

2

Lab

PHỤC VỤ MỤC ĐÍCH GIÁO DỤC
FOR EDUCATIONAL PURPOSE ONLY

Windows Service

Tạo Một Dịch Vụ Windows Service Trong C#

Thực hành môn Cơ chế hoạt động của mã độc

Tháng 3/2025

Lưu hành nội bộ

<Nghiêm cấm đăng tải trên internet dưới mọi hình thức>

A. TỔNG QUAN

1. Mục tiêu

- Giới thiệu Windows services trong .NET.
- Cách thức tạo Windows service trong .NET và C# qua Visual Studio.

2. Thời gian thực hành

- Thực hành tại lớp: 5 tiết tại phòng thực hành.
- Hoàn thành báo cáo kết quả thực hành: tối đa 7 ngày.

3. Kiến thức nền tảng

Windows service là ứng dụng chạy nền khởi động cùng hệ điều hành và được lên lịch để chạy một số dịch vụ. Windows service có thể được chạy được tự động hoặc thủ công với các chế độ pause, stop và restart.

Một số ví dụ chương trình Windows service: auto-update của Windows, kiểm tra emails, in văn bản, SQL Server agent, quét và lập chỉ mục tập tin và thư mục... Mở Task Manager và chọn tab Services sẽ thấy có hàng trăm service đã và đang chạy, đồng thời cũng có thể pause, stop và restart service thông qua giao diện.

Name	PID	Description	Status	Group
AarSvc		Agent Activation Runtime	Stopped	AarSvcGroup
AarSvc_89d8e8		Agent Activation Runtime_89d8e8	Stopped	AarSvcGroup
AdobeARMservice	3812	Adobe Acrobat Update Service	Running	
AJRouter		AllJoyn Router Service	Stopped	LocalServiceNet...
ALG		Application Layer Gateway Service	Stopped	
AppIDSvc		Application Identity	Stopped	LocalServiceNet...
Appinfo	11524	Application Information	Running	netsvcs
Apple Mobile Device Service	3792	Apple Mobile Device Service	Running	
AppMgmt		Application Management	Stopped	netsvcs
AppReadiness		App Readiness	Stopped	AppReadiness
AppVClient		Microsoft App-V Client	Stopped	
AppXSvc	11900	AppX Deployment Service (AppXSVC)	Running	wsappx
ArcHttpProxyServer	3864	ArcHttpProxyServer	Running	
AssignedAccessManagerSvc		AssignedAccessManager Service	Stopped	AssignedAccess...
AudioEndpointBuilder	2572	Windows Audio Endpoint Builder	Running	LocalSystemNet...
Audiosrv	3156	Windows Audio	Running	LocalServiceNet...
autotimesvc		Cellular Time	Stopped	autoTimeSvc
AxlnstSV		ActiveX Installer (AxlnstSV)	Stopped	AxlnstSVGroup
BcastDVRUserService		GameDVR and Broadcast User Service	Stopped	BcastDVRUserService
BcastDVRUserService_89d8e8		GameDVR and Broadcast User Service...	Stopped	BcastDVRUserService
BDDESVC		BitLocker Drive Encryption Service	Stopped	netsvcs
BFE	3500	Base Filtering Engine	Running	LocalServiceNo...
BITS	9472	Background Intelligent Transfer Service	Running	netsvcs
BluetoothUserService		Bluetooth User Support Service	Stopped	BthAppGroup

Cũng có thể tìm thấy tất cả các service đang chạy trên máy theo những cách sau:

- Đi đến Control Panel chọn “Services” bên trong “Administrative Tools”.
- Mở cửa sổ Run (Window + R) và nhập services.msc nhấn Enter.

B. CHUẨN BỊ MÔI TRƯỜNG

1. Cài đặt máy ảo Windows

**Bạn nào dùng Windows bỏ qua bước này*

Đối với các bạn sử dụng hệ điều hành Linux, MacOS thì cài đặt máy ảo Windows 10 (có sẵn Visual Studio) có sẵn [tại đây](#), hỗ trợ các trình ảo hoá VMWare, Hyper-V, VirtualBox, và Parallels.

2. Các tập tin chuẩn bị

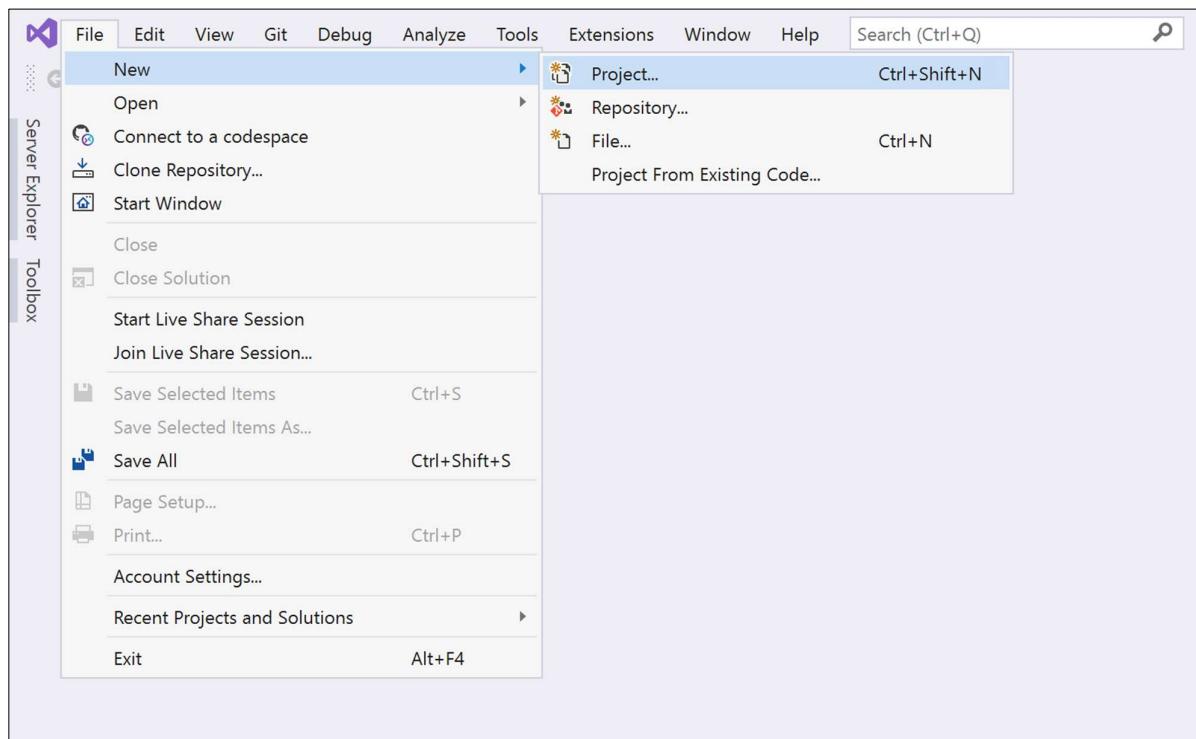
Tải tập tin và cài đặt Visual Studio Community cho hệ điều hành [Windows](#).

C. THỰC HÀNH

1. Tạo tập tin Windows service bằng C#

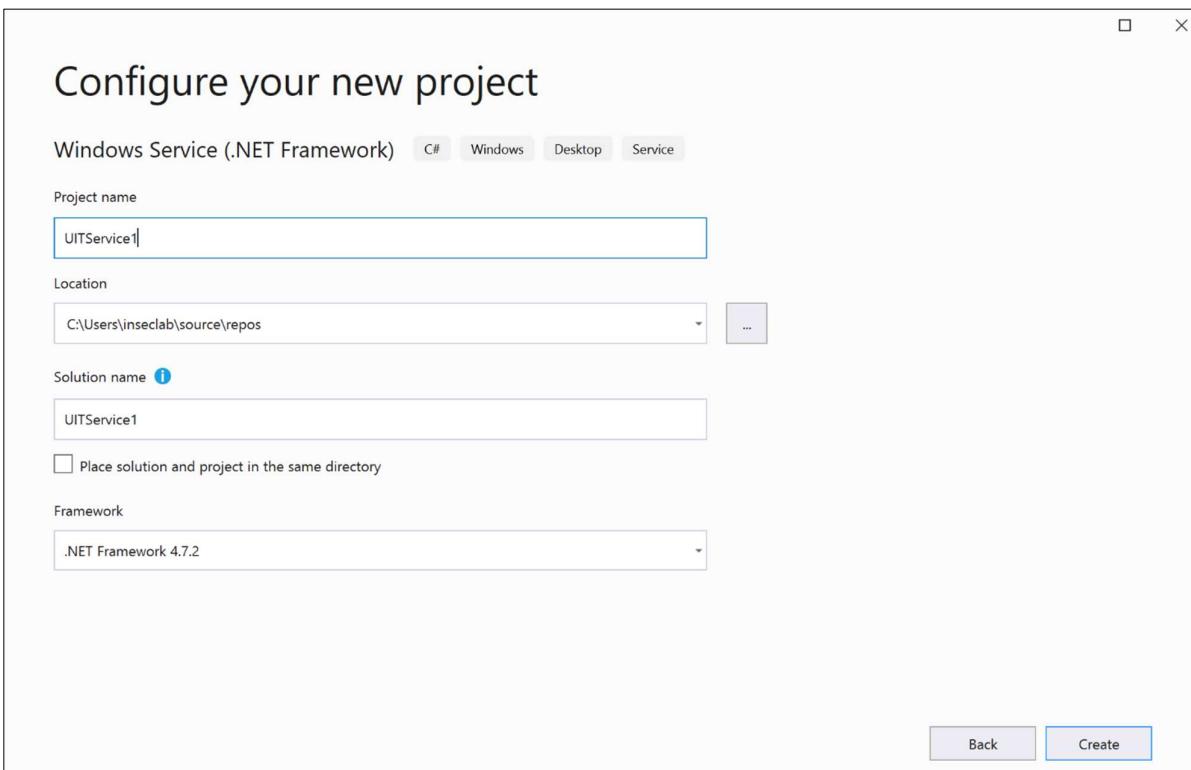
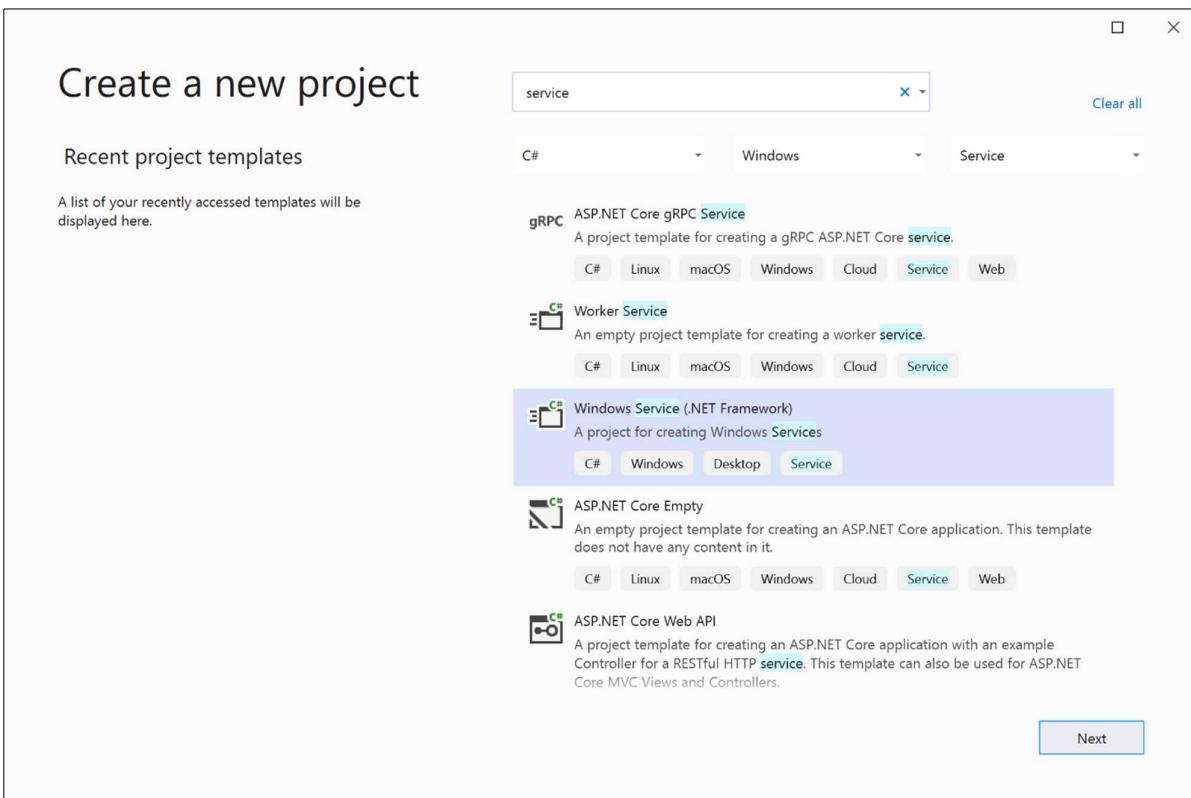
- **Bước 1:**

Mở Visual Studio, đi đến File > New và chọn Project. Tiếp tục chọn new project từ Dialog box và chọn “Window Service”, nhấp OK để hoàn tất.



- **Bước 2:**

Chọn các tùy chọn sau: C# -> "Windows Desktop" -> "Windows Service", đặt một cái tên (**MSSV**) rồi nhấn OK.

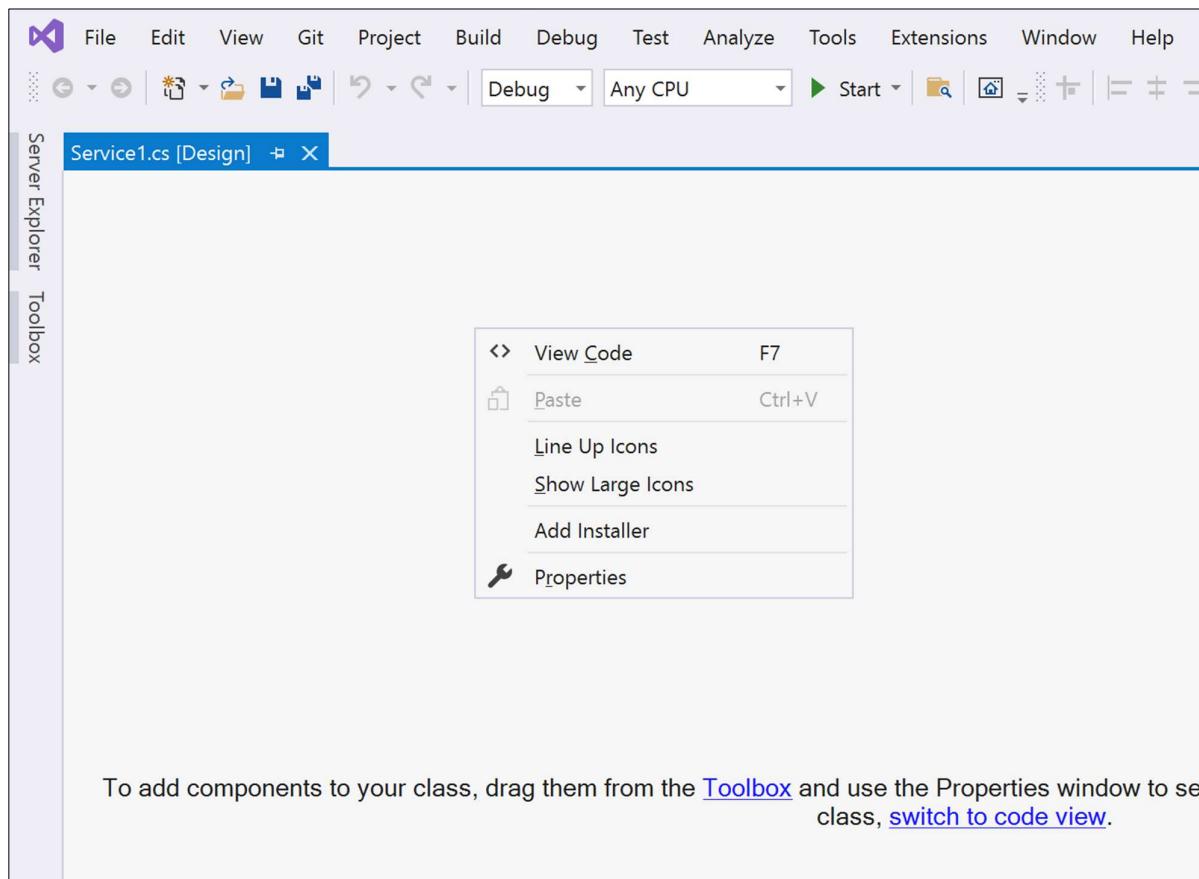


2. Thêm Installer cho Windows service

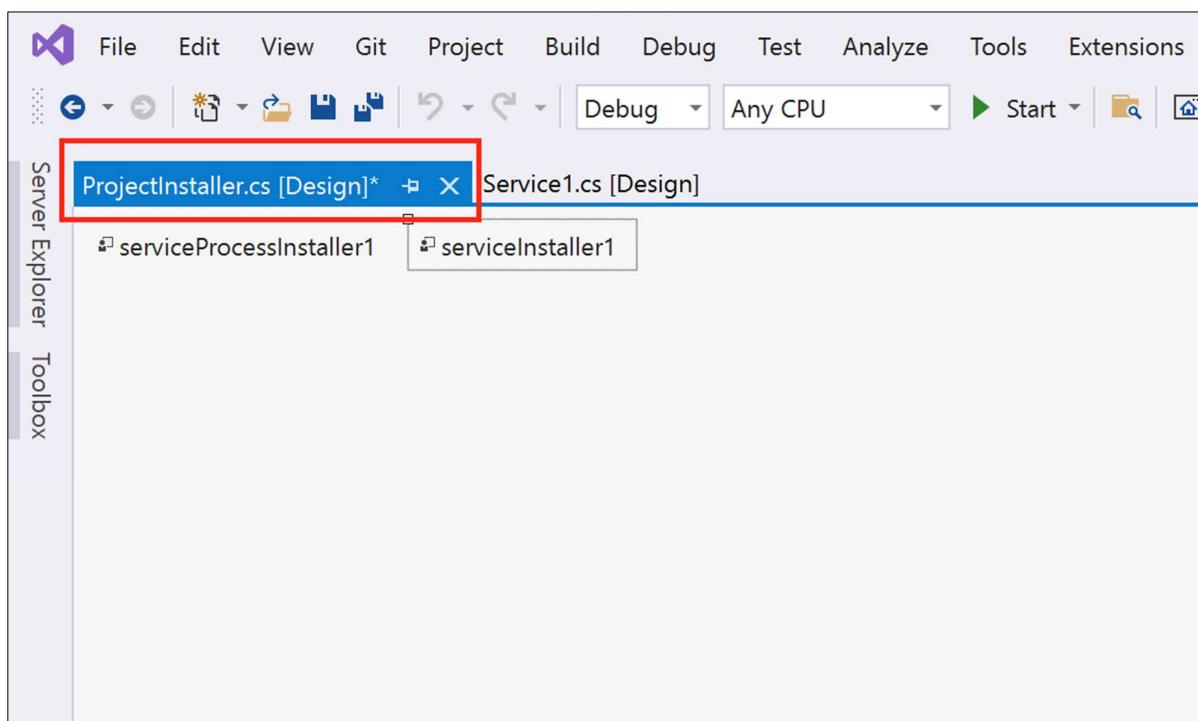
- **Bước 3:**

Trước khi có thể chạy Windows server, cần cài đặt Installer, đăng ký nó với Service Control Manager.

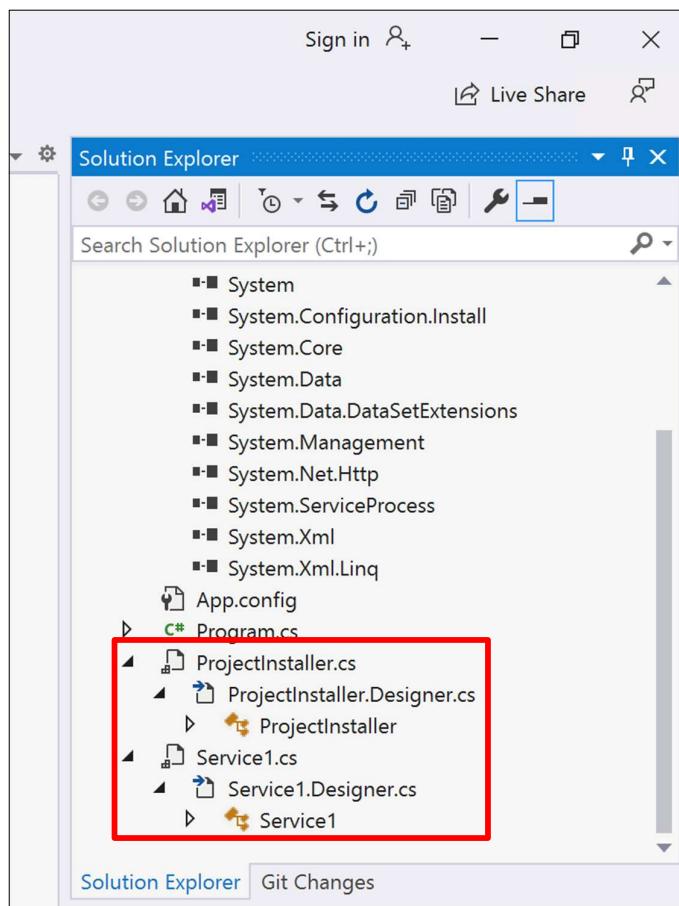
Nhập chuột phải vào vùng trống và chọn ““Add Installer”.



ProjectInstaller đã được thêm vào Project và ProjectInstakker.cs đã được tạo.

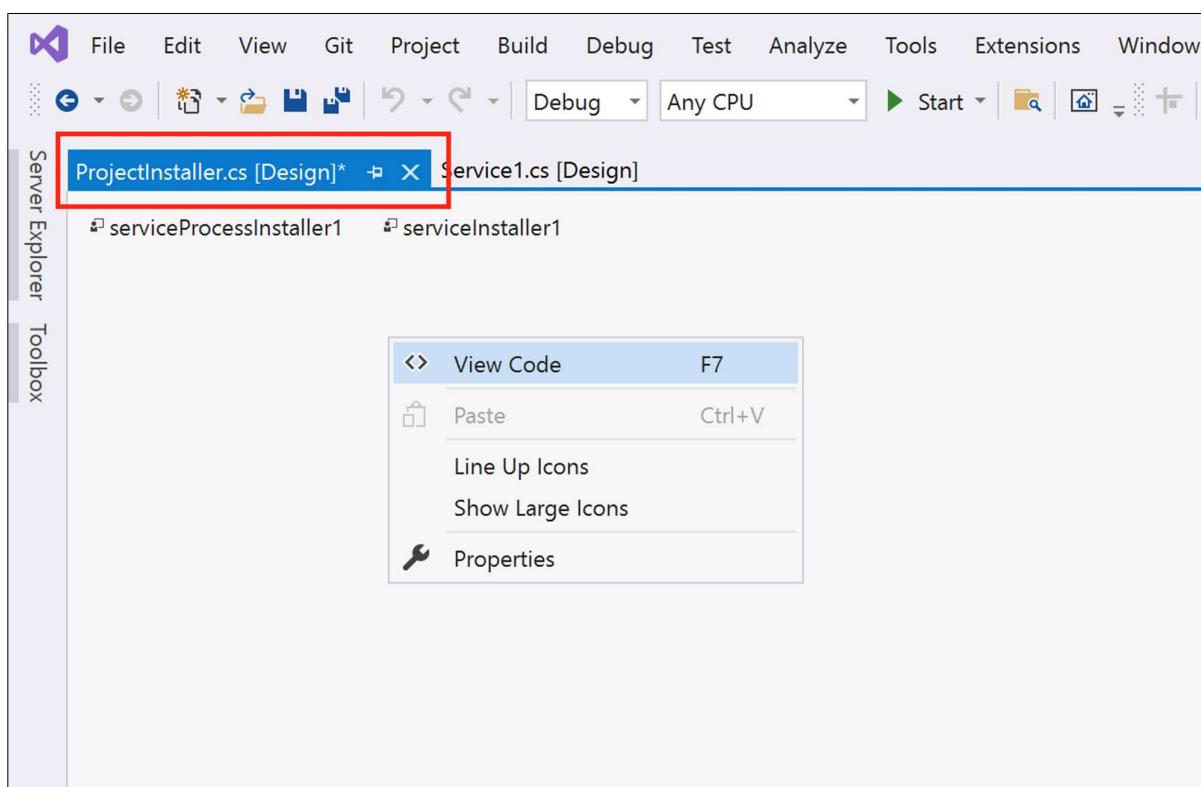


Solution Explore sẽ trông như thế này.



• Bước 4:

Nhập chuột phải vào vùng trống và chọn “View Code”

**• Bước 5:**

Trong Constructor tồn tại phương thức InitializeComponent.

InitializeComponent chứa các logic để khởi tạo giao diện người dùng được kéo thả trên Property Grid của Form Designer. Cho nên đừng bao giờ gọi các phương thức khác trước phương thức InitializeComponent.

```

ProjectInstaller.cs*  X  ProjectInstaller.cs [Design]*  Service1.cs [Design]
C# UITService1
1 using System;
2 using System.Collections;
3 using System.Collections.Generic;
4 using System.ComponentModel;
5 using System.Configuration.Install;
6 using System.Linq;
7 using System.Threading.Tasks;
8
9 namespace UITService1
10 {
11     [RunInstaller(true)]
12     public partial class ProjectInstaller : System.Configuration.Install.Installer
13     {
14         public ProjectInstaller()
15         {
16             InitializeComponent();
17         }
18     }
19 }
20

```

• Bước 6:

Trỏ vào InitializeComponent và nhấn F12 hoặc làm như thao tác trong hình để đi đến “go to definition”

View Designer	Shift+F7
Quick Actions and Refactorings...	Ctrl+.
Rename...	Ctrl+R, Ctrl+R
Remove and Sort Usings	Ctrl+R, Ctrl+G
<> View Code	F7
Peek Definition	Alt+F12
Go To Definition	F12
Go To Base	Alt+Home
Go To Implementation	Ctrl+F12
Find All References	Shift+F12
View Call Hierarchy	Ctrl+K, Ctrl+T
<i>Create Unit Tests</i>	

• Bước 7:

Thêm đoạn code bên dưới:

```

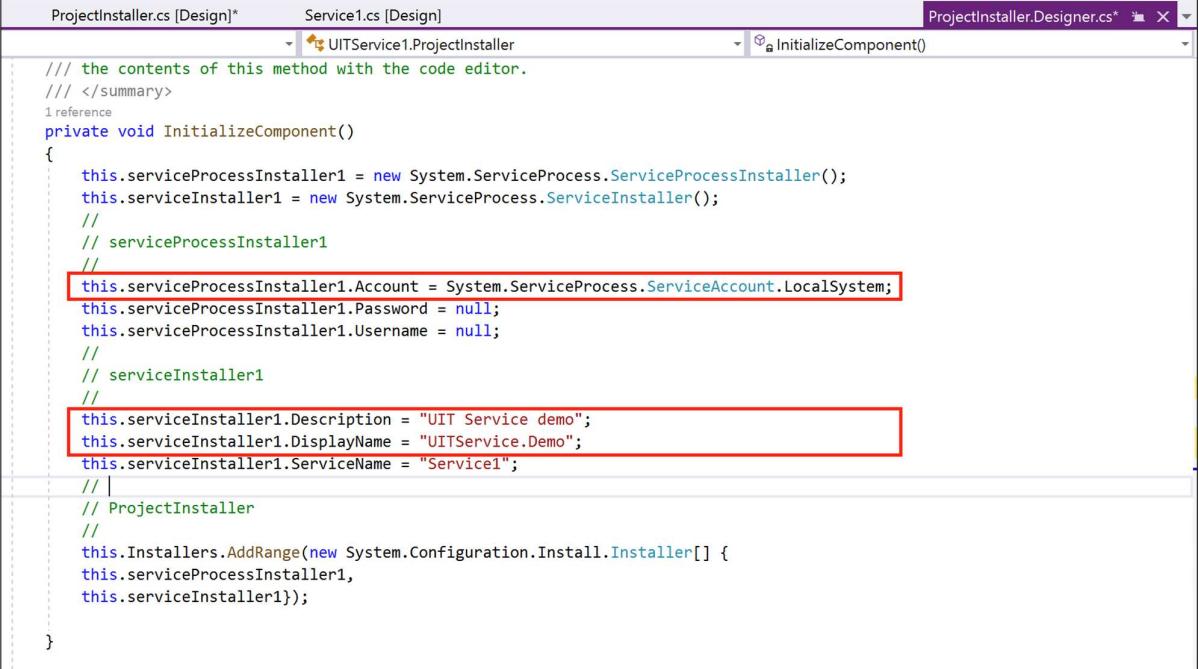
this.serviceProcessInstaller1.Account
System.ServiceProcess.ServiceAccount.LocalSystem;

```

=

Cũng có thể thay đổi tên hiển thị:

```
this.serviceInstaller1.Description = "UIT Service demo";
this.serviceInstaller1.DisplayName = "UITService.Demo";
```



```
ProjectInstaller.cs [Design]*          Service1.cs [Design]          ProjectInstaller.Designer.cs*  X
                                         UITService1.ProjectInstaller      InitializeComponent()

    /// the contents of this method with the code editor.
    /// </summary>
1 reference
private void InitializeComponent()
{
    this.serviceProcessInstaller1 = new System.ServiceProcess.ServiceProcessInstaller();
    this.serviceInstaller1 = new System.ServiceProcess.ServiceInstaller();
    //
    // serviceProcessInstaller
    //
    this.serviceProcessInstaller1.Account = System.ServiceProcess.ServiceAccount.LocalSystem;
    this.serviceProcessInstaller1.Password = null;
    this.serviceProcessInstaller1.Username = null;
    //
    // serviceInstaller1
    //
    this.serviceInstaller1.Description = "UIT Service demo";
    this.serviceInstaller1.DisplayName = "UITService.Demo";
    this.serviceInstaller1.ServiceName = "Service1";
    //
    // ProjectInstaller
    //
    this.Installers.AddRange(new System.Configuration.Install.Installer[] {
        this.serviceProcessInstaller1,
        this.serviceInstaller1});
}

}
```

- **Bước 8:**

Trong hướng dẫn này, sẽ viết công việc thực hiện timer và đoạn mã sẽ gọi dịch vụ trong một thời điểm nhất định. Có nghĩa là sẽ tạo một tập tin văn bản và ghi thời gian hiện tại vào tập tin văn bản bằng cách sử dụng dịch vụ.

Lab 2: Windows Services

```

1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Diagnostics;
6  using System.Linq;
7  using System.ServiceProcess;
8  using System.Text;
9  using System.Threading.Tasks;
10
11 namespace UITService1
12 {
13     public partial class Service1 : ServiceBase
14     {
15         public Service1()
16         {
17             InitializeComponent();
18         }
19
20         protected override void OnStart(string[] args)
21         {
22         }
23
24         protected override void OnStop()
25         {
26         }
27     }
28 }
29

```

Đoạn mã sẽ gọi dịch vụ sau mỗi 5 giây và tạo một thư mục nếu chưa có thư mục nào tồn tại và ghi thông điệp.

Service1.cs

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.IO;
using System.Linq;
using System.ServiceProcess;
using System.Text;
using System.Threading.Tasks;
using System.Timers;
namespace MyFirstService {
    public partial class Service1: ServiceBase {
        Timer timer = new Timer(); // name space(using System.Timers);
        public Service1() {
            InitializeComponent();
        }
        protected override void OnStart(string[] args) {
            WriteToFile("Service is started at " + DateTime.Now);
            timer.Elapsed += new ElapsedEventHandler(OnElapsedTime);
            timer.Interval = 5000; //number in milliseconds
            timer.Enabled = true;
        }
        protected override void OnStop() {
            WriteToFile("Service is stopped at " + DateTime.Now);
        }
    }
}

```

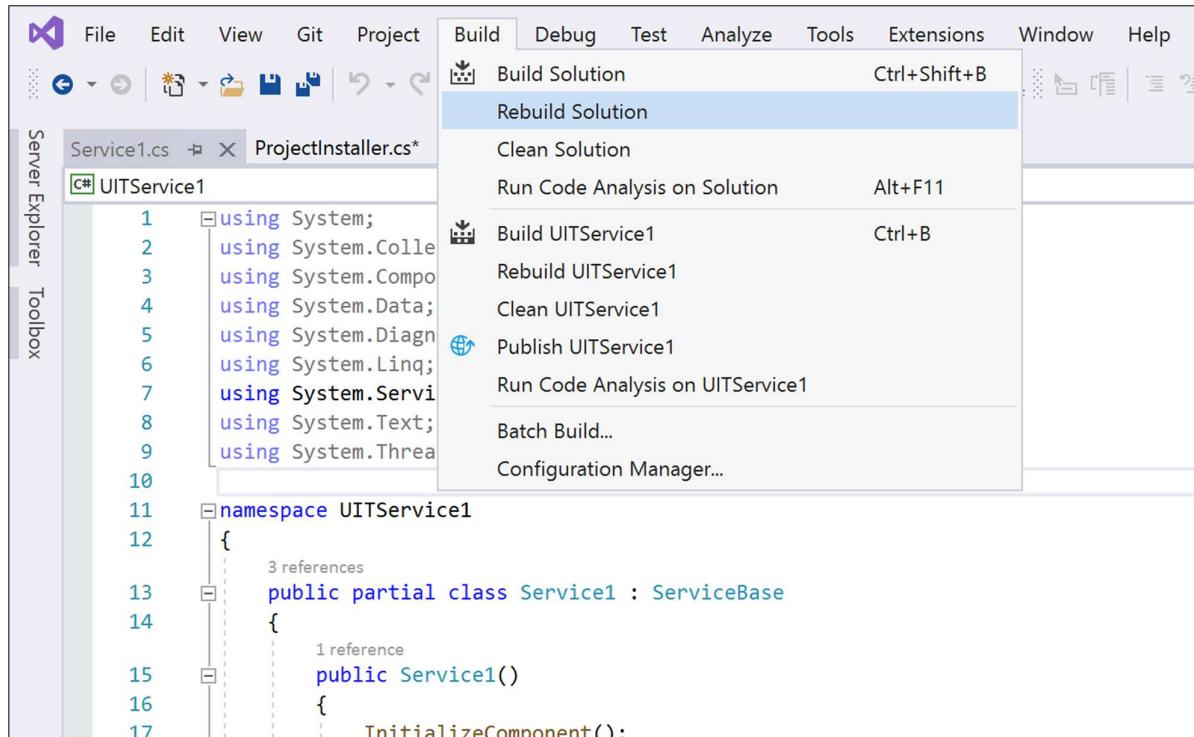
```
}

private void OnElapsedTimer(object source, ElapsedEventArgs e) {
    WriteToFile("Service is recall at " + DateTime.Now);
}

public void WriteToFile(string Message) {
    string path = AppDomain.CurrentDomain.BaseDirectory + "\\Logs";
    if (!Directory.Exists(path)) {
        Directory.CreateDirectory(path);
    }
    string filepath = AppDomain.CurrentDomain.BaseDirectory + "\\Logs\\ServiceLog_" +
DateTime.Now.Date.ToShortDateString().Replace('/', '_') + ".txt";
    if (!File.Exists(filepath)) {
        // Create a file to write to.
        using(StreamWriter sw = File.CreateText(filepath)) {
            sw.WriteLine(Message);
        }
    } else {
        using(StreamWriter sw = File.AppendText(filepath)) {
            sw.WriteLine(Message);
        }
    }
}
}
```

• Bước 9: Build ứng dụng

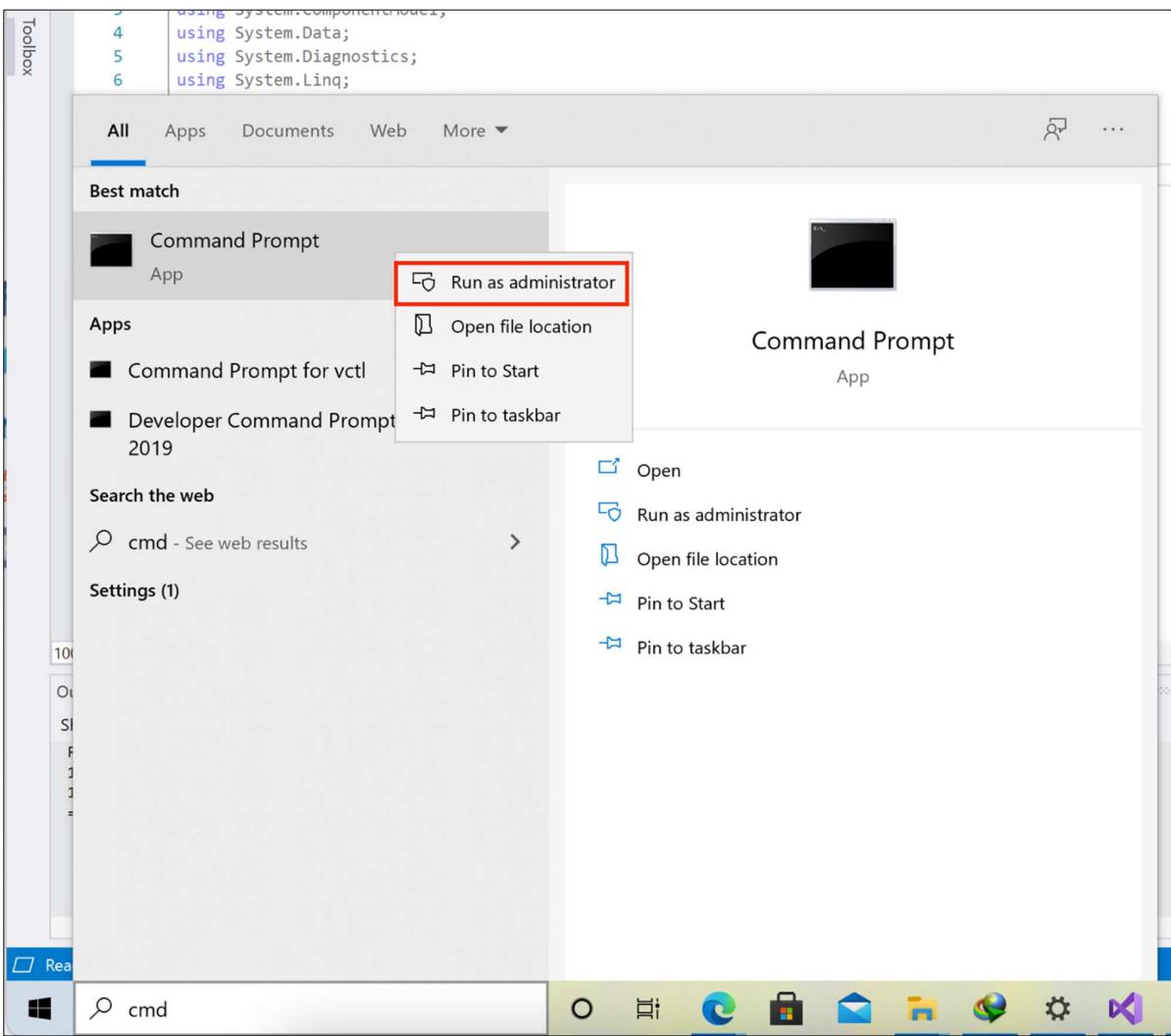
Chọn Build > Rebuild Solution.



```
Output
Show output from: Build
Rebuild started...
1>----- Rebuild All started: Project: UITService1, Configuration: Debug Any CPU -----
1>  UITService1 -> C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.exe
----- Rebuild All: 1 succeeded, 0 failed, 0 skipped -----
```

• **Bước 10:**

Chạy “Command Prompt” dưới quyền administrator.



• **Bước 11:**

Di chuyển bằng lệnh sau:

```
cd C:\Windows\Microsoft.NET\Framework\v4.0.30319
```

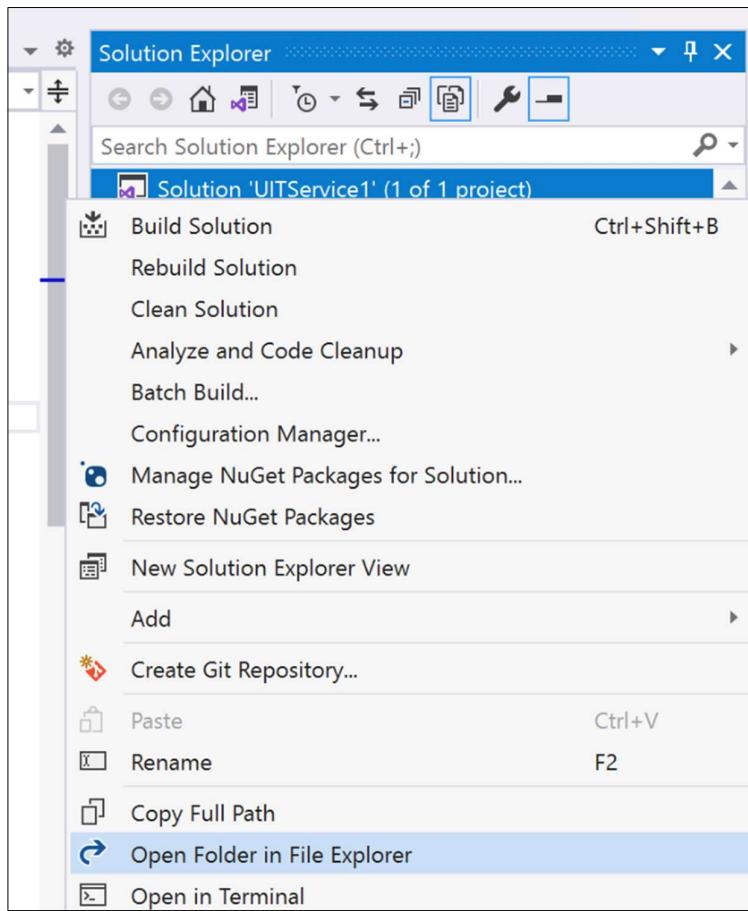
```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.19042.867]
(c) 2020 Microsoft Corporation. All rights reserved.

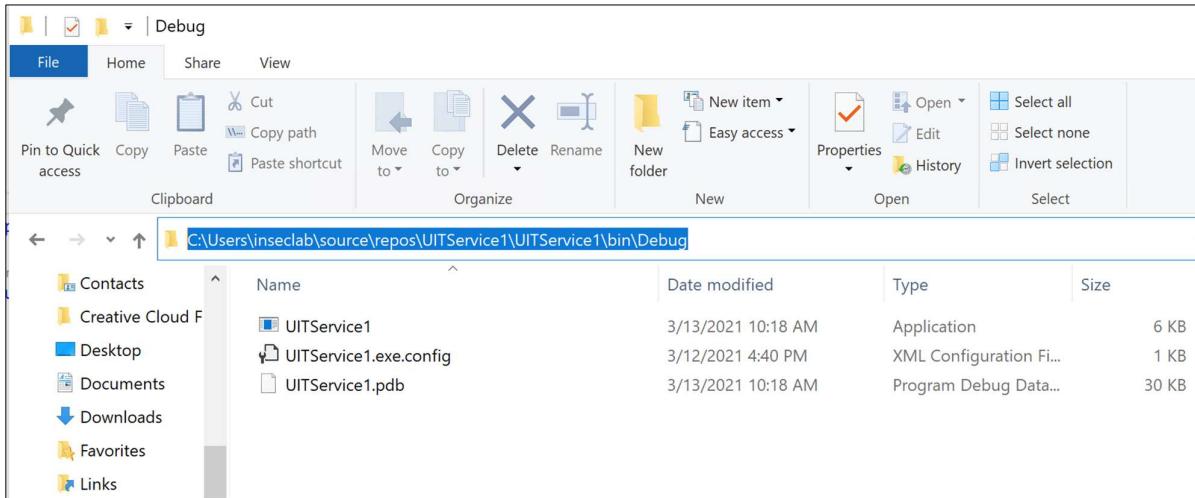
C:\WINDOWS\system32>cd C:\Windows\Microsoft.NET\Framework\v4.0.30319

C:\Windows\Microsoft.NET\Framework\v4.0.30319>
```

• **Bước 12:**

Đi đến thư mục của mã nguồn, tiếp tục di chuyển đến *bin > Debug* và sao chép đường dẫn.





3. Cài đặt Windows service

Chạy dòng lệnh theo cấu trúc sau:

`InstallUtil.exe + Your_copied_path+\your_service_name+.exe`

Ví dụ: `InstallUtil.exe C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.exe`

```
Administrator: Command Prompt
C:\Windows\Microsoft.NET\Framework\v4.0.30319>InstallUtil.exe C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.exe
Microsoft (R) .NET Framework Installation utility Version 4.8.4084.0
Copyright (C) Microsoft Corporation. All rights reserved.

Running a transacted installation.

Beginning the Install phase of the installation.
See the contents of the log file for the C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.exe assembly's progress.
The file is located at C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.InstallLog.
Installing assembly 'C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.exe'.
Affected parameters are:
logtoconsole =
logfile = C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.InstallLog
assemblypath = C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.exe
Installing service Service1...
Service Service1 has been successfully installed.
Creating EventLog source Service1 in log Application...

The Install phase completed successfully, and the Commit phase is beginning.
See the contents of the log file for the C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.exe assembly's progress.
The file is located at C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.InstallLog.
Committing assembly 'C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.exe'.
Affected parameters are:
logtoconsole =
logfile = C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.InstallLog
assemblypath = C:\Users\inseclab\source\repos\UITService1\UITService1\bin\Debug\UITService1.exe

The Commit phase completed successfully.

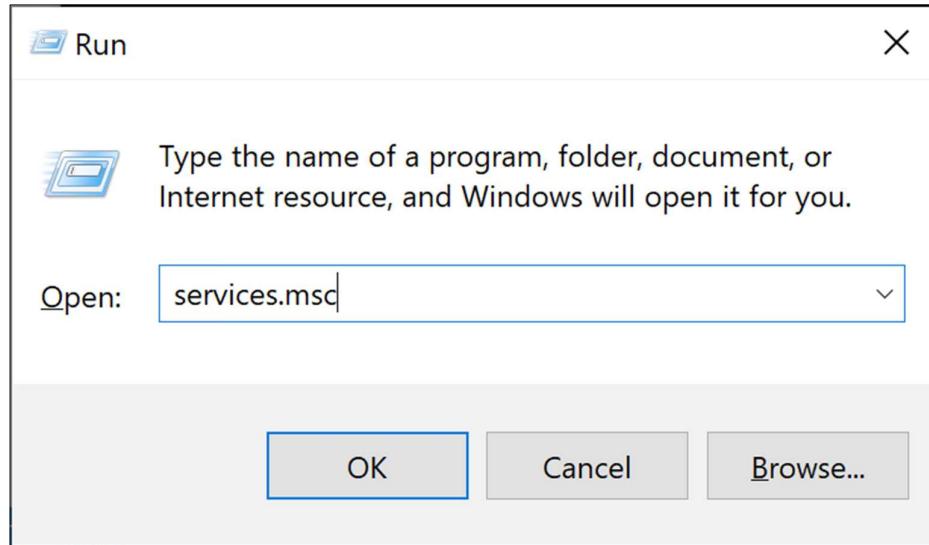
The transacted install has completed.

C:\Windows\Microsoft.NET\Framework\v4.0.30319>
```

4. Kiểm tra trạng thái của Windows Service

Mở services theo các bước sau:

1. Nhấn phím Windows + R
2. Nhập services.msc
3. Tìm service



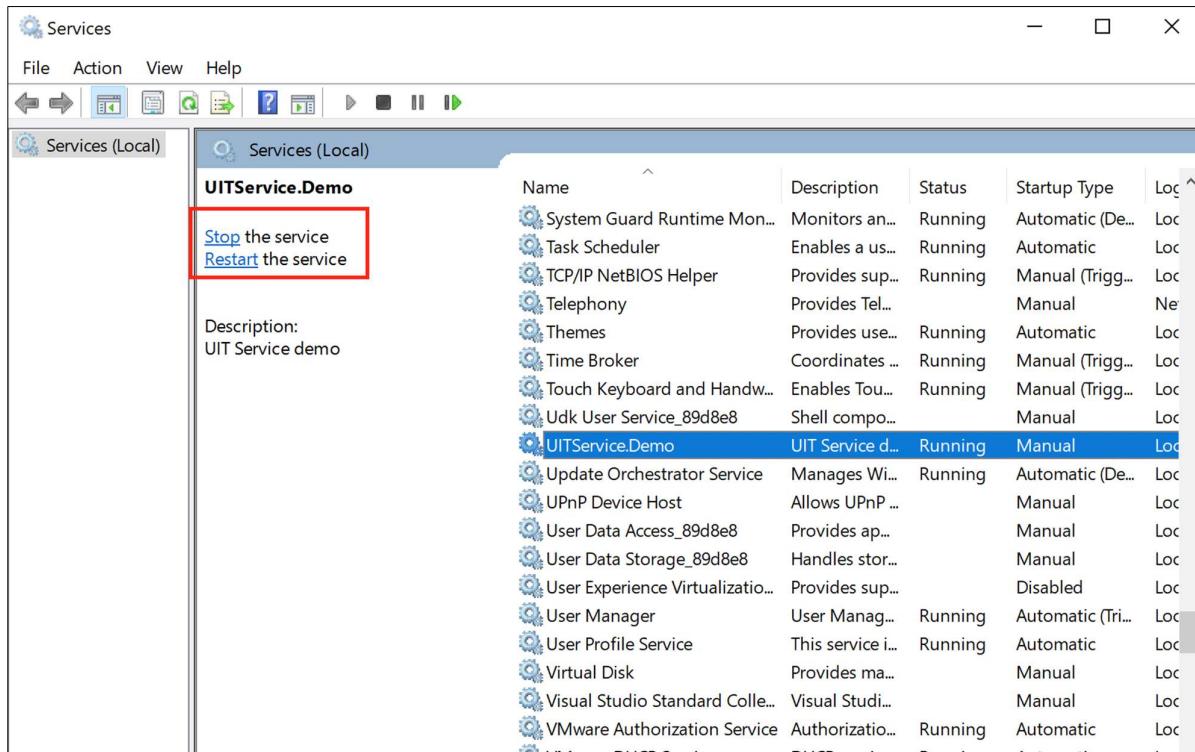
Services

File Action View Help

Services (Local)

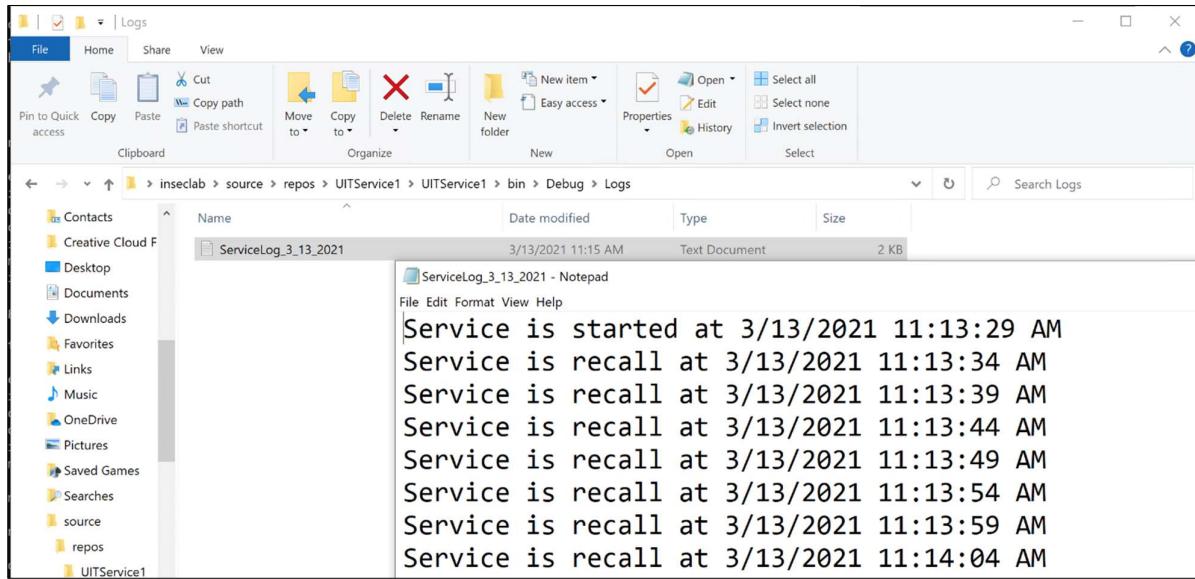
Name	Description	Status	Startup Type	Loc
System Guard Runtime Mon...	Monitors an...	Running	Automatic (De...	Loc
Task Scheduler	Enables a us...	Running	Automatic	Loc
TCP/IP NetBIOS Helper	Provides sup...	Running	Manual (Trigg...	Loc
Telephony	Provides Tel...	Manual	Loc	
Themes	Provides use...	Running	Automatic	Loc
Time Broker	Coordinates ...	Running	Manual (Trigg...	Loc
Touch Keyboard and Handw...	Enables Tou...	Running	Manual (Trigg...	Loc
Udk User Service_89d8e8	Shell compo...	Manual	Loc	
UITService.Demo	UIT Service d...	Manual	Loc	
Update Orchestrator Service	Manages Wi...	Running	Automatic (De...	Loc
UPnP Device Host	Allows UPnP ...	Manual	Loc	
User Data Access_89d8e8	Provides ap...	Manual	Loc	
User Data Storage_89d8e8	Handles stor...	Manual	Loc	
User Experience Virtualizatio...	Provides sup...	Disabled	Loc	
User Manager	User Manag...	Running	Automatic (Tri...	Loc
User Profile Service	This service i...	Running	Automatic	Loc
Virtual Disk	Provides ma...	Manual	Loc	
Visual Studio Standard Colle...	Visual Studi...	Manual	Loc	

Winndows service lúc này đã chạy



5. Kiểm tra output của Windows service

Lúc này tập tin đã được tạo theo đường dẫn ta quy định.



Bài thực hành 1: Sinh viên trình bày cách gỡ cài đặt Window service trên.

Bài thực hành 2: Viết một Windows service có nhiệm vụ kiểm tra một “process” ở trạng thái hoạt động run/stop hay không và run/stop “process” theo một lịch biểu.

Bài thực hành 3: Viết một Windows service có nhiệm vụ kiểm tra kết nối internet của máy hiện tại (HTTP) và tạo reverse shell đơn giản.

D. YÊU CẦU & ĐÁNH GIÁ

- Sinh viên tìm hiểu và thực hành theo hướng dẫn, thực hiện **theo nhóm đã đăng ký**.
- Nộp báo cáo kết quả gồm **Code, File được export** và chi tiết những việc (**Report**) mà nhóm đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).

Báo cáo:

- File **.PDF**. Tập trung vào nội dung, không mô tả lý thuyết.
- Đặt tên theo định dạng: **[Mã lớp]-LabX_MSSV1-MSSV2-MSSV3**.
Ví dụ: *[NT230.N2X.ATCL]-Lab1_2052xxxx-2052yyyy*.
- Nếu báo cáo có nhiều file, nén tất cả file vào file **.ZIP** với cùng tên file báo cáo.
- Nộp file báo cáo trên theo thời gian đã thống nhất tại courses.uit.edu.vn.

Bài sao chép, trẽ, ... sẽ được xử lý tùy mức độ vi phạm.

Đặt sai tên file cáo cáo không chấm bài.

HẾT

Chúc các bạn hoàn thành tốt!