**BÁO CÁO CÔNG VIỆC**

**Công việc số:** 64

**Mô tả công việc:** Viết chương trình đọc ảnh Barcode

**Người thực hiện:** Hạ Quang Dũng

**Ngày bắt đầu:** 03/01/2025

**Ngày kết thúc:** 03/01/2025

***Ghi chú:***

**NỘI DUNG TÀI LIỆU**

[**1. TẠO MÃ BARCODE VÀ QR CODE TRONG ỨNG DỤNG C# 2**](#_yiczw1gcr0h0)

[**2. Chương trình tạo mã QR dựa theo file DLL 2**](#_ytyue9kzsw5q)

[2.1. Cài đặt các thư viện cần thiết 2](#_10qp06jpzp46)

[2.2. Hướng dẫn mã nguồn 5](#_qk24wx4auvaq)

[2.3. Kết quả 6](#_t6dpnpe94epj)

[**3. Chương trình tạo mã QR sử dụng thư viện trong trình quản lý gói 6**](#_m0mbi5ysudg)

[3.1. Cài đặt các thư viện cần thiết 6](#_djm5s4d6698y)

[3.2. Hướng dẫn mã nguồn 7](#_8vub6wkbco1u)

[3.3. Kết quả 9](#_uz87dm3socnt)

# 

# 1. TẠO MÃ BARCODE VÀ QR CODE TRONG ỨNG DỤNG C#

Việc tích hợp mã Barcode và QRCode trong c# rất hữu ích cho các phần mềm ứng dụng trong thực tế. Đặc biệt là các phần mềm mua bán hàng hàng hóa và các phần mềm quản lý khác. Bài viết này sẽ hướng dẫn cụ thể cách tích hợp cả 2 dạng mã này.

**Mã Barcode là gì?**

Mã Barcode được dùng để mã hóa chuỗi dữ liệu thường là mã số. Barcode ứng dụng rất phổ biến trong thực tế với ưu điểm là quét với tốc độ cao. Thông thường, tia hồng ngoại được dùng chủ yếu để quét mã này. Ví dụ trong thanh toán hóa đơn ở siêu thị. Ngoài ra, có thể quét ở dạng hình ảnh. Tuy nhiên, khi dùng phần mềm để quét mã barcode dưới dạng hình ảnh thì tốc độ sẽ chậm hơn.

**Mã QRCode có ưu điểm gì?**

Mã QRCode ra đời sau mã Barcode nên có nhiều ưu điểm vượt trội. Đặc biệt là khả năng mã hóa được nhiều ký tự hơn Barcode. QRCode có thể mã hóa dữ liệu chuỗi unicode có dấu tiếng Việt trong khi mã barcode chỉ mã hóa được dữ liệu số hoặc chuỗi không dấu. Ngoài ra, QRCode còn cho khả năng sửa lỗi cao. Nghĩa là có thể đọc một mã QRCode bị hư hỏng một phần. Tuy nhiên, hạn chế của QRCode là không thể quét được bằng tia hồng ngoại mà chỉ có thể quét được ở dạng hình ảnh. Do đó, tốc độ đọc mã này sẽ chậm hơn barcode. Đây là lý do tại sao hiện nay mã Barcode vẫn còn sử dụng rộng rãi trong việc mã hóa sản phẩm và thanh toán hóa đơn mua bán hàng hóa.

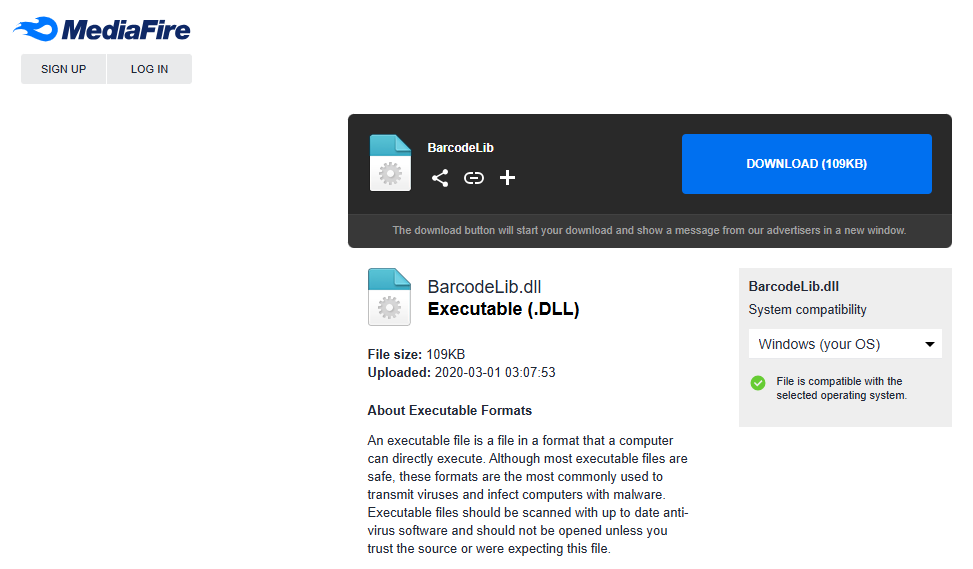
# 2**. Chương trình tạo mã QR dựa theo file DLL**

## **2.1. Cài đặt các thư viện cần thiết**

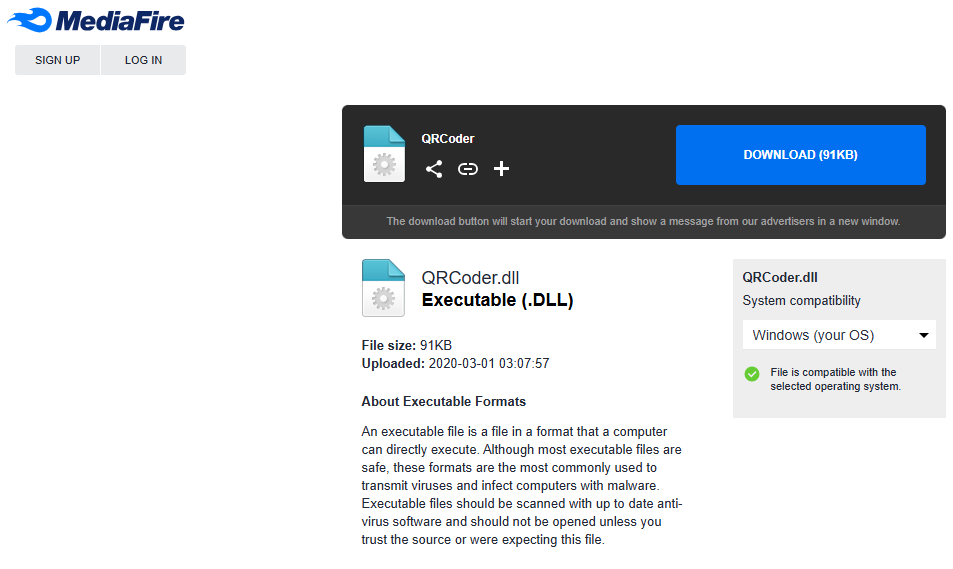
Thư viện sử dụng:

Trong lập trình C#, để sử dụng các dạng mã này cần phải sử dụng thư viện tương ứng. Cụ thể đối với:

* Mã Barcode thì sử dụng thư viện BarcodeLib ([download](http://www.mediafire.com/file/tw0nl5ak0upq1z3/BarcodeLib.dll/file)).

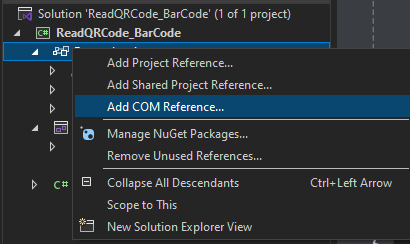


* Mã QRCode thì sử dụng thư viện QRCoder ([download](http://www.mediafire.com/file/z7hwaoywj0qp9yk/QRCoder.dll/file)).

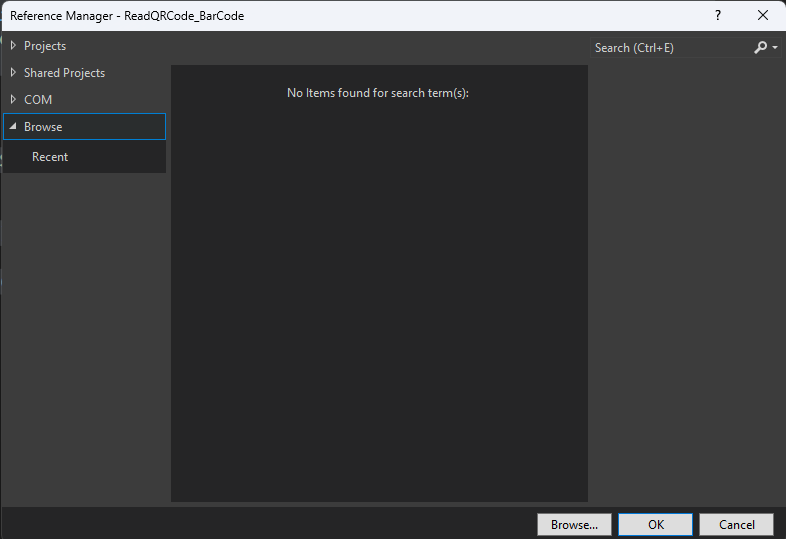


Tiến hành cài đặt các file DLL xuống dự án của bạn, và tiếp tục thực hiện các bước dưới đây:

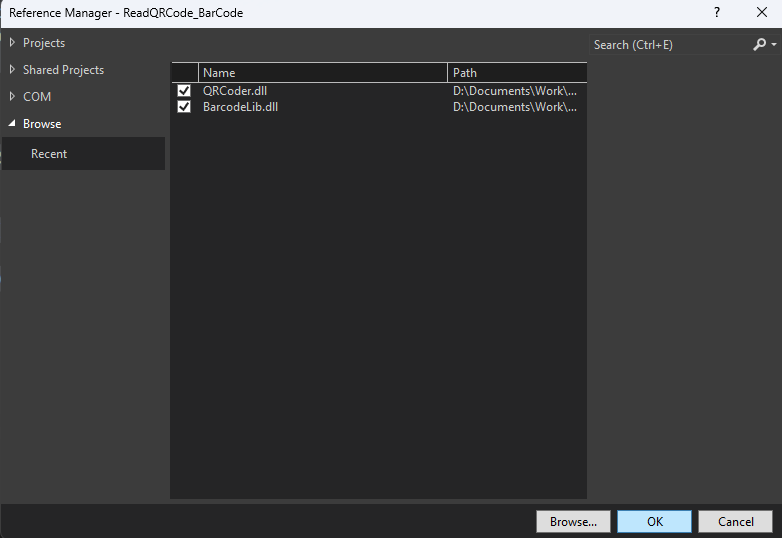
* Bước 1: Tạo dự án
* Bước 2: Thêm các thư viện vừa tải về
  + 2.1: Add COM Reference…



* + 2.1: Chọn Browse và tìm đến các file vừa tải về



* + 2.3: Nhấn OK

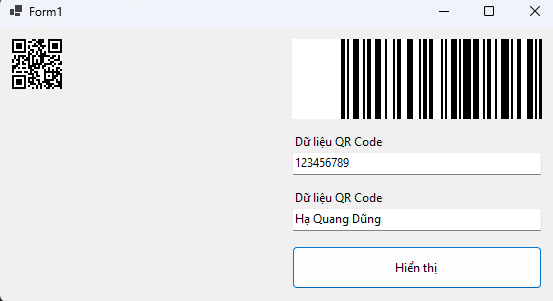


Như vậy, thư viện đã được thêm vào dự án thành công.

## **2.2. Hướng dẫn mã nguồn**

| **using** System.Windows.Forms; **using** BarcodeLib; **using** QRCoder;  **namespace** **ReadQRCode\_BarCode** {  **public** **partial** **class** **Form1** : **Form**  {  BarcodeLib.Barcode code128;  **public** **Form1**()  {  InitializeComponent();  code128 = **new** Barcode();  }   **private** **void** **btn\_Show\_Click**(**object** sender, EventArgs e)  {  Image barcode = code128.Encode(BarcodeLib.TYPE.CODE128, txb\_BarCode.Text);  pb\_BarCode.Image = barcode;   QRCodeGenerator qrGenerator = **new** QRCodeGenerator();  QRCode qrCode = **new** QRCode(qrGenerator.CreateQrCode(txb\_QRCode.Text, QRCodeGenerator.ECCLevel.Q));  *//QRCodeGenerator.ECCLevel.Q là mức chịu lỗi 25%; .L là 7%; .M là 15% và .H là trên 25%*  pb\_QRCode.Image = qrCode.GetGraphic(2, Color.Black, Color.White, false);  qrGenerator.Dispose();  qrCode.Dispose();  }  } } |
| --- |

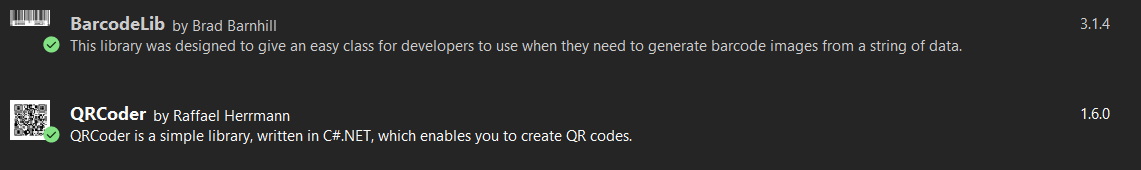
## **2.3. Kết quả**



# 3. Chương trình tạo mã QR sử dụng thư viện trong trình quản lý gói

## **3.1. Cài đặt các thư viện cần thiết**

Tiến hành cài đặt các thư viện sau trong trình quản lý gói của C#.



## **3.2. Hướng dẫn mã nguồn**

| **using** QRCoder; **using** BarcodeStandard; **using** SkiaSharp;  **namespace** **ReadQRCode\_BarCode** {  **public** **partial** **class** **Form1** : **Form**  {  **private** Barcode code128;   **public** **Form1**()  {  InitializeComponent();  code128 = **new** Barcode();  }   **private** **void** **btn\_Show\_Click**(**object** sender, EventArgs e)  {  **try**  {  *// Generate Barcode*  **if** (!**string**.IsNullOrWhiteSpace(txb\_BarCode.Text))  {  *// Tạo SKImage*  SKImage skImage = code128.Encode(BarcodeStandard.Type.Code128, txb\_BarCode.Text);   *// Chuyển đổi từ SKImage sang System.Drawing.Image*  **using** (SKBitmap skBitmap = SKBitmap.FromImage(skImage))  **using** (**var** ms = **new** System.IO.MemoryStream())  {  SKPixmap pixmap = skBitmap.PeekPixels();  pixmap.Encode(SKEncodedImageFormat.Png, 100).SaveTo(ms);   pb\_BarCode.Image = **new** Bitmap(ms);  pb\_BarCode.SizeMode = PictureBoxSizeMode.StretchImage;  }  }  **else**  {  MessageBox.Show("Please enter text for the barcode.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Warning);  }   *// Generate QR Code*  **if** (!**string**.IsNullOrWhiteSpace(txb\_QRCode.Text))  {  QRCodeGenerator qrGenerator = **new** QRCodeGenerator();  QRCodeData qrCodeData = qrGenerator.CreateQrCode(txb\_QRCode.Text, QRCodeGenerator.ECCLevel.Q);  QRCode qrCode = **new** QRCode(qrCodeData);   pb\_QRCode.Image = qrCode.GetGraphic(2, Color.Black, Color.White, false);  pb\_QRCode.SizeMode = PictureBoxSizeMode.StretchImage;  qrCode.Dispose();  qrGenerator.Dispose();  }  **else**  {  MessageBox.Show("Please enter text for the QR code.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Warning);  }  }  **catch** (Exception ex)  {  MessageBox.Show($"An error occurred: {ex.Message}", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);  }  }  } } |
| --- |

## **3.3. Kết quả**

