

Tworzenie programów dla platformy .NET

Wprowadzenie

Marek Mittmann

Plan wykładu

- Założenia projektowe .NET
- Platforma .NET, języki i narzędzia
- Środowisko uruchomieniowe
- Języki .NET
- Rodzaje aplikacji

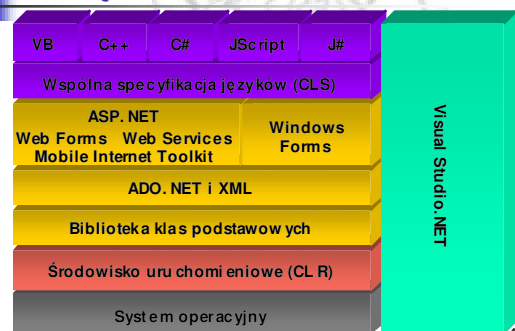
2

Założenia projektowe .NET

- Uprościć pisanie i wdrażanie oprogramowania
- Ujednolicić model programowania
- Dostarczyć wydajne i bezpieczne środowisko uruchomieniowe
- Wsparcie dla wielu języków programowania

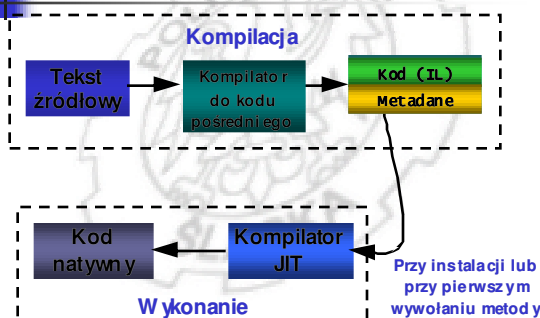
3

Platforma .NET, języki, narzędzia



4

Kompilacja i wykonanie



5

Uproszczenia w pisaniu programów

- Pełne wsparcie dla programowania obiektowego
 - Dziedziczenie wizualne
 - Dziedziczenie między różnymi językami
 - Wszystko jest obiektem
- Mechanizmy współpracy
 - z COM
 - z kodem Win32

6

Wydajne środowisko

- Automatyczne zarządzanie czasem życia obiektów
 - Wszystkie obiekty podlegają „odśmiecaniu”
- Obsługa wyjątków
- Bezpieczne typy
 - Bez: przepełnień buforów, niebezpiecznych rzutowań, niezainicjowanych zmiennych

7

Wdrażanie i zarządzanie

- Instalowanie bez wpływu na system
 - Aplikacje i komponenty mogą być współdzielone lub prywatne
- Wykonywanie równoczesne
 - Wiele wersji tego samego komponentu może współistnieć w systemie
- Podzespoły (assemblies)
 - Zawierają informacje o swoich zależnościach

8

Jednolite ale rozszerzalny

- Platforma .NET nie jest „czarną skrzynką”
- Każda klasa .NET może być rozszerzona przez dziedziczenie
- Można dodać komponenty i podsystemy

9

Języki .NET

- Wspólna specyfikacja (CLS)
- Języki dostarczane przez Microsoft
 - Visual Basic, C#, C++, J#, JScript
- Języki dostarczane przez inne firmy
 - APL, Cobol, Component Pascal, Eiffel, Fortran, Haskell, Mercury, Oberon, Oz, Perl, Python, RPG, Scheme, Smalltalk, Standard ML

10

Rodzaje aplikacji

- Aplikacje konsolowe
 - Kompilacja z linii poleceń, dowolny język, mały rozmiar
- Usługi Windows (Windows Services)
 - Prostota, pełna obsługa i instalacja
- Aplikacje okienkowe (Windows Forms)
 - Dziedziczenie wizualne, instalacja przez kopiowanie
- Aplikacje WWW – ASP.NET (Web Forms)
 - RAD, kontrolki walidacji, oddzielenie warstwy prezentacji od logiki, zaawansowane kontrolki, strony jako komponenty, obsługa farm serwerów, buforowanie, śledzenie i uruchamianie krokowe
- Usługi WWW (Web Services)
 - Następca RPC i DCOM, obsługa XML, SOAP, WSDL

11

Aplikacje konsolowe

```
class Hello
{
    static void Main(string[] args)
    {
        Console.WriteLine("Hello world!");
        for (int i = 0; i < args.Length; i++)
        {
            Console.WriteLine("args[{0}] = {1}",
                               i, args[i]);
        }
    }
}
```

12

Czas na przykład

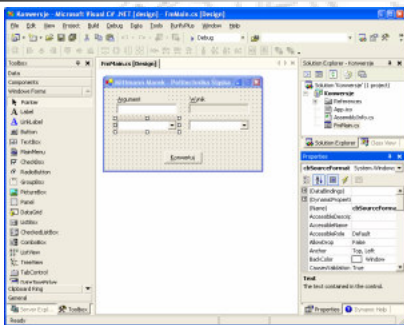
Aplikacje konsolowe

Aplikacje okienkowe

- Klasy wspierające tworzenie aplikacji okienkowych zawarte w platformie .NET
- Model programowania oparty na formularzach, kontrolkach, właściwościach i zdarzeniach
- Projektowanie interfejsu za pomocą narzędzi wizualnych VS.NET
- Dziedziczenie wizualne

14

Aplikacje okienkowe i VS.NET



15

Czas na przykład

Aplikacje okienkowe

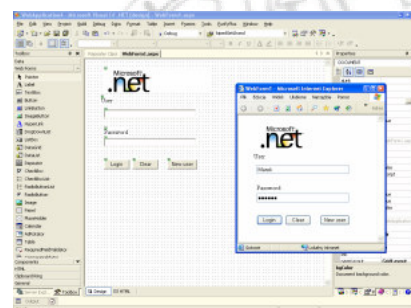
Aplikacje ASP.NET

- Model formularzy ASP.NET z wykorzystaniem kontroltek
- Programowanie oparte na zdarzeniach
- Tworzenie stron za pomocą narzędzi wizualnych



17

Aplikacje ASP.NET i VS.NET



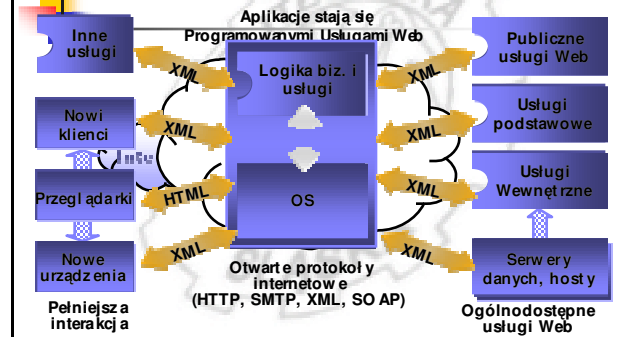
18

Usługi WWW

- Czym są usługi WWW ?
 - Komponent, który definiuje i dostarcza usługę do klienta
- Cechy
 - Potencjalnie ogólnodostępne
 - Klienci używają HTTP i XML
 - Mogą być tworzone w wielu językach
 - Klient abstrahuje od implementacji
- Podstawa aplikacji B2B i B2C

19

Usługi WWW



20

Wiadomości o .NET

melmac.iinf.polsl.gliwice.pl/~dotnet/
www.gotdotnet.com
www.msdn.microsoft.com

21

Dziękuję za uwagę