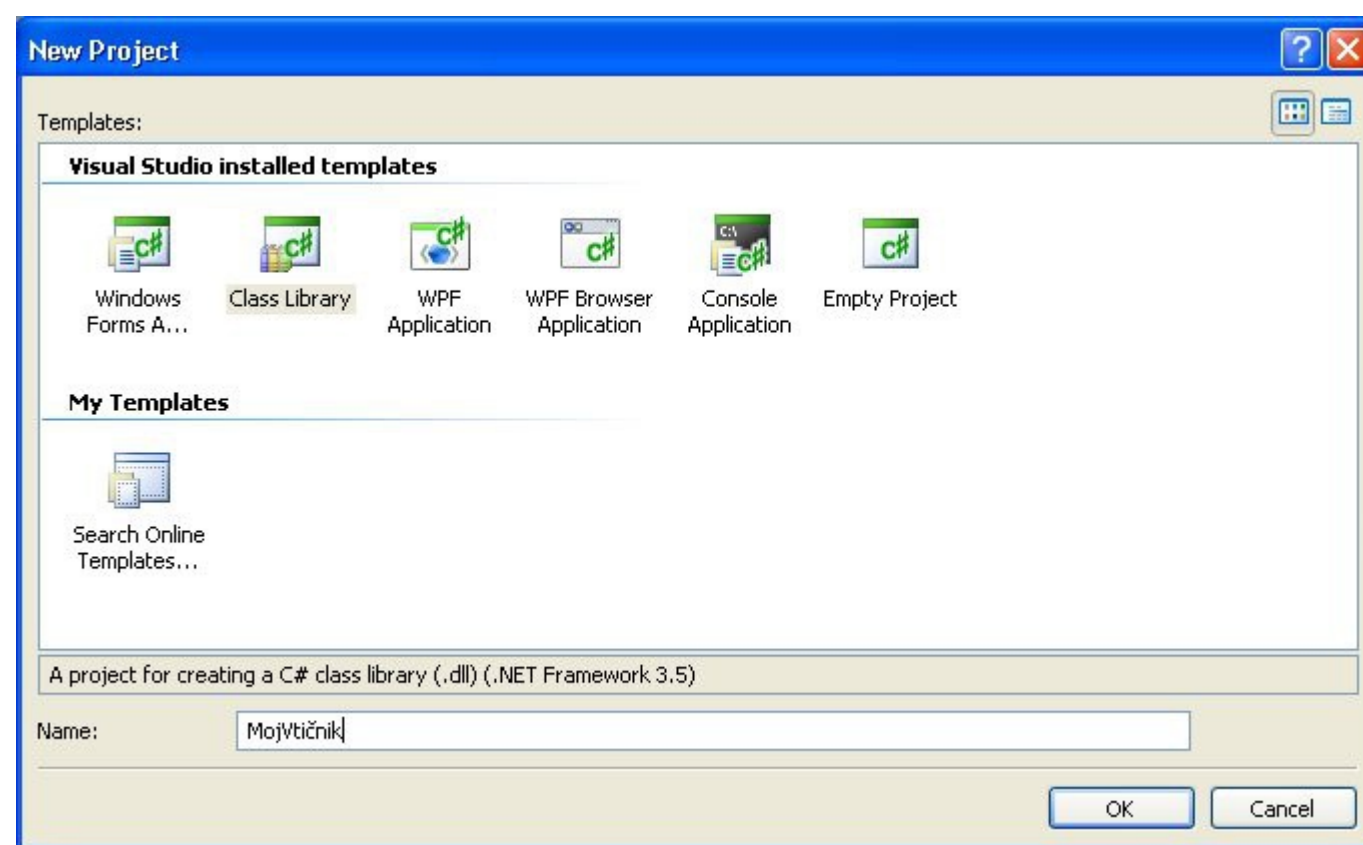
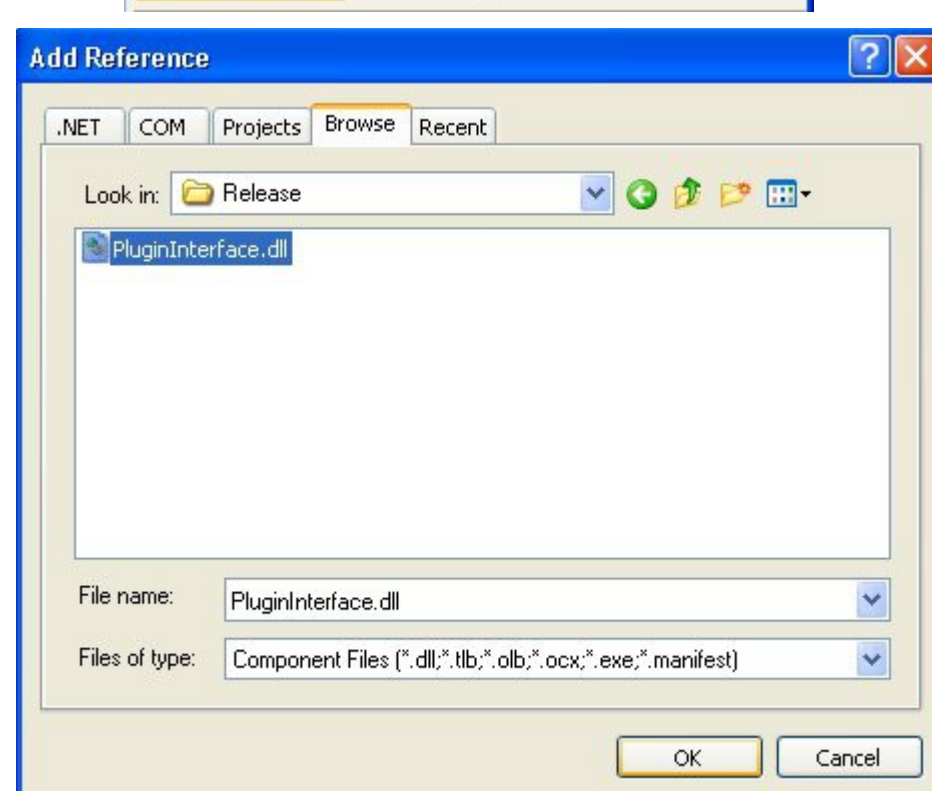
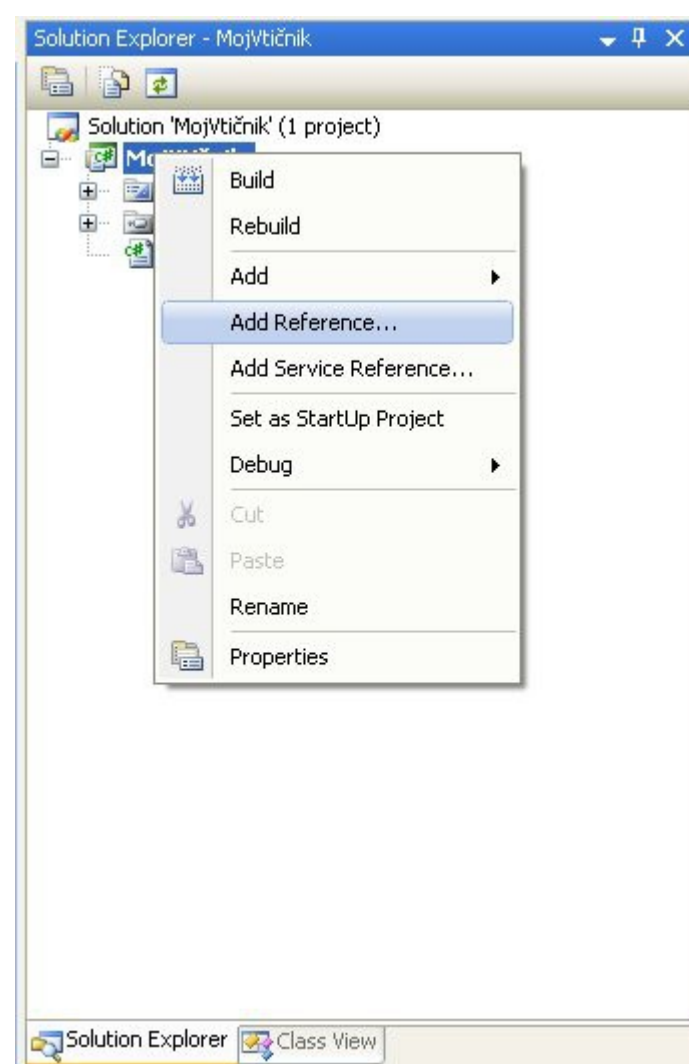


## Navodila za pisanje web epg vtičnikov:

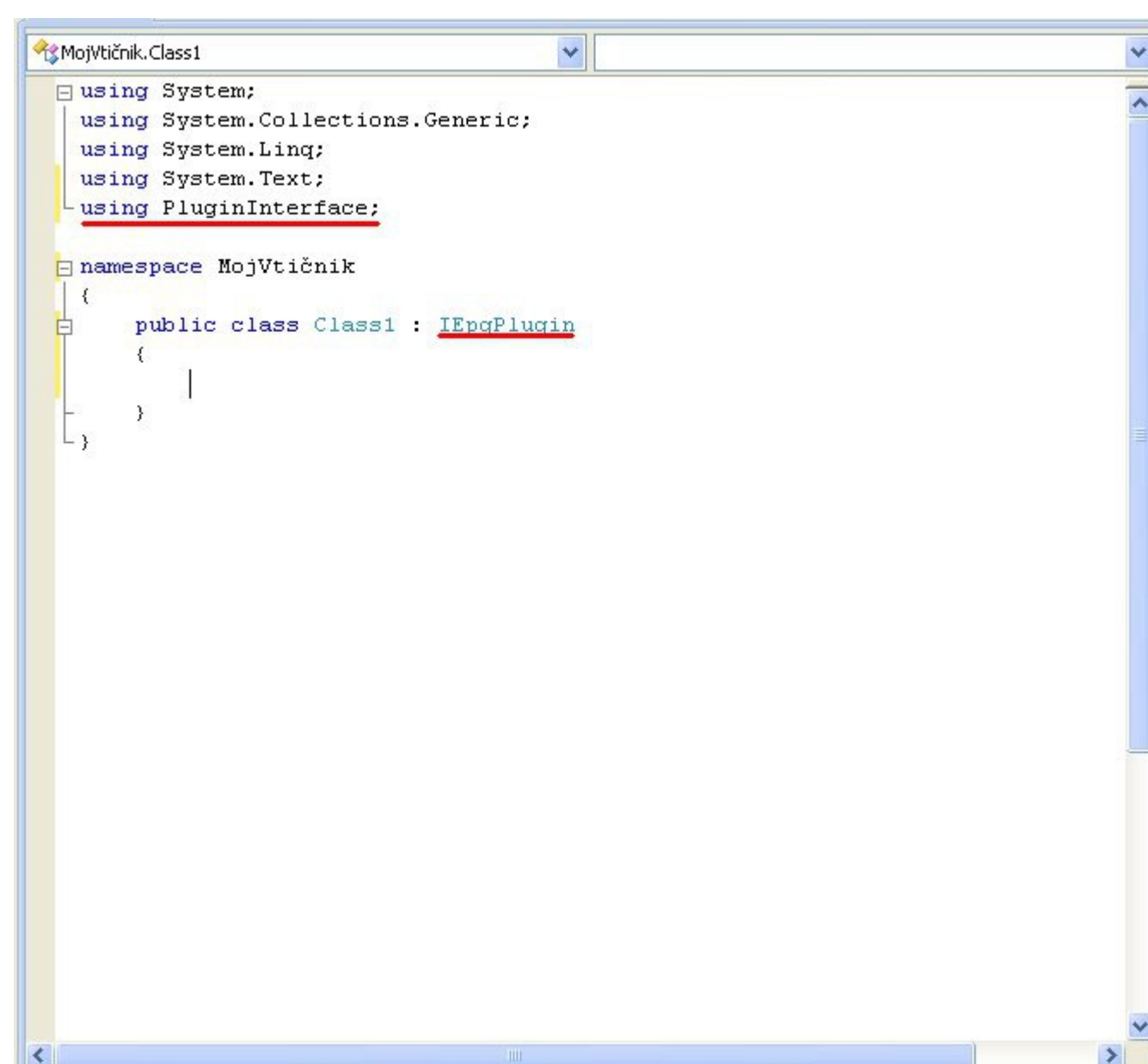
1. Ustvarimo nov projekt in izberemo Class Library:



2. V "Solution Explorer" oknu dodamo "PluginInterface.dll". PluginInterface.dll lahko najdete na strani [uross-digital-tools.appspot.com](http://uross-digital-tools.appspot.com) ( Source code & API ). Prenesite celotno izvorno kodo za MulticastTV.Vlc. Dll in primeri se nahajajo v mapi "plugins".



3. Ustvarimo razred, ki mora podedovati IEpgPlugin interface.



4. Ta razred mora implementirati vse lastnosti in metode :

```
public interface IEpgPlugin
{
    /// <summary>
    /// Avtor vtičnika
    /// </summary>
    string Author
    {
        get;
    }

    /// <summary>
    /// Verzija vtičnika
    /// </summary>
    string Version
    {
        get;
    }

    /// <summary>
    /// Multicastv verzija s katero je komptibilen
    /// </summary>
    string MulticastvVersion
    {
        get;
    }

    /// <summary>
    /// Komentar
    /// </summary>
    string Comment
    {
        get;
    }

    /// <summary>
    /// Koda jezika na katerega je vtičnik vezan: za Slovenski jrezik je koda "sl-si"
    /// "" za vse jezike.
    /// </summary>
    string LockedToUserInterfaceLangue
    {
        get;
    }

    /// <summary>
    /// Tip vtičnika "online" ali "offline"
    /// Online bere direktno iz interneta. Trenutno je možen le "online".
    /// Podatkov ni potrebno shranjevati na disk, ker to že stori program sam.
    /// </summary>
    string PluginType
    {
        get;
    }

    /// <summary>
    /// Ime vtičnika
    /// </summary>
    string Name
    {
        get;
    }

    /// <summary>
    /// Url naslov strani. Npr: "http://www.siol.net"
    /// </summary>
    string HostName
    {
        get;
    }

    /// <summary>
    /// POST ali GET http metoda.
    /// </summary>
    string RequestMethod
    {
        get;
    }
}
```

```

// <summary>
// Prva zahteva - brez naslova gostitelja.
// Npr. "http://www.siol.net/tv-spored.aspx" mora biti le "tv-spored.aspx".
// Potrebno izpolniti le, če je InitialRequest true.
// Prva zahteva je uporabna, če rabimo še kakšno dodatno variabla, ki je ne vemo vnaprej.
// </summary>
string InitialRequest
{
    get;
}

// <summary>
// Ali je prva zahteva sploh potrebna?
// </summary>
bool InitialRequestNeeded
{
    get;
    set;
}

// <summary>
// POST ali GET metoda za prvo zahtevo.
// </summary>
string InitialRequestMethod
{
    get;
}

// <summary>
// Ta funkcija je lahko uporabna za nalaganje dodatnih nastavitev iz datoteke .
// </summary>
void Initialize(string Path);

// <summary>
// Počisti.
// </summary>
void Dispose();

// <summary>
// array[0]=http zahteva (brez imena gostitelja (hostname),
// array[1]=referer
// array[2]=POST string – prazno, če je motodas GET

// ChannelId je identifikacija programa na internetu.
// Seznam z identifikacijami programov se mora nahajati v mapi parse!
// Ime seznama mora biti enak imenu vtičnika!!!
// Glej siol.txt za primer.
// <summary>
// <param name="ChannelId">Channel web identification</param>
string[] GetTodayRequest(string ChannelId);

// <summary>
// Še ni implementirano.
// </summary>
string[] GetTodaySummaryRequest();

// <summary>
// Ta funkcija je klicana, le če je InitialRequest true.
// Prva zahteva je uporabna, če rabimo, še kakšno dodatno variabla, ki je ne vemo vnaprej.
// </summary>
void ParseInitPage(string Html);

// <summary>
// Ta funkcija ekstrahira osnovne epg podatke za tekoči dan.
// Html je "string", kateri vsebuje celotno web stran (rezultat GetTodayRequest funkcije).
// Časi oddaj morajo biti od 0:00 in do največ 11:59 naslednjega dne.
// Format:
// array[0] = čas začetka oddaje; format: "hh:mm - " !!!! Pazi na pomišljaj in oba presledka !!!
// array[1] = naslov oddaje;
// array[2] = žanr;
// array[3] = http povezava do strani , ki vsebuje podrobnosti o oddaji.;
// <summary>
// <param name="Html">Html page</param>
List<string[]> ParseToday(string Html);

// <summary>
// Še ni implementirano.
// </summary>
// <param name="Html">Html</param>
List<string[]> ParseTodaySummary(string Html);

// <summary>
// Ta funkcija ekstrahira podrobnosti o oddaji, kot so npr. igralci, vsebina, režiser ipd.
// Format:
// array[0] = Naslov;
// array[1] = Žanr;
// array[2] = Opis;
// array[3] = Ime kanala;
// array[4] = Čas začetka (ni važen format);
// array[5] = Poljubno 1 (npr. igralec1, igralec2;);
// array[6] = Opis za polje Poljubno 1(npr. Igrajo;);
// array[7] = Poljubno 2 (npr: ime režiserja);
// array[8] = Opis za polje Poljubno 2 ( npr.: Režiser);
// array[9] = Link do slike;
// </summary>
// <param name="Html">Html</param>
string[] ParseDetails(string Html);

IPluginHost Host { get; set; }
}

public interface IPluginHost
{
    void Feedback(string Feedback, IEpgPlugin Plugin);
}

```

Primere lahko najdete tukaj : <http://sites.google.com/site/multicasttv/downloads> (Web epg plugins).

5. Ustvarimo seznam z id-ji. Ime seznama mora biti **identično imenu epk vtičnika**.

Npr. t2.txt:

```

POP TV=4
Kanal A=5
TV3=7
SLO 1=2
SLO1=2

```

To datoteko prekopiramo v Multicasttv namestitveno mapo in sicer v podmapo **parse** .

6. Ko končamo, prekopiramo MojVtičnik.dll v podmapo **webepgplugins**. Datoteke PluginInterface.dll ne kopiramo !

