Tìm hiểu về thư viện GMap.Net trong Winform

# 1. Giới thiệu Thư viện GMap.Net

**GMap.NET** là một thư viện mã nguồn mở phổ biến được sử dụng để tích hợp bản đồ Google Maps vào ứng dụng Windows Forms (WinForms) của .NET. Thư viện này cung cấp các công cụ và API cho việc hiển thị và tương tác với bản đồ Google Maps trong ứng dụng của bạn.

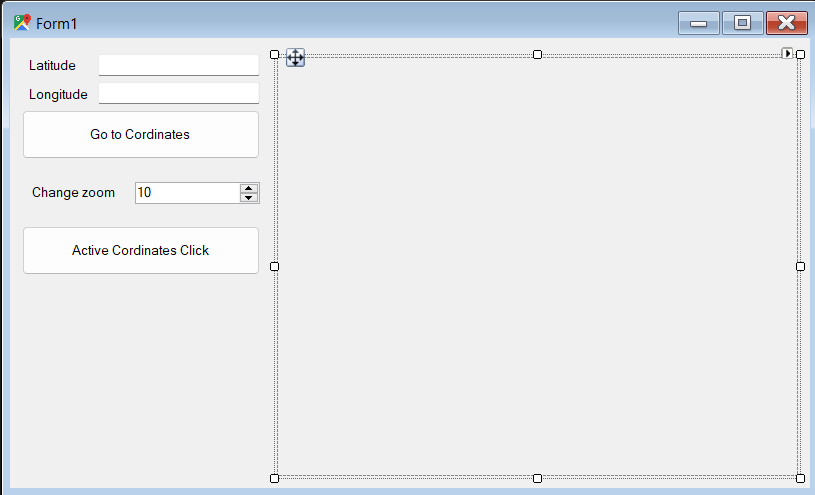
Dưới đây là một số điểm nổi bật của thư viện GMap.NET:

* **Hiển thị bản đồ**: GMap.NET cho phép bạn hiển thị bản đồ từ Google Maps trực tiếp trong ứng dụng WinForms của mình. Bạn có thể chỉ định tọa độ địa lý, địa chỉ hoặc tên địa điểm để hiển thị trên bản đồ.
* **Tùy chỉnh hiển thị**: Bạn có thể tùy chỉnh các loại bản đồ được hiển thị (ví dụ: bản đồ địa lý, bản đồ vệ tinh), mức độ phóng to và thu nhỏ, cũng như các thông tin khác như biểu đồ độ cao và dòng thông tin địa lý.
* **Tích hợp với dữ liệu**: GMap.NET cho phép bạn tích hợp dữ liệu tùy chỉnh vào bản đồ, bao gồm các điểm đánh dấu, đường đi, đa giác và hình ảnh nền.
* **Tương tác người dùng**: Bạn có thể cho phép người dùng tương tác với bản đồ bằng cách di chuyển, phóng to, thu nhỏ và thậm chí vẽ hoặc chỉnh sửa các đối tượng trên bản đồ.
* Hỗ trợ nhiều dịch vụ bản đồ: Bên cạnh Google Maps, GMap.NET cũng hỗ trợ các dịch vụ bản đồ khác như Bing Maps, OpenStreetMap, Yahoo Maps và nhiều dịch vụ khác.
* Dễ dàng tích hợp: Thư viện GMap.NET có cú pháp rõ ràng và dễ hiểu, cho phép bạn nhanh chóng tích hợp bản đồ vào ứng dụng WinForms của mình mà không cần nhiều kiến thức chuyên sâu về lập trình bản đồ.

Tóm lại, GMap.NET là một công cụ mạnh mẽ và linh hoạt cho phép bạn tích hợp các tính năng liên quan đến bản đồ Google Maps vào ứng dụng WinForms của mình một cách dễ dàng và hiệu quả.

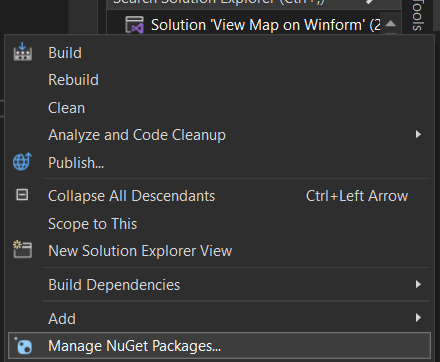
# 2. Ứng dụng Demo

## 2.1 Giao diện



## 2.2. Cài đặt gói cần thiết

Nhấp chuột phải vào dự án và chọn **Manage NuGet Packages…**

****

Tìm kiếm gói như hình bên dưới và tiến hành cài đặt



## 2.3. Hướng dẫn chi tiết mã nguồn

### 2.3.1. Khai báo thư viện

| using GMap.NET;  using GMap.NET.MapProviders;  using GMap.NET.WindowsForms; |
| --- |

* **GMap.NET**: Sử dụng các thành phần từ namespace GMap.NET. Namespace này chứa các lớp và phần mềm cần thiết để làm việc với bản đồ Google Maps trong ứng dụng của bạn.
* **GMap.NET.MapProviders**: Sử dụng các thành phần từ namespace GMap.NET.MapProviders. Namespace này chứa các lớp và phương thức để cung cấp dịch vụ bản đồ từ các nhà cung cấp khác nhau như Google Maps, Bing Maps, và OpenStreetMap
* **GMap.NET.WindowsForms**: Dòng này khai báo rằng mã sẽ sử dụng các thành phần từ namespace GMap.NET.WindowsForms. Namespace này chứa các lớp và thành phần để hiển thị bản đồ và tương tác với bản đồ trong ứng dụng Windows Forms.

### 2.3.2. Khai báo và khởi tạo ban đầu

| private GMapControl gMapControl1;  private int defaultZoom = 10;  public Form1()  {  InitializeComponent();  // Khởi tạo control GMapControl  gMapControl1 = new GMapControl();  // Đặt thuộc tính Dock của GMapControl thành Fill để nó lấp đầy Panel1  gMapControl1.Dock = DockStyle.Fill;  // Thêm GMapControl vào Panel1  panel1.Controls.Add(gMapControl1);  }  private void Form1\_Load(object sender, EventArgs e)  {  // Thiết lập nhà cung cấp bản đồ và tọa độ ban đầu của GMapControl  gMapControl1.MapProvider = GMapProviders.GoogleMap;  gMapControl1.Position = new PointLatLng(21.0285, 105.8542); // Set initial position  gMapControl1.MinZoom = 1;  gMapControl1.MaxZoom = 20;  gMapControl1.Zoom = defaultZoom;  } |
| --- |

### 2.3.3. Xử lý sự kiện di chuyển đến tọa độ dựa trên Kinh độ và Vĩ độ nhập vào

| private void goCordinates\_Click(object sender, EventArgs e)  {  gMapControl1.Position = new PointLatLng(Convert.ToDouble(latitude.Text), Convert.ToDouble(longitude.Text));  gMapControl1.Zoom = defaultZoom;  gMapControl1.Update();  gMapControl1.Refresh();  } |
| --- |

### 2.3.4. Xử lý sự kiện phóng to thu nhỏ khi thay đổi giá trị NumericUpDown

| private void zoomLevel\_ValueChanged(object sender, EventArgs e)  {  gMapControl1.Zoom = Convert.ToDouble(zoomLevel.Value);  gMapControl1.Update();  gMapControl1.Refresh();  } |
| --- |

### 2.3.5. Nhóm Hàm điều khiển và xử lý việc lấy tọa độ khi người dùng nhấp chọn vào 1 vị trí bất kỳ trên bản đồ

| void gmap\_MouseClick(object sender, MouseEventArgs e)  {  // Lấy tọa độ của điểm được nhấp chuột trên bản đồ  PointLatLng point = gMapControl1.FromLocalToLatLng(e.X, e.Y);  // Hiển thị tọa độ trên TextBox latitude và longitude  latitude.Text = point.Lat.ToString();  longitude.Text = point.Lng.ToString();  }  private bool isActive = false;  private void btnReturnCordinate\_Click(object sender, EventArgs e)  {  if (!isActive)  {  // Gắn sự kiện MouseClick cho control GMapControl  gMapControl1.MouseClick += gmap\_MouseClick;  btnReturnCordinate.Text = "Unactive Cordinates Click";  isActive = true;  }  else  {  // Hủy bỏ sự kiện MouseClick từ control GMapControl  gMapControl1.MouseClick -= gmap\_MouseClick;  btnReturnCordinate.Text = "Active Cordinates Click";  isActive = false;  }  } |
| --- |

## 2.4. Mã nguồn đầy đủ

| using System;  using System.Collections.Generic;  using System.ComponentModel;  using System.Data;  using System.Drawing;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.Windows.Forms;  using GMap.NET;  using GMap.NET.MapProviders;  using GMap.NET.WindowsForms;  namespace View\_Map\_on\_Winform  {  public partial class Form1 : Form  {  private GMapControl gMapControl1;  private int defaultZoom = 10;  public Form1()  {  InitializeComponent();  // Khởi tạo control GMapControl  gMapControl1 = new GMapControl();  // Đặt thuộc tính Dock của GMapControl thành Fill để nó lấp đầy Panel1  gMapControl1.Dock = DockStyle.Fill;  // Thêm GMapControl vào Panel1  panel1.Controls.Add(gMapControl1);  }  private void Form1\_Load(object sender, EventArgs e)  {  // Thiết lập nhà cung cấp bản đồ và tọa độ ban đầu của GMapControl  gMapControl1.MapProvider = GMapProviders.GoogleMap;  gMapControl1.Position = new PointLatLng(21.0285, 105.8542); // Set initial position  gMapControl1.MinZoom = 1;  gMapControl1.MaxZoom = 20;  gMapControl1.Zoom = defaultZoom;  }  private void goCordinates\_Click(object sender, EventArgs e)  {  gMapControl1.Position = new PointLatLng(Convert.ToDouble(latitude.Text), Convert.ToDouble(longitude.Text));  gMapControl1.Zoom = defaultZoom;  gMapControl1.Update();  gMapControl1.Refresh();  }  private void zoomLevel\_ValueChanged(object sender, EventArgs e)  {  gMapControl1.Zoom = Convert.ToDouble(zoomLevel.Value);  gMapControl1.Update();  gMapControl1.Refresh();  }  void gmap\_MouseClick(object sender, MouseEventArgs e)  {  // Lấy tọa độ của điểm được nhấp chuột trên bản đồ  PointLatLng point = gMapControl1.FromLocalToLatLng(e.X, e.Y);  // Hiển thị tọa độ trên TextBox latitude và longitude  latitude.Text = point.Lat.ToString();  longitude.Text = point.Lng.ToString();  }  private bool isActive = false;  private void btnReturnCordinate\_Click(object sender, EventArgs e)  {  if (!isActive)  {  // Gắn sự kiện MouseClick cho control GMapControl  gMapControl1.MouseClick += gmap\_MouseClick;  btnReturnCordinate.Text = "Unactive Cordinates Click";  isActive = true;  }  else  {  // Hủy bỏ sự kiện MouseClick từ control GMapControl  gMapControl1.MouseClick -= gmap\_MouseClick;  btnReturnCordinate.Text = "Active Cordinates Click";  isActive = false;  }  }  }  } |
| --- |

## 2.5. Kết quả nhận được

