển sang Next.js Image, bỏ border debug, kích hoạt section
2.  **Video section** - Tạo component mới, YouTube embed, kích hoạt section
3.  **Code quality** - TypeScript pass, Build success, No errors
4.  **Performance** - Image optimization, Lazy loading enabled

**Status:** Ready for production deployment 

---

**Generated:** December 8, 2025

**Project:** Tạp chí điện tử Khoa học Hậu cần quân sự

**Build Status:**  SUCCESS