

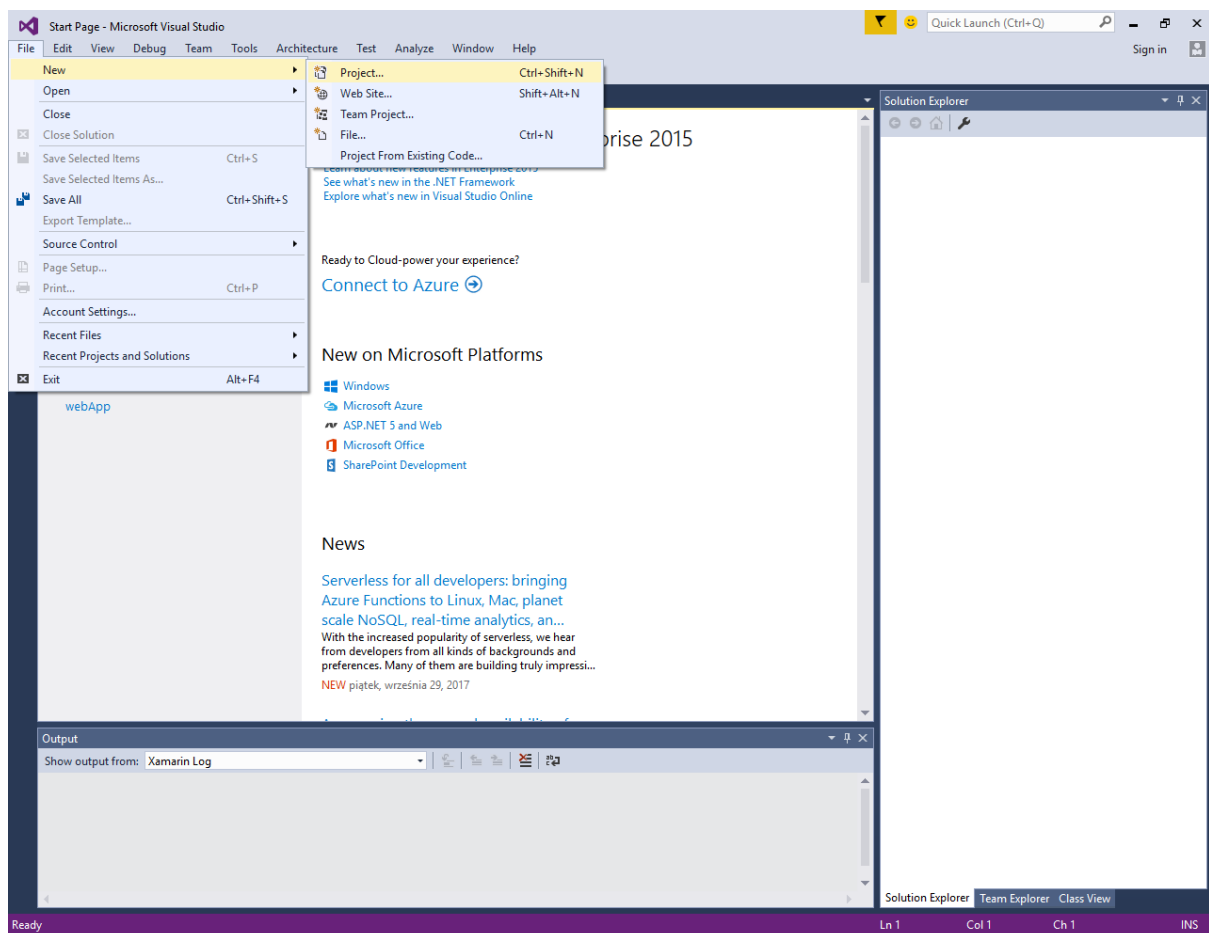


ASP

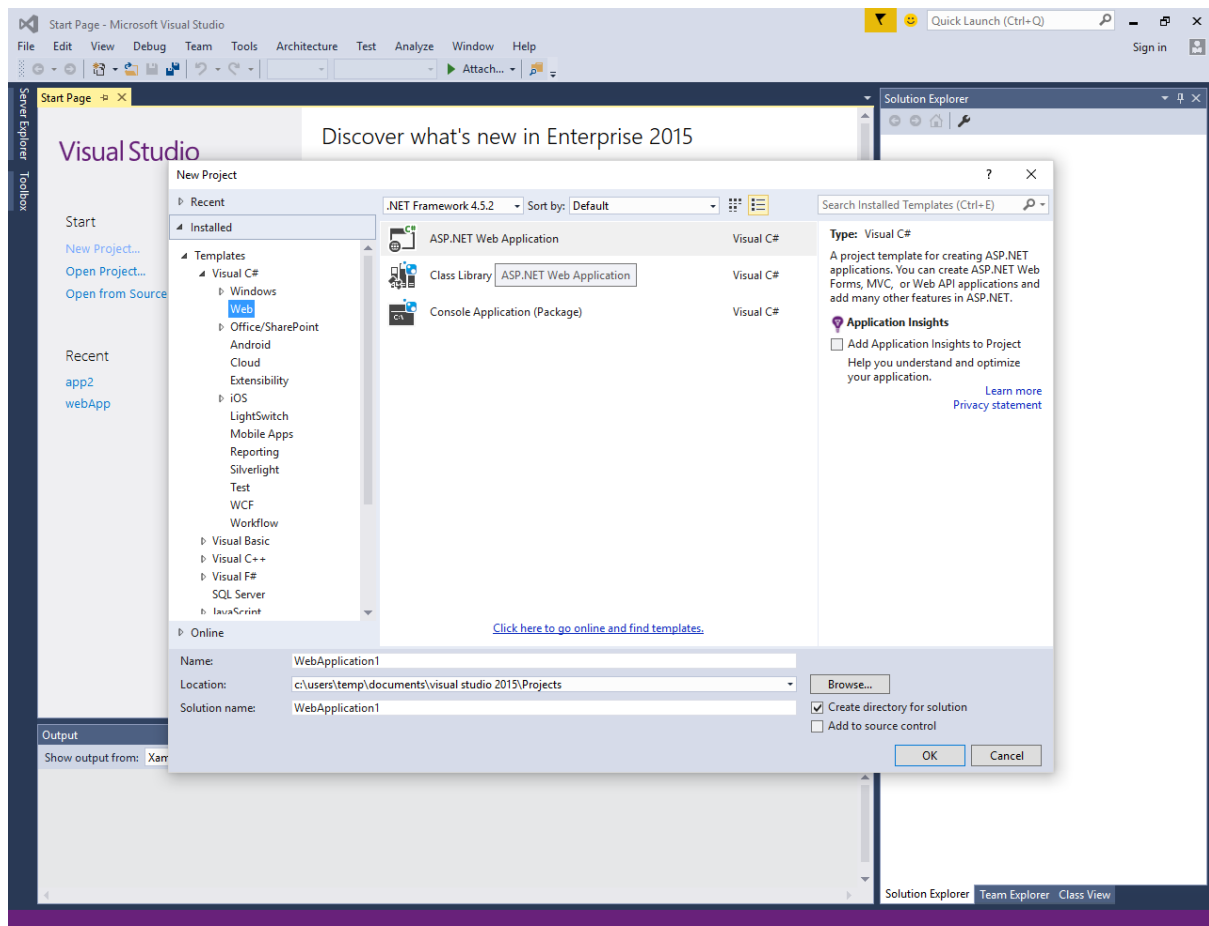
ASP .Net

I. Przygotowanie i wstęp:

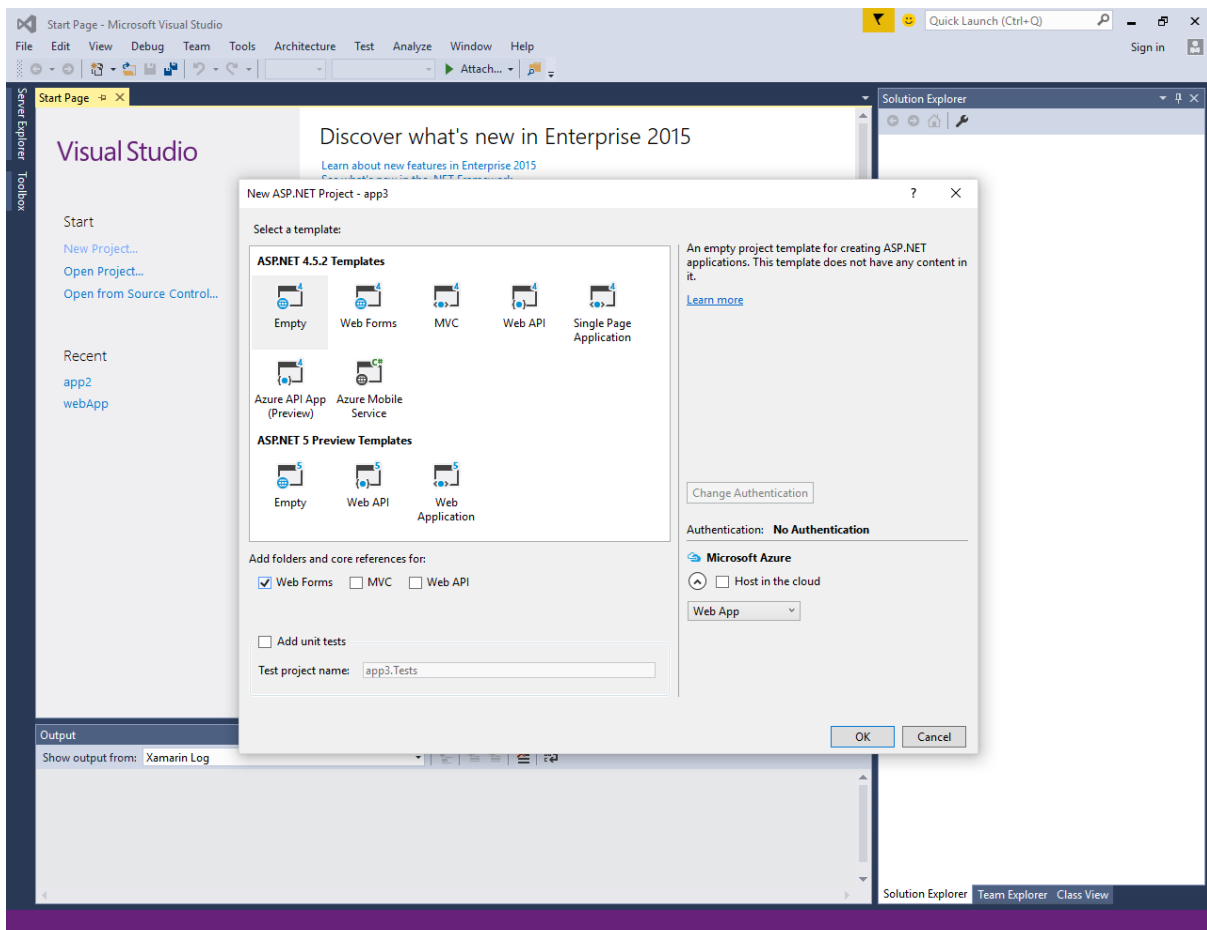
1. Tworzymy nowy projekt:



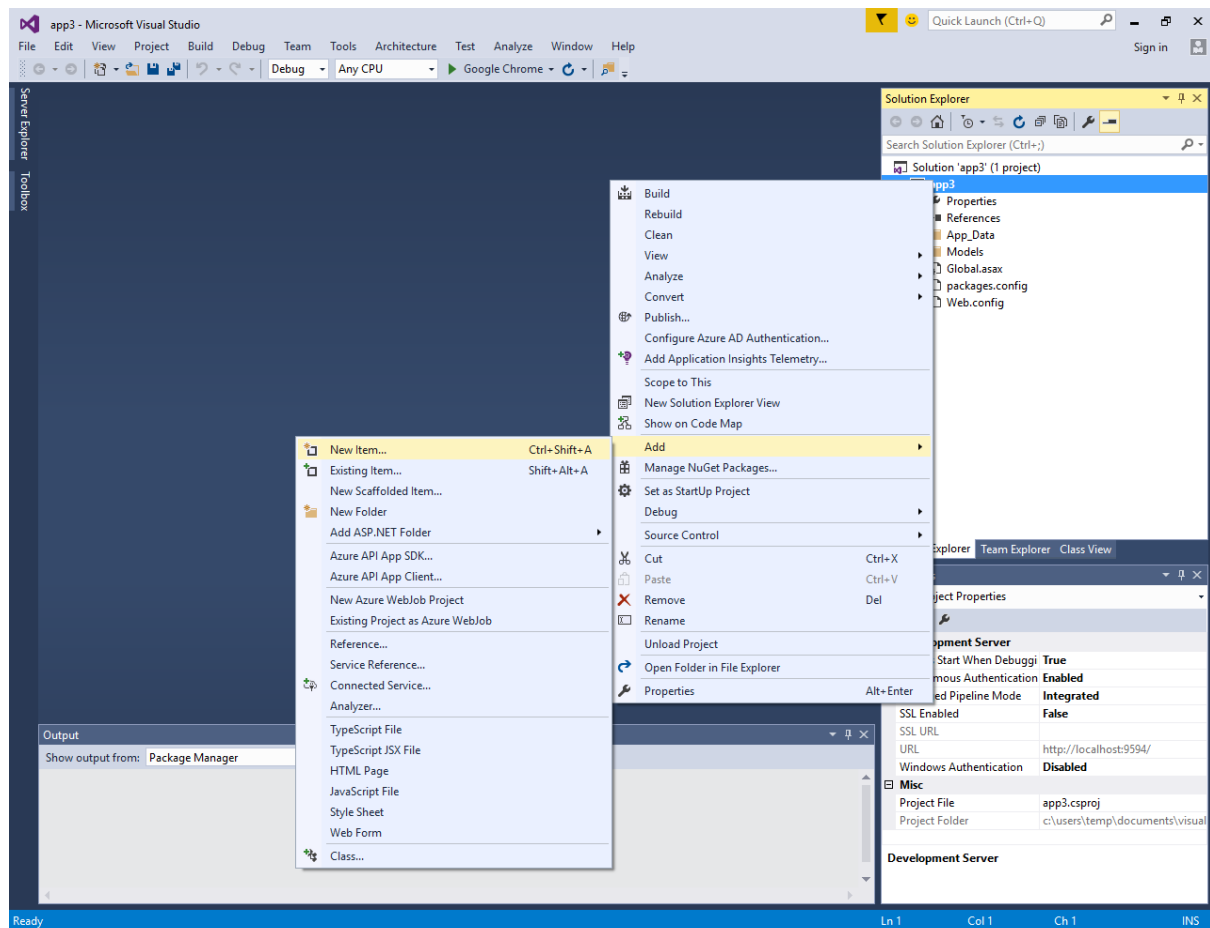
2. Wybieramy Asp Web Application



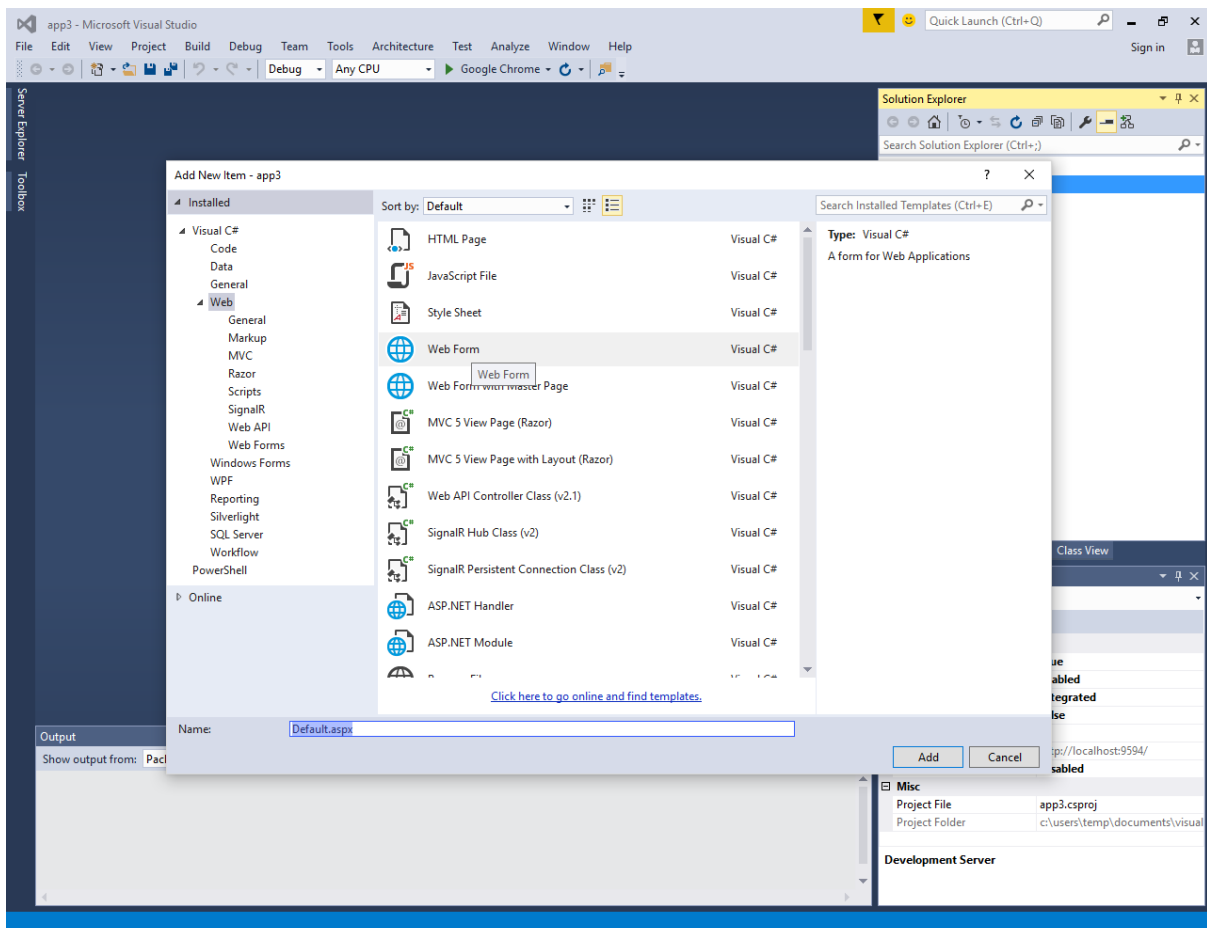
3. Wybieramy Empty i zaznaczamy web for ptaszkiem.



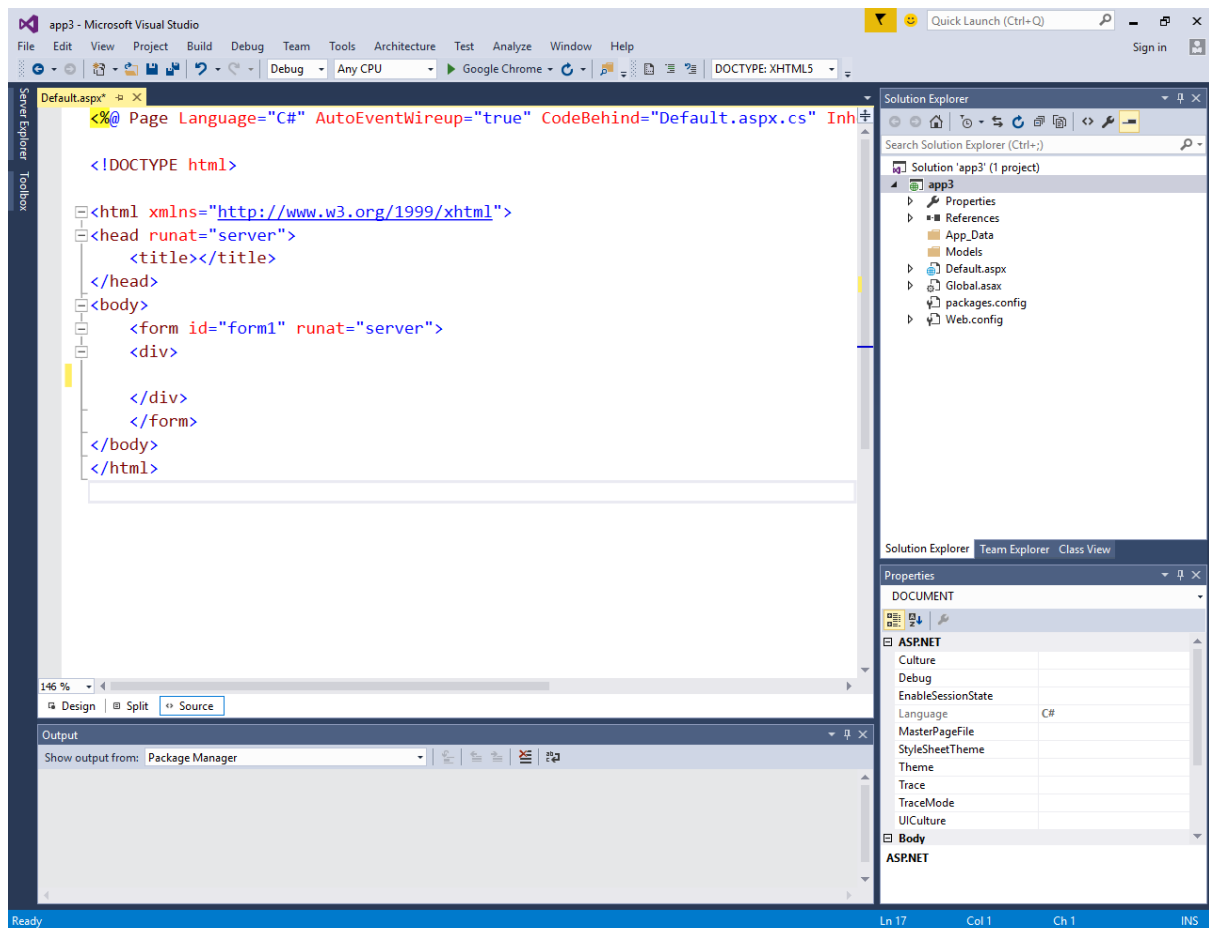
4. Dodajemy nowy element:



5. dodajemy nową formę i nazywamy ją Default.aspx



6. Powinno to wyglądać następująco:

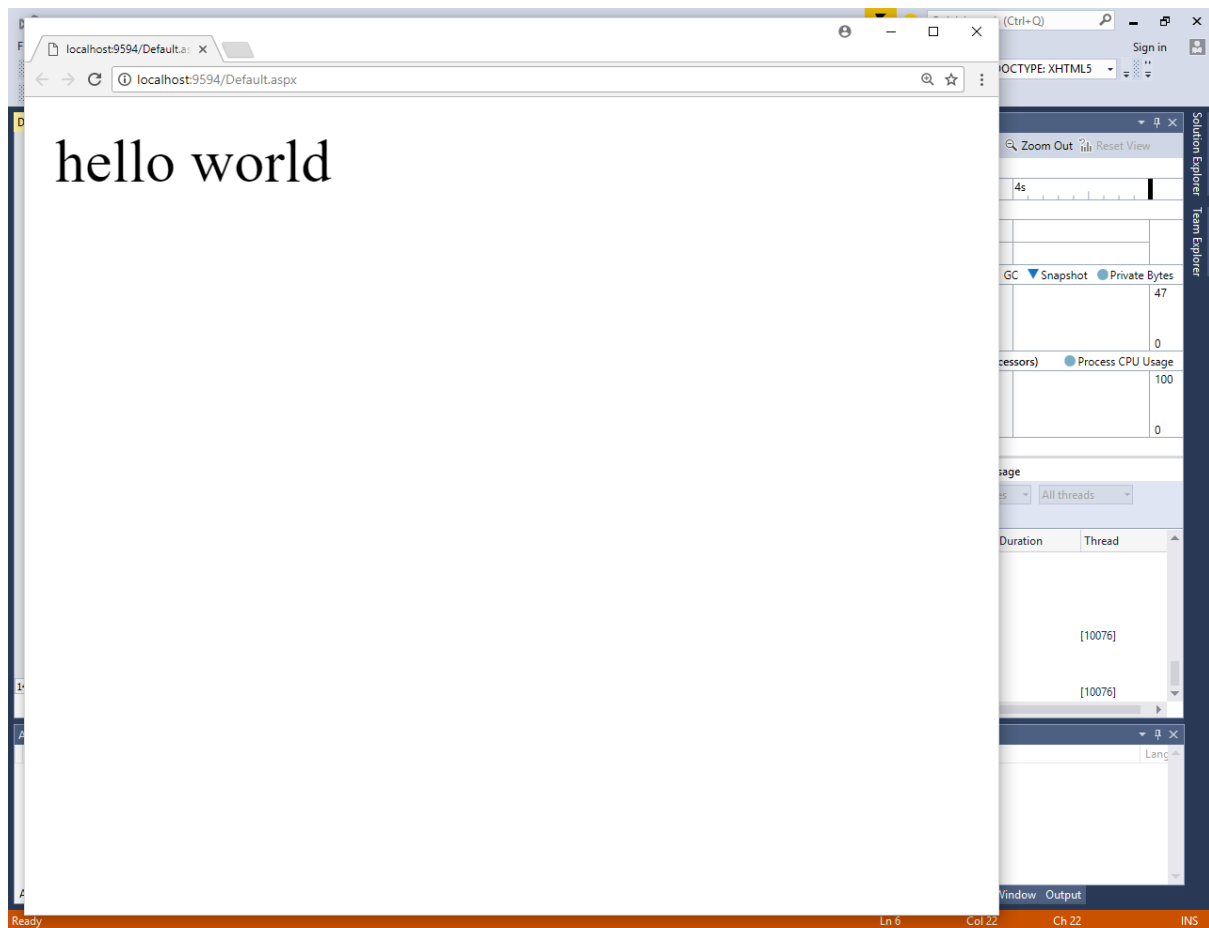


7. Nie tworzymy pliku master wybieramy prostszą opcję z samym plikiem standardowym

8. do div wstawiamy następujący kod:

```
<div>  
    hello world  
</div>
```

9. Klikamy runn i naszym oczu powinno się ukazać coś takeigo:

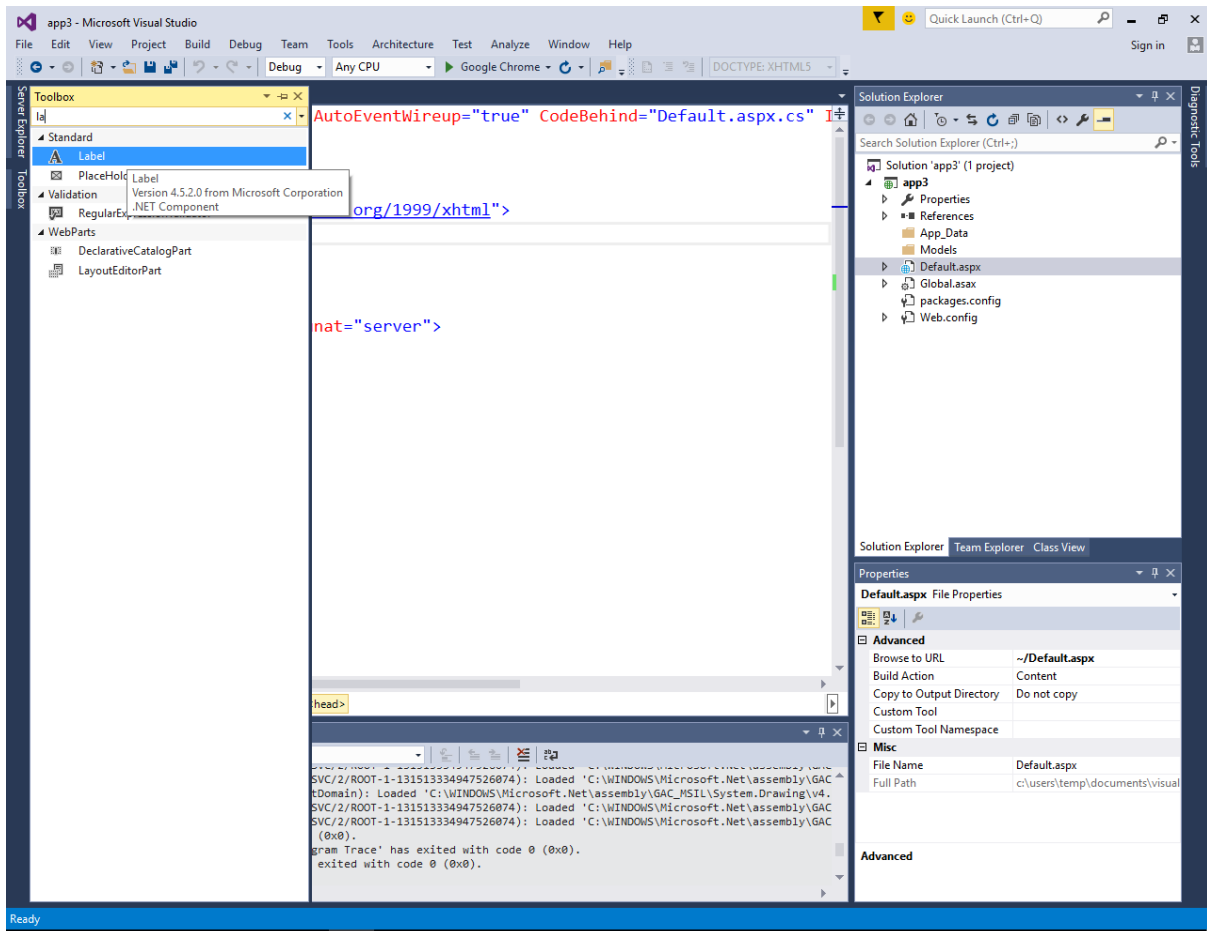


II. Prosty program (zegarek)

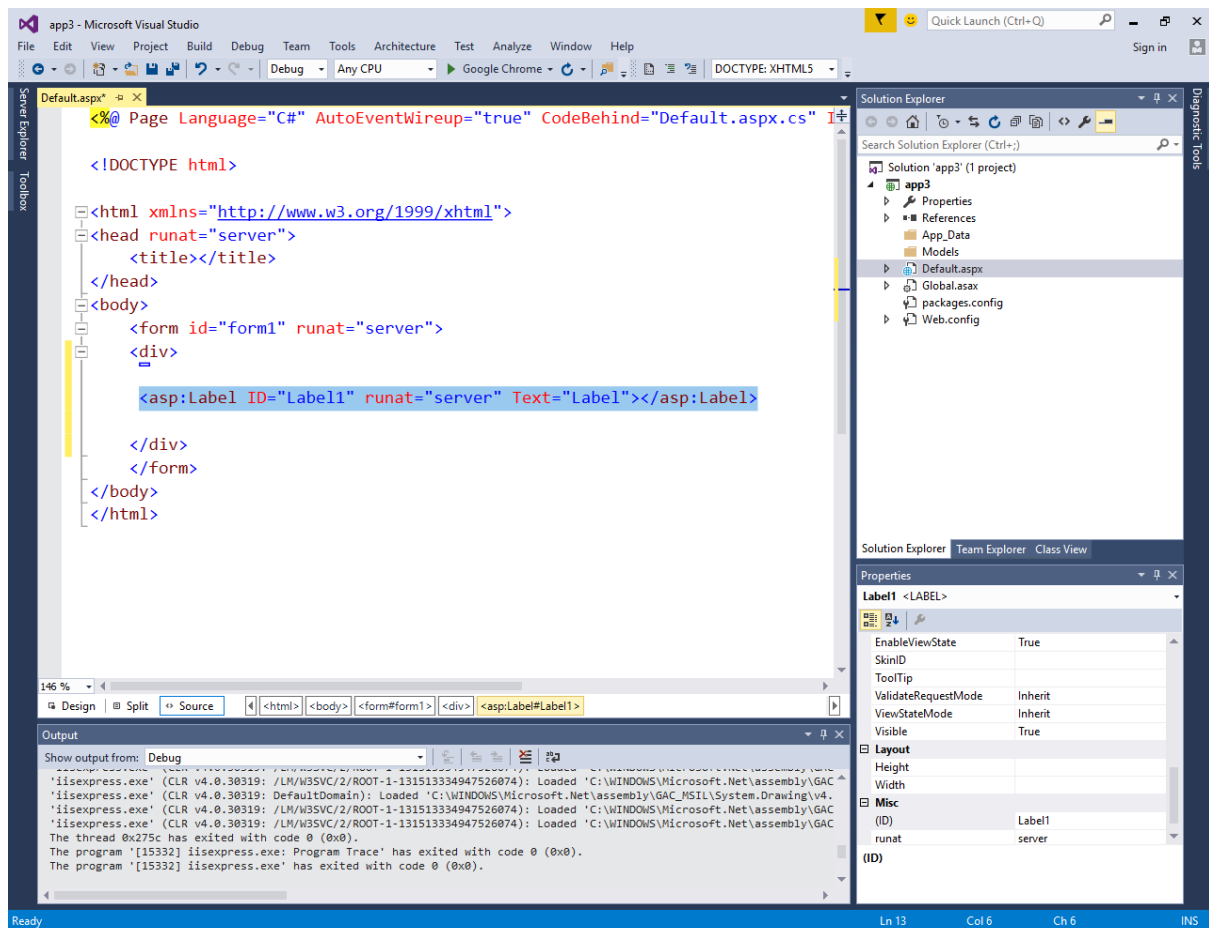
1. Tworzymy w pliku default Default.aspx następujący element

```
<asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
```

1.1 Wybieramy go z tolllboxa



1.3 Efekt:



2. W ten sam sposób dodajemy kolejne dwa elementy

2.1 Element timera przeładowujący przeładowujący dane na stronie:

```
<asp:Timer ID="Timer1" runat="server" Interval="100"
OnTick="Timer1_Tick"></asp:Timer>
```

ID: nazwa obiektu timera

Interval: Co ile timer się wykona wartość podana w milisekundach.

OnTick: nazwa metody wykonywanej za każdym odplaceniem Timera

INFO: dokładne info z strony msdn: [Link](#)

Performs asynchronous or synchronous Web page postbacks at a defined interval.

Namespace: *System.Web.UI*

Assembly: *System.Web.Extensions (in System.Web.Extensions.dll)*

Przykłady użycie timera: [Link](#)

2.2 Następnie do element zarządzający skryptami:

```
<asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>
```

INFO: dokładne info z strony msdn: [Link](#)

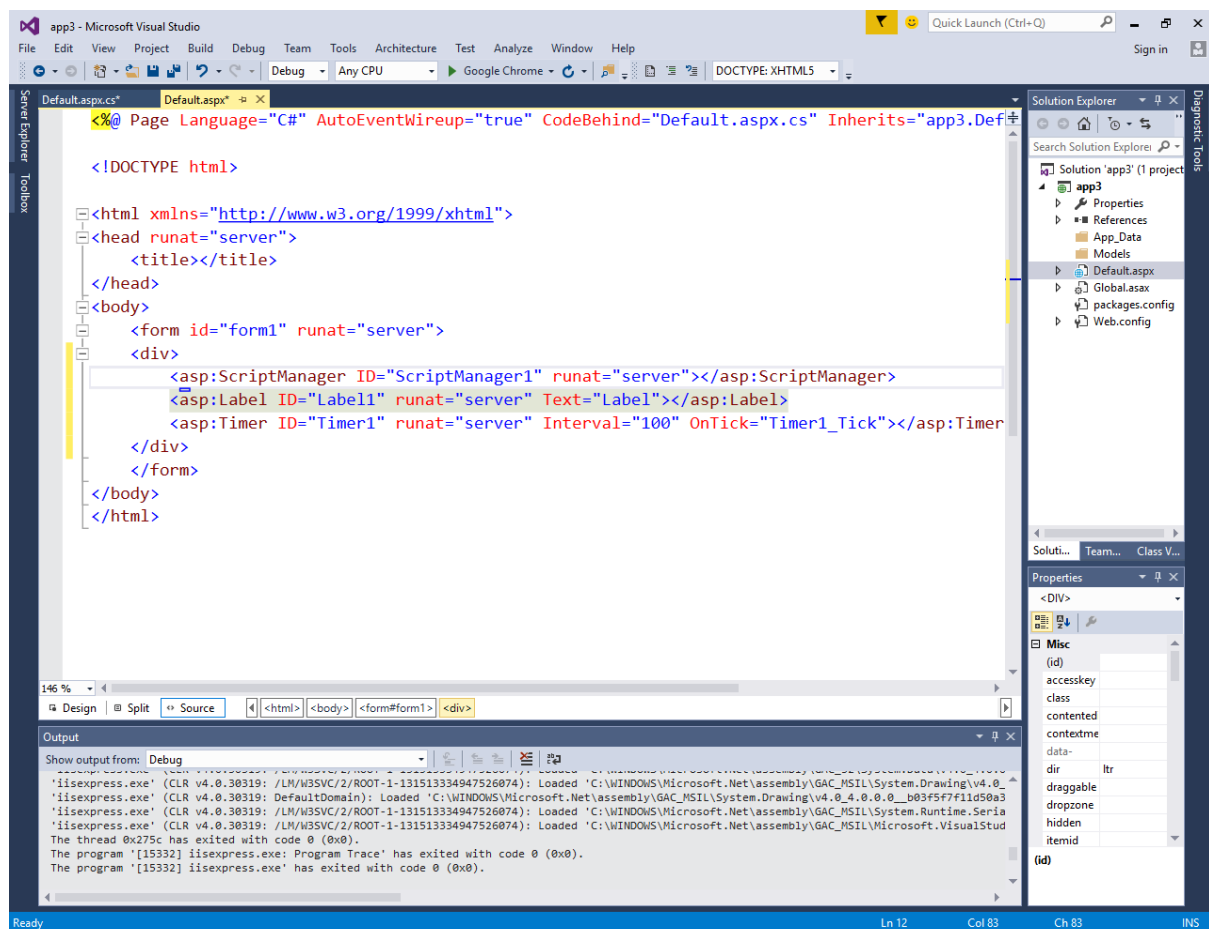
Zarządza ASP.NET Ajax, biblioteki skryptów i plików skryptów, częściowe renderowanie strony i generowania klasy serwera proxy klienta dla aplikacji sieci Web i usług.

UWAGA!!!: obiekt musi się znajdować nad wszystkimi skryptami wykonywalnymi

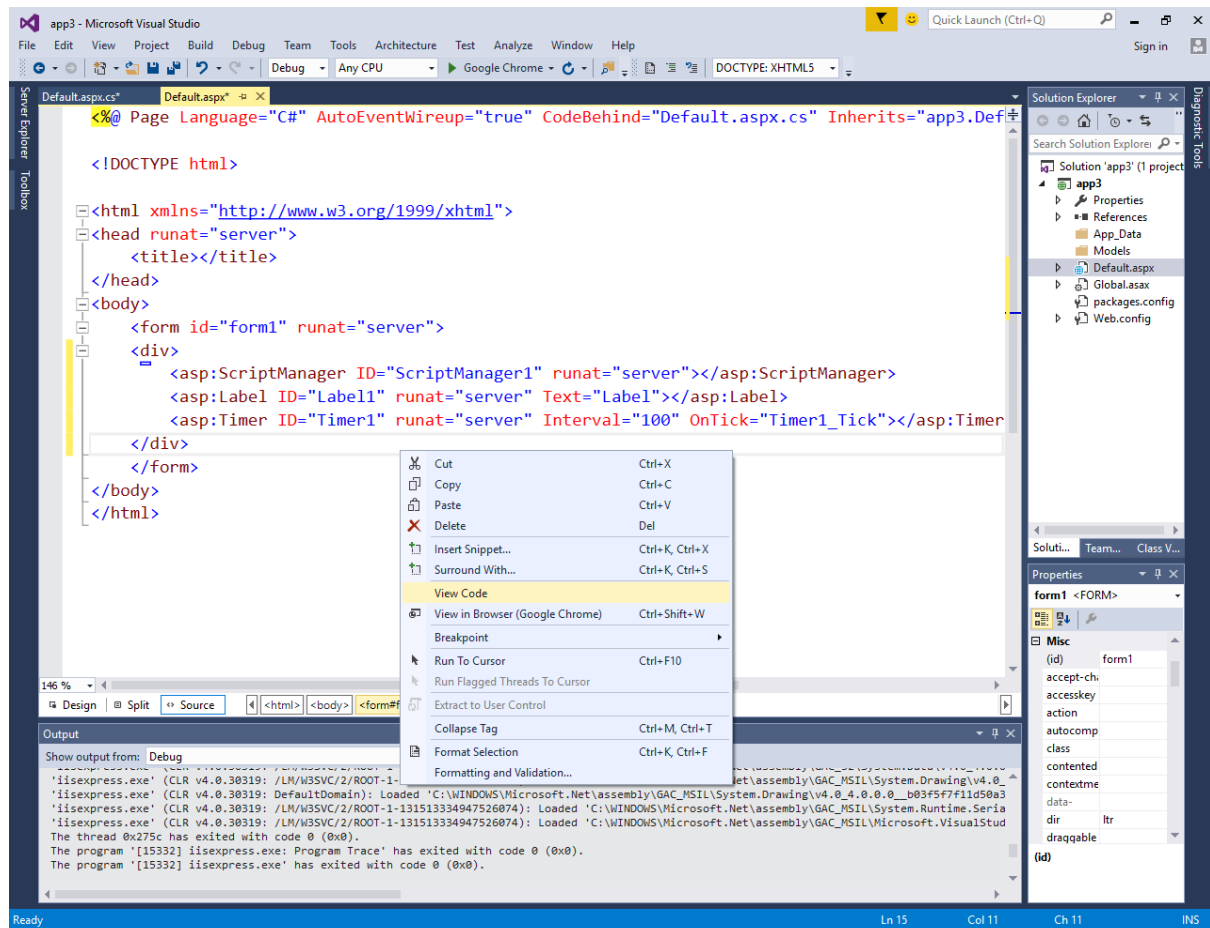
Przestrzeń nazw: *System.Web.UI*

Zestaw: *System.Web.Extensions* (w *System.Web.Extensions.dll*)

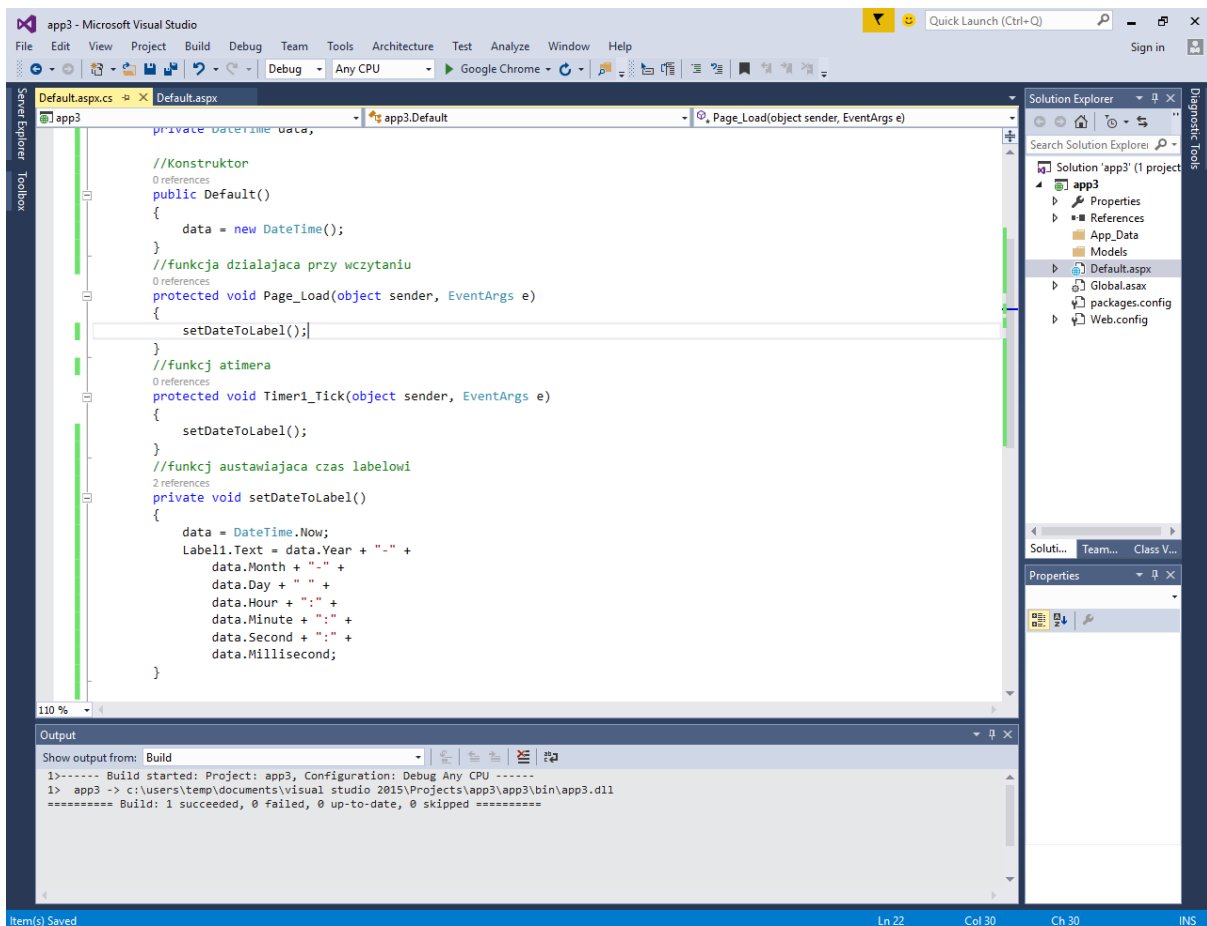
2.3 Efekt



3. Tworzymy kod obsługujący zdarzenia: (edytujemy Default.aspx a konkretnie jego kod C#)



3.1 tworzymy konstruktor i wypełniamy metody timera oraz wczytywania strony logiką:



Kod: (Uwaga nie wklejać bezmyślenie przanalizwoać komentarze)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
```

```
namespace app3 //uwaga to się musi zgadzać z wasza nazwa projektu
{
```

```
    public partial class Default : System.Web.UI.Page
    {
        private DateTime data;

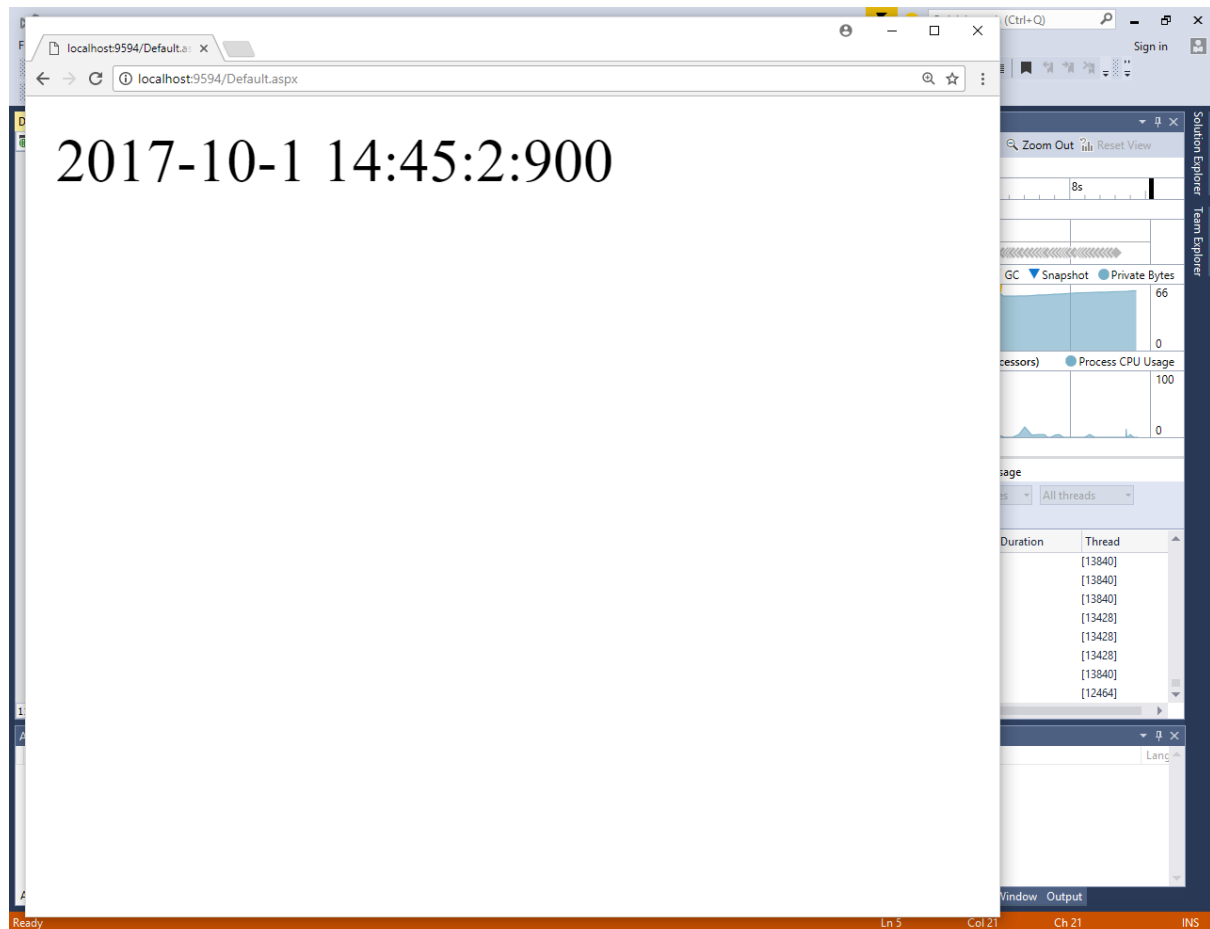
        //Konstruktor
        public Default()
        {
            data = new DateTime();
        }
        //funkcja działająca przy wczytaniu
        protected void Page_Load(object sender, EventArgs e)
        {
            setDateToLabel();
        }
        //funkcj atimera
        protected void Timer1_Tick(object sender, EventArgs e)
        {
            setDateToLabel();
        }
    }
}
```

```

    }
    //funkcj austawiajaca czas labelowi
    private void setDateToLabel()
    {
        data = DateTime.Now;
        Label1.Text = data.Year + "-" +
        data.Month + "-" +
        data.Day + " " +
        data.Hour + ":" +
        data.Minute + ":" +
        data.Second + ":" +
        data.Millisecond;
    }
}
}
}

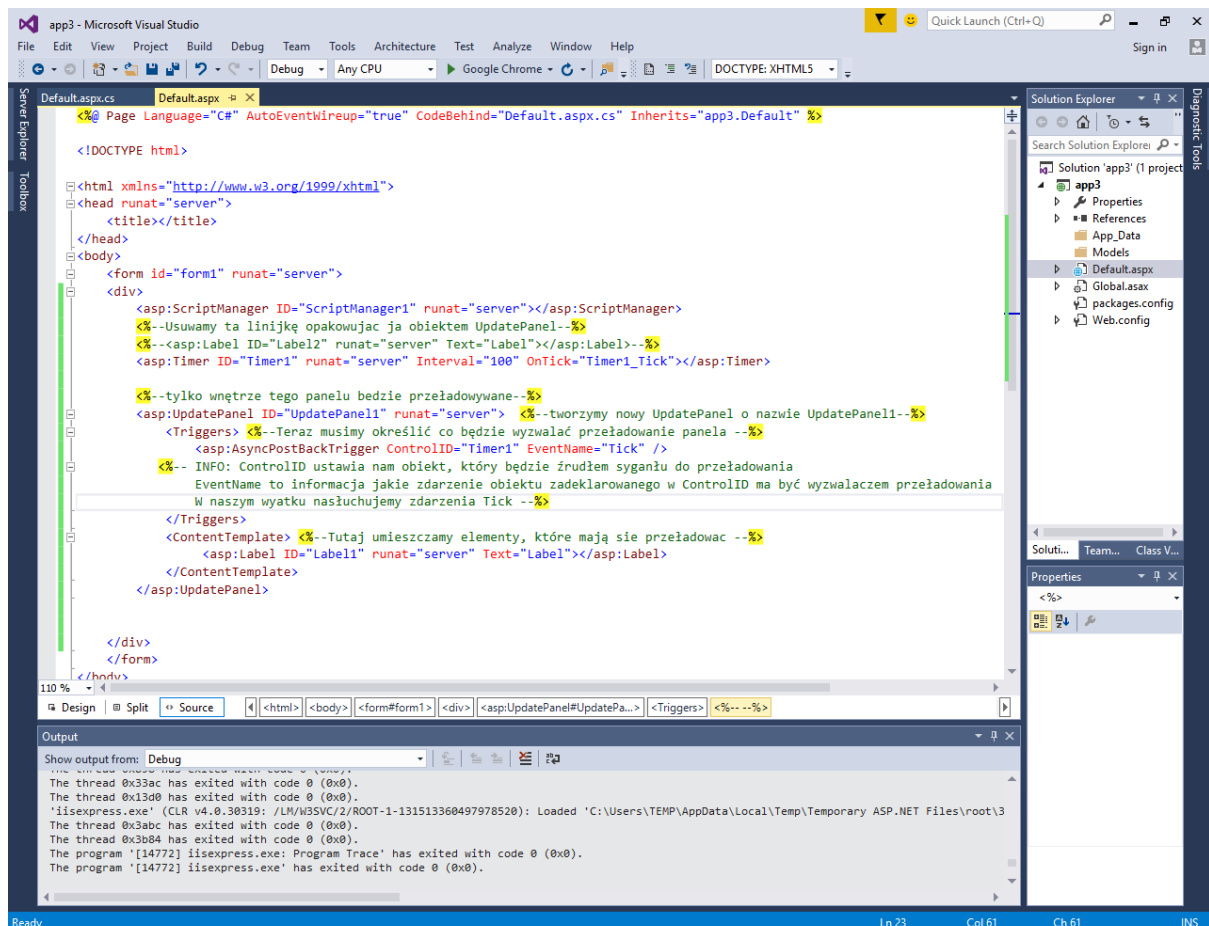
```

3.2 Efekt:



III. Prosty program (zegarek) - Użyjmy Ajaxa

1. Modyfikujemy naszą aplikację w następujący sposób



KOD: (Uwaga czytamy komentarze)

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits="app3.Default" %>
```

```
<!DOCTYPE html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>
            <!--Usuwamy ta linijkę opakowujac ja obiektem UpdatePanel-->
            <!--<asp:Label ID="Label2" runat="server" Text="Label"></asp:Label>-->
            <asp:Timer ID="Timer1" runat="server" Interval="100"
OnTick="Timer1_Tick"></asp:Timer>
```

```

            <!--tylko wewnątrz tego panelu będzie przeładowywane-->
            <asp:UpdatePanel ID="UpdatePanel1" runat="server"> <!--tworzymy nowy
UpdatePanel o nazwie UpdatePanel1-->
                <Triggers> <!--Teraz musimy określić co będzie wyzwać przeładowanie
panela -->
                    <asp:AsyncPostBackTrigger ControlID="Timer1" EventName="Tick" />
                    <!-- INFO: ControlID ustawia nam obiekt, który będzie źródłem sygnału
do przeładowania
```

```
        EventName to informacja jakie zdarzenie obiektu zadeklarowanego w
ControlID ma być wyzwalaczem przeładowania
        W naszym wyatku nasłuchujemy zdarzenia Tick --%>
        </Triggers>
        <ContentTemplate> <%--Tutaj umieszczamy elementy, które mają się
przeładowac --%>
        <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
        </ContentTemplate>
    </asp:UpdatePanel>

</div>
</form>
</body>
</html>
```

INFO: Tutaj są informacje o wszystkich elementach użytych w powyższym przykładzie:

Tick: [Link](#)

UpdatePanel: [Link](#)

Triggers: [Link](#)

ContentTemplate: [Link](#)

AsyncPostBackTrigger: [Link](#)

Zadanie Domowe:

Proszę wykonać stronę z zgodnie z powyższym tutorialiem.