

Universitatea "Politehnica" din București

Facultatea de Electronică, Telecomunicații și Tehnologia Informației

WEB interactiv : tehnologia ASP.NET

Conducător științific

Conf.dr.ing. Ștefan STĂNCESCU

Absolvent

Cătălina-Alexandra RUSU

Anul 2015

CUPRINS

INTRODUCERE.....	3
CAPITOLUL 1. NOȚIUNI WEB	4
CAPITOLUL 2. ASP.NET.....	6
2.1. GENERALITAȚI.....	6
2.2. AVANTAJE ASP.NET	6
2.3. NOUȚĂȚI ADUSE DE ASP.NET.....	7
2.4. MODELUL DE COD.....	8
CAPITOLUL 3. ASP.NET VS.PHP	10
CAPITOLUL 4. ASP.NET VS.COLDFUSION	15
CAPITOLUL 5. MICROSOFT VISUAL STUDIO.....	20
5.1. NOȚIUNI GENERALE	20
5.2. CREAREA UNEI APLICAȚII DE TIP CONSOLĂ ÎN VISUAL STUDIO 2013.....	20
BIBLIOGRAFIE	23

INTRODUCERE

Platforma .NET este un nou cadru de dezvoltare a softului, sub care se vor realiza, distribui și rula aplicațiile de tip forme Windows, aplicații WEB și servicii WEB. Ea constă în trei părți principale: Common Language Runtime, clase pentru platforma și ASP.NET. O infrastructură ajutătoare, Microsoft .NET Compact Framework este un set de interfețe de programare care permite dezvoltatorilor realizarea de aplicații pentru dispozitive mobile precum telefoane inteligente și PDA-uri [13].

.NET Framework constituie un nivel de abstractizare între aplicație și kernel-ul sistemului de operare (sau alte programe) pentru a asigura portabilitatea codului; de asemenea integrează tehnologii care au fost lansate de către Microsoft începând cu mijlocul anilor '90 (COM, DCOM, ActiveX, etc) sau tehnologii actuale (servicii Web, XML).

Platforma constă în câteva grupe de produse:

1. Unelte de dezvoltare - un set de limbaje (C#, Visual Basic .NET, J#, Managed C++, Objective-C, Python, Smalltalk, Eiffel, Perl, Fortran, Cobol, Lisp, Haskell, Pascal, RPG, etc), un set de medii de dezvoltare (Visual Studio .NET, Visio), infrastructura .NET Framework, o bibliotecă cuprinzătoare de clase pentru crearea serviciilor Web (Web Services), aplicațiilor Web (Web Forms) și aplicațiilor Windows (Windows Forms).
2. Servere specializate - un set de servere Enterprise .NET, continuatoare ale lui SQL Server 2000, Exchange 2000, BizTalk 2000, etc, care pun la dispoziție funcționalități diverse pentru stocarea bazelor de date, email, aplicații B2B3.
3. Servicii Web - cel mai notabil exemplu este .NET Passport - un mod prin care utilizatorii se pot autentifica pe site-urile Web vizitate, folosind un singur nume și o parolă pentru toate. Deși nu este omniprezent, multe site-uri îl folosesc pentru a ușura accesul utilizatorilor.
4. Dispozitive - noi dispozitive non-PC, programabile prin .NET CompactFramework, o versiune redusă a lui .NET Framework: Pocket PC Phone Edition, Smartphone, Tablet PC, Smart Display, XBox, set-top boxes, etc.

CAPITOLUL 1. NOȚIUNI WEB

Aplicațiile web moderne sunt sisteme software complexe, iar dezvoltarea acestora necesită o abordare metodologică a proiectării lor. Similar cu proiectarea aplicațiilor software, proiectarea web implică utilizarea unei abordări sistematice și cuantificabile pentru realizarea specificațiilor, implementării, operațiilor și întreținerii aplicațiilor web de calitate superioară.

Elementul distinctiv al aplicațiilor web, comparativ cu aplicațiile software tradiționale, este modul în care este utilizat web-ul: de exemplu tehnologiile și standardele sale sunt utilizate ca o platformă de dezvoltare și ca platformă utilizator în același timp.

O aplicație online (sau aplicație web) este accesibilă printr-un browser pe internet sau într-o rețea locală (un intranet). Punctul forte al unei aplicații web este că poate fi folosită de un număr infinit de utilizatori, fără ca aceștia să trebuiască să o instaleze și configureze pe propriul calculator.

Ca toate aplicațiile client-server, aplicațiile web au două componente distincte [1]:

- **Clientul**, componenta cunoscută sub denumirea de front-end interface; browserul web prezintă interfața utilizatorului, acceptă intrările acestuia și trimite date serverului pentru procesare.
- **Serverul**, componenta cunoscută sub denumirea de back-end interface; serverul web răspunde cererilor clientului cu ajutorul unei pagini HTML care include instrucțiuni referitoare la generarea interfeței utilizatorului.

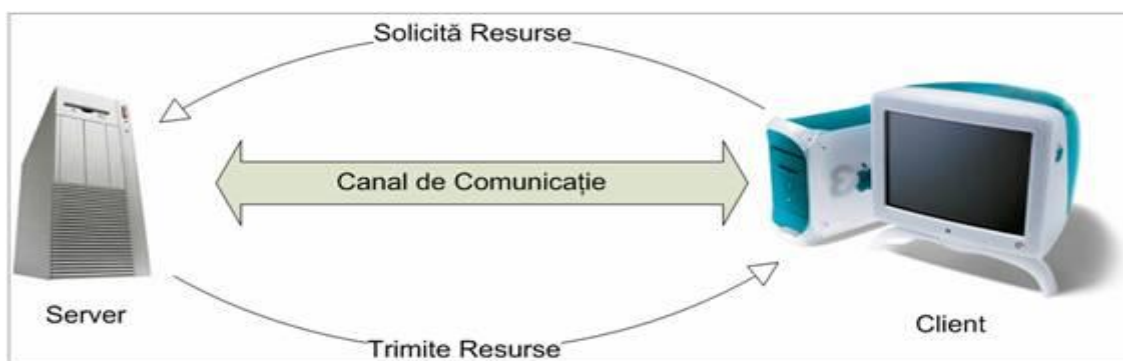


Figura 1. Arhitectura client-server

Noțiunea de client-server este întâlnită între oricare două mașini între care exista o cale de acces, o conexiune directă sau indirectă. De exemplu, se presupune ca avem o mașină pe care ținem o bază de date MySQL, o mașină pe care o folosim ca web server [2] (deci unde stocăm fișierele unui web site) și o mașină care va fi folosită drept client. Astfel, inițiem o cerere de la client către serverul web pentru a primi o anumită pagină în care vom vizualiza, de exemplu, produsele de pe un magazin online. Serverul web primește cererea și va acționa întocmai, însă datele necesare generării acestei pagini HTML care va conține informațiile despre produse, se află în baza de date, deci pe un alt server. Serverul web, prin intermediul librăriilor din PHP va iniția acum o cerere către serverul de MySQL pentru a întoarce informațiile despre produse, pe care cel din urmă le va furniza. În această ultimă situație serverul web este client pentru serverul de MySQL pentru că el este cel care inițiază o cerere de date iar baza de date va furniza răspunsul la această cerere. În final, cu datele primite se va furniza pagina HTML complet formatată.

CAPITOLUL 2. ASP.NET

2.1. GENERALITAȚI

.NET este soluția propusă de Microsoft pentru programarea aplicațiilor distribuite în Internet, fiind identificată prin următoarele trei componente esențiale:

- un mediu de execuție independent de limbaj optimizat pentru prelucrări distribuite — .NET Framework;
- un mediu de dezvoltare care oferă suport pentru mai multe limbaje de programare standardizate sau proprietate Microsoft — Visual Studio .NET;
- sistemul de operare care oferă suport pentru aplicațiile distribuite dezvoltate pe platforma .NET Framework — Windows Server.

ASP.NET este un set de tehnologii care ne permit crearea de aplicații web. Este evoluția de la Microsoft Active Server Pages (ASP), dar beneficiază de suportul platformei de dezvoltare Microsoft .NET [3], fiind lansat în ianuarie 2002 cu versiunea 1.0 a .NET Framework [5]. Conform [4], ASP.NET reprezintă un framework web gratuit utilizat pentru crearea site-urilor, aplicațiilor și serviciilor cu ajutorul HTML, CSS și JavaScript.

ASP.NET a fost construit pe Common Language Runtime (CLR), ceea ce permite programatorilor să scrie cod folosind orice limbaj suportat de .NET.

2.2. AVANTAJE ASP.NET

Una dintre cele mai importante calități ale ASP.NET îl reprezintă timpul redus necesar dezvoltării aplicațiilor web. Atât tehnologia în sine, cât și uneltele de dezvoltare de aplicații web de la Microsoft (cum ar fi Visual Web Developer Express - VWD) reduc considerabil timpul de dezvoltare al aplicațiilor web față de alte tehnologii – prin simplitatea unui limbaj de programare ”managed” de genul C# sau Visual Basic .NET, prin colecția bogată de biblioteci de clase și controale .NET .

Chiar dacă ASP.NET este gândit pentru a dezvolta aplicații web foarte complexe – prin faptul că se bazează pe .NET, prin faptul că se insistă pe un model de dezvoltare OOP, respectiv pe

separarea interfeței de logica aplicației – totuși, este extrem de simplu ca folosind ASP.NET să dezvoltăm aplicații mici, de genul magazinelor online, al aplicațiilor care sunt pur și simplu un ”front-end” pentru o bază de date, sau al site-urilor personale.

ASP.NET cuprinde toate tehnologiile necesare pentru a dezvolta o aplicație web, scriind o cantitate redusă de cod. Limbajele de programare care pot fi utilizate pentru a crea aplicații ASP.NET sunt cele suportate de platforma .NET – cum sunt Visual Basic .NET și C#, iar o altă caracteristică importantă a acestor limbaje (exceptând faptul că sunt ”managed”) este că au fost create având în vedere paradigma programării orientată pe obiecte. Totul din .NET, și evident din ASP.NET, este un obiect.

2.3. NOUȚĂȚI ADUSE DE ASP.NET

Majorele diferențe dintre ASP.NET și platformele de dezvoltare existente includ următoarele [5]:

- ASP.NET propune un modelul de programare în întregime obiect-orientat, care include o arhitectură bazată pe evenimente și controale, care încurajează încapsularea și re folosirea codului;
- ASP.NET oferă posibilitatea de a programa în orice limbaj acceptat de .NET;
- ASP.NET sprijină ideea de executare rapidă: paginile și componentele ASP.NET sunt compilate (fie la prima cerere, fie la instalarea site-ului pe server), în loc de a fi interpretate de fiecare dată când acestea sunt utilizate;
- modul de programare este orientat pe obiecte; orice pagină este văzută ca un obiect, derivat din clasa pusă la dispoziție de .NET Framework;
- are un mecanism încorporat care permite păstrarea stării paginii (chiar dacă protocolul de comunicare HTTP este fără stare);
- ASP.NET permite dezvoltarea de aplicații web bazate pe evenimente, similar cu situația întâlnită în cadrul aplicațiilor de tip formă Windows;
- este sensibil la browserul pentru care se face trimiterea codului HTML, alegând automat codul optimizat pentru o colecție de browsere;

- ASP.NET este gazduit de catre CLR; aceasta aduce beneficii notabile precum: managementul automat al memoriei prin garbage collection (eliberare a memoriei alocate dinamic [6]);
- tipuri de date sigure, verificabile; manipularea structurata a erorilor; fire de executie - pentru fiecare cerere facuta de catre un client, se lanseaza un fir de executie separat;
- usor de configurat.

2.4. MODELUL DE COD

O pagina web aspx este compusa din cod ASP.NET de forma:

- cod HTML; acesta este interpretat de către browser;
- controale web; acestea se recunosc prin tagurile care incep cu prefixul asp:, ca de exemplu:

```
<asp:CheckBox id="myCheckBox" Text="vreau bicicleta" runat="server" />
```

```
<asp:TextBox id="usernameText" Text="Nume de utilizator" runat="server" />
```

Controalele web sunt procesate pe server si vor produce cod HTML care este inteligibil pentru un browser.

- cod C# sau VB.NET

Acest ultim cod este executat pe server (deci browser-ul nu trebuie sa aiba niciun tip de suport pentru executarea de cod .NET); relativ la locatia in care se poate depune acest cod, avem posibilitatile:

1. cod inline, adica scris in interiorul paginii aspx; utilizat pentru pagini web simple;
2. cod “in spate” (behind), intr-un fisier separat cu extensia .cs (daca este cod

C#). Este modelul preferat pentru dezvoltarea de pagini.

Conectarea dintre pagina aspx si code-behind se face prin intermediul directivei Page de pe primul rand.

Exista doua diferente intre a avea tot codul intr-un singur fisier si a avea code-behind [7]. In modelul cu code-behind nu exista scripturi care sa aiba atributul runat = „server”.

Cea de-a doua diferenta consta in faptul ca la modelul cu code-behind directiva „Page” contine attribute care fac referire la un fisier extern (SamplePage.aspx.vb sau SamplePage.aspx.cs) si o clasa.

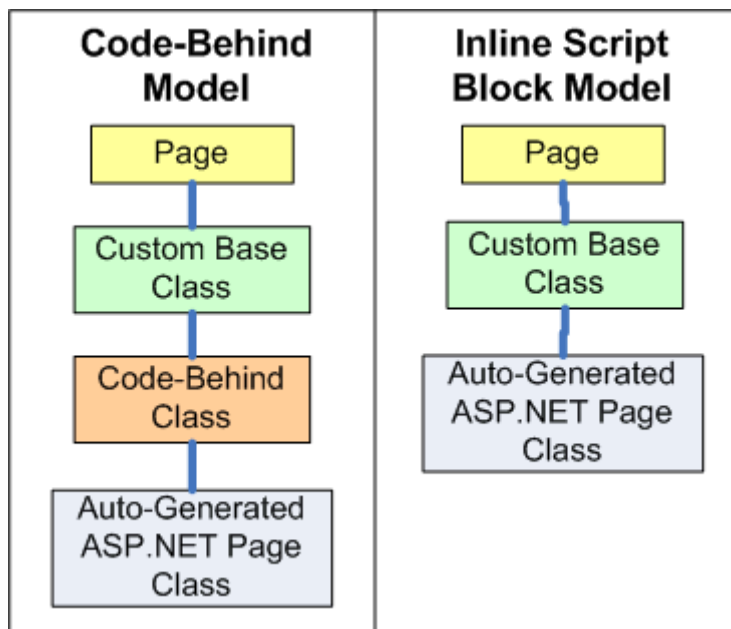


Figura 2. Modelul code-behind vs modelul cod inline [8]

CAPITOLUL 3. ASP.NET VS.PHP

Intr-unul din colturi este PHP, cel mai cunoscut limbaj de scripting de pe internet cu milioane de dezvoltatori dedicati si cu o legiune si mai numeroasa de fani. In celalalt colt se gaseste ASP. NET, o platforma sprijinita de Microsoft, ce foloseste un limbaj bazat pe .NET [9].

Costuri

PHP este un limbaj gratuit, pe cand ASP.NET este un produs Microsoft. Prin urmare apar anumite costuri asociate dezvoltarii ASP.NET, cum ar fi:

- cumpararea sistemului de operare Windows deoarece acest limbaj poate opera doar pe o masina Windows;
- ASP.NET are nevoie de un serviciu de hosting compatibil Windows;
- un mediu de dezvoltare. Cel mai cunoscut IDE (Mediu de Dezvoltare Integrat) pentru ASP.NET este Visual Studio. Microsoft ofera si o versiune gratuita, denumita Visual Studio Express.

Ca si incepator, nu sunt necesare costuri suplimentare pentru folosirea ASP.NET, insa odata ce se trece la nivelul profesionist, este necesara achizitionarea mediului Visual Studio, care poate costa si cateva sute de dolari. PHP, in schimb, ruleaza pe un serviciu de hosting Linux, este gratuit, si poate fi folosit si pe Windows sau MAC, fiind suportat de un numar mare de IDE-uri.

Performanțe

Performantele celor mai multe aplicatii web sunt o functie determinata de interactiunea intre script, server si baza de date. Majoritatea aplicatiilor web scrise in PHP urmeaza coada LAMP – sistem de operare Linux, server Apache, baza de date MySQL si limbaj PHP. Aceasta coada a fost optimizata, in anii precedenti, pentru a imbunatati performantele aplicatiilor.

Baza de date folosita in ASP.NET este MSSQL (Server SQL Microsoft), desi se poate folosi si MSOL. S-a descoperit ca un sistem PHP+MySQL este mai bun decat ASP.NET+MSSQL, insa diferenta este doar de cateva procente.

Un alt factor care poate afecta performanta este sistemul de operare si fisierul de sistem folosit pe un server. Majoritatea testelor au concluzionat ca Linux si fisierul ext4 au o performanta de I/O mai buna decat windows si NTFS.

In tabelul 1 sunt prezentate comparativ caracteristicile PHP-ului si cele ale ASP.NET-ului.

Caracteristici	PHP	ASP.NET
Limbaj de codare	Limbaj de scripting bazat pe C, C++ , cu accente de ASP, versiune mai veche. Suporta concepte orientate pe proces	Suporta mai mult de 25 de limbaje, dar cele mai folosite sunt C# si Visual Basic .NET. Majoritatea dezvoltatorilor aleg un anumit limbaj dar pot folosi si componente ce suporta si alte limbaje.
Aplicatia Logica de Compilare	Compilabil si poate rula ca un executabil.	Suportat, in modele de compilare dinamica si precompilare.
Memorarea in cache a iesirii in format full-page (pagina intreaga)	Fara suport nativ.	Suportat, diferitele versiuni ale memorarii fiind bazate pe unul sau mai multi parametri ale URL, tip de browser, o functie specializata sau orice combinatie a elementelor de mai sus.
Memorarea in cache a iesirii in format partial-page (pagina partiala)	Fara suport nativ.	Suport de tip built-in prin folosirea componentei User Controls. Datele si celelalte obiecte po fi memorate prin reguli sofisticate utilizand Cache API.
Acces Baza de Date	Are drivere pentru majoritatea bazelor de date de pe piata, dar si pentru cele open source.	Suporta OLE-BD si ODBC in mod direct, incluzand drivere native pentru Microsoft SQL Server si Oracle.
Iesire Baza de Date	Datele sunt returnate ca si variabile PHP si vor fi reprezentate ca orice alta variabila.	Foloseste sabloane pentru legarea datelor la comenzile de control ale serverului pentru usurinta la utilizare.
Componente externe	Poate avea diferite pachete, iar motorul Zend se poate modifica direct de dezvoltator, pentru ca este open source. Programatorii Zend lucreaza la caracteristici	Suport foarte bun pentru biblioteci native C si obiecte COM, dar si asamblari scrise in limbaje de tip .NET, inclusiv Managed C++. Nu are

	noi, ce vor permite obiectelor .NET sa fie apelate din PHP.	suport pentru CORBA sau clase Java.
XML/XSLT	Suportat in librarii si pachete din programe de completare	Suport usor de folosit pentru validare, XML DOM, XSLT, analiza documentelor XML.
Servicii Web XML	Pachete PHP dezvoltate pentru suportul serviciilor Web pe motorul Apache AXIS	Furnizeaza standarde de suport flexibile si extensive, pentru publicarea si consumul mult mai usor al serviciilor Web.
Starea sesiunii	Management bazat pe cookie-uri	Management bazat pe cookie-uri, printr-un mecanism de stocare de memorie interna, pentru un singur server, al unui server de stare centralizat sau al unei baze de date interne.
Functionalitate de tip built-in	Functii built-in pentru acoperirea celor mai obisnuite sarcini, pe care le poate face o aplicatie bazata pe pagini Web. Acceseaza librarii Java.	Are acces direct la librariile .NET, ceea ce ii da o functionalitate mare.
Expresii obisnuite	Suporta sintaxa obisnuita pentru expresii POSIX si Perl.	Suporta expresii compatibile Perl, prin caracteristici suplimentare cum ar fi potrivirea stanga-dreapta, expresii precompilate, grupuri, suport unicode complet. Permite utilizatorului sa specifice o functie ce va fi apelata printr-o operatie de inlocuire.
Depanare	Nu are un proces de depanare extensiv, se folosesc produse de la Zend si de la alti distribuitori pentru o depanare si testare mai buna.	Incluide informatii de urmarire extensive, care pot fi incluse in pagina curenta sau in una noua. Se face depanare interactiva, in timpul executiei paginii, prin Microsoft Visual Studio .NET, in plus fata de procesul de depanare pe partea clientului, dar si a procedurilor SQL Server.
Managementul erorilor	Nu poate sa faca error trapping, dar are mecanisme de management al erorilor.	Suporta un management al exceptiilor (impreuna cu un bloc de cod care se executa indiferent daca eroarea a aparut), aplicarea de exceptii specifice, dar si specificarea paginilor eronate pentru diferite tipuri de erori

		nedescoperite.
Manipularea Imaginilor	Fara suport intern, sunt disponibile componente suplimentare.	Detine catacteristici extensive de creare si manipulare a imaginilor (clase System.Drawing)
Refolosirea codului	Functii definite pentru utilizator, abilitatea de a crea clase, include fisiere	Comenzi de control pentru utilizatori, pentru Server, clase specifice.
Fir de executie	PHP are un model bun pentru firele de executie.	Realizeaza cereri simultate pentru pagini in fire de executie separate, insa fiecare pagina poate executa propriile fire.
Memorarea in cache a datelor	Are un mecanism limitat de memorare.	Include mecanisme cache API, pentru aproape toate tipurile de date, cu data de expirare in functie de timp, folosinta, dependenta de un fisier.
Internationalizare	Suport complet pentru Unicode	Foloseste Unicode si diferite codificari de caractere. Functiile de date si numere isi modifica iesirea in functie de setarile de cultura. Suport pentru localizarea dinamica a aplicatiilor.
SMTP, HTTP, FTP, POP3	Suport nativ pentru o gama larga de protocoale de Internet	Suportul SMTP depinde de Serviciul SMTP IIS. Suport bun pentru HTTP. Nu are suport pentru FTP, POP3 pe .NET, dar se gasesc in componente de la distribuitori terti.
Mediu de Dezvoltare Integrat	Sunt diisponibile numeroase unelte de dezvoltare pentru o gama variata de carractiristici la preturi ce variaza pana la cateva sute de dolari.	Sunt disponibile unelte gratuite pentr dezvoltarea ASP.NET de la Microsoft, cum ar fi ASP.NET Wevb Matrix. Cea mai folosita unealta este Visual Studio .NET ce are suport complet pentru limbajele .NET, unelte de creare si testare baze de date SQL, unelte pentru design Web, depanare avansata si alte caracteristici, ce pot fi vazute in Visual Studio Developer Center. Alte unelte, precum Borland C# Builder si Macromedia Dreamweaver MX, pot suporta ASP.NET.

Suport Server Web	Suport pentru orice server Web	Suport pentru versiuni IIS si Apache 2.0.
Suport Sistem de Operare	Sunt componente ce lucreaza cu Microsoft Windows, Mac, OS X, amiga, Solaris, Linux, Free BSD, AIX.	Suporta Windows 2000, Windows XP, Windows Server 2003

TABELUL1. Caracteristici PHP si ASP.NET [10]

CAPITOLUL 4. ASP.NET VS.COLDFUSION

Deși cadrele de nivel inferior nu pot fi ignorate, cea mai mare diferență între ColdFusion MX și ASP.NET, pe care dezvoltatorii nu o sesizează, se petrece la nivel superior – în modelele de dezvoltare și execuție de pagini. În timp ce detaliile variază, ColdFusion urmează același model de dezvoltare și execuție de pagini ca și ASP, PHP, JSP și alte limbaje de scripting Web. Mai precis, codul este încorporat în HTML și, în timp ce o pagină execută de sus în jos, ieșirea codului după execuție va lua locul codului în documentul HTML. Acest model de dezvoltare este ușor de înțeles, dar are și dezavantaje: unul ar fi lipsa separării între aplicația logică și forma de prezentare. Împietirea codului cu prezentarea va face codul mai greu de citit, ceea ce va duce la creșterea timpului și efortului depus în mentenanță și creează schimbări semnificative pentru dezvoltatorii ce nu se ocupă cu dezvoltarea grafică, ce au nevoie să modifice o pagină. De-a lungul anilor, ColdFusion a introdus un număr de căi de atenuare a lipsei de separare, inclusiv taburi specializate însă modelul fundamental rămâne [14].

ASP.NET, însă, realizează o modificare fundamentală față de modelul tradițional al dezvoltării aplicațiilor Web. În loc să fie liniar și orientat pe pagină, codul este puternic tipizat și orientat pe obiecte. Dezvoltarea codului este o împlinire a dezvoltării Web tradiționale și a dezvoltării interfețelor GUI pentru utilizatori. Elementele paginii sunt reprezentate prin obiecte tip server, fiecare cu propriile proprietăți, metode și evenimente. Un exemplu ar fi o pagină ce postează către ea însăși și postează informații de la un formular din baza de date. În ColdFusion, se pot scrie coduri de verificare a existenței unor formulare variabile corespunzătoare numelui unui buton de submit în HTML dacă pagina se încarcă pentru prima dată, sau ca și rezultat al supunerii formularului. Apoi se pot citi variabilele formularului, se poate valida și se poate face o inserare în baza de date. Dacă una sau mai multe variabile nu au trecut de validare, se va scrie un cod pentru evidențierea erorilor către utilizator și pentru păstrarea valorilor deja introduse. În ASP.NET, se va lega o funcție la evenimentul Click al unui Buton de control al serverului. În această funcție, o parte din cod va citi proprietățile de tip text ale câmpurilor formularului, reprezentat de obiecte de tip TextBox. În timp ce butonul

Click este apasat, validarea campului formularului a fost deja realizata de orice validator pe pagina, ceea ce se va ocupa si de prezentarea oricaror mesaje de erori. Fiecare TextBox se ocupa de pastrarea continutului in cazul in care utilizatorii vor sa editeze intrarile.

In tabelul 2 sunt prezentate comparative caracteristile ASP.NET-ului cu cele ale ColdFusion-ului.

Caracteristici	Coldfusion MX	ASP.NET
Limbaj de codare	Limbaj de codare CFML si CFScript. Majoritatea dezvoltatorilor folosesc o combinatie intre cele 2.	Suporta mai mult de 25 de limbaje, dar cele mai folosite sunt C# si Visual Basic .NET. Majoritatea dezvoltatorilor aleg un anumit limbaj dar pot folosi si componente ce suporta si alte limbaje.
Aplicatia Logica de Compilare	Compilare dinamica	Suportat, in modele de compilare dinamica si precompilare.
Memorarea in cache a iesirii in format full-page (pagina intreaga)	Memoreaza in cache diferite versiuni ale paginii in functie de parametrii URL-ului.	Suportat, diferitele versiuni ale memorarii fiind bazate pe unul sau mai multi parametrii ale URL, tip de browser, o functie specializata sau orice combinatie a elementelor de mai sus.
Memorarea in cache a iesirii in format partial-page (pagina partiala)	Fara suport intern.	Suport de tip built-in prin folosirea componentei User Controls. Datele si celelalte obiecte po fi memorate prin reguli sofisticate utilizand Cache API.
Acces Baza de Date	Suporta drivere pentru JDBC Tip III si IV, dar si pentru ODBC printr-o legatura JDBC-ODBC	Suporta OLE-BD si ODBC in mod direct, incluzand drivere native pentru Microsoft SQL Server si Oracle.
Iesire Baza de Date	Prin bucle cu tagurile CFOUTPUT si CFLOOP.	Foloseste sabloane pentru legarea datelor la comenzile de control ale serverului pentru usurinta la utilizare.
Componente externe	Suporta obiecte CORBA, Java si obiecte EJB, dar si librariile de taguri JSP si taguri speciale scrise in C++ si Java.	Suport foarte bun pentru biblioteci native C si obiecte COM, dar si asamblariscrise in limbaje de tip .NET, inclusiv Managed C++. Nu are suport pentru CORBA sau clase Java.
XML/XSLT	Suport pentru transformari XML	Suport usor de folosit pentru

	DOM si XSLT de baza, nu sunt permise transformari avansate precum folosirea spatiului de nume XML, validarea XSD si pparametrii XSLT.	validare, XML DOM, XSLT, analiza documentelor XML.
Servicii Web XML	Suporta publicarea si consumul serviciilor Web cu o versiune beta a platformei Apache Axis, insa exista probleme de compatibilitate ce pot fi corectate printr-un update ce foloseste Apache Axis 1.0. O alta limitare majora este cea a apelurilor asincrone la serviciile Web.	Furnizeaza standarde de suport flexibile si extensive, pentru publicarea si consumul mult mai usor al serviciilor Web.
Starea sesiunii	Bazat pe cookie-uri, printr-un mecanism de stocare de memorie interna, pentru un singur server sau printr-un proces de stare „client” folosind o baza de date interna. Sesiunile de cookie-uri sunt suportate insa sunt necesare modificari in codul existent.	Management bazat pe cookie-uri, printr-un mecanism de stocare de memorie interna, pentru un singur server, al unui server de stare centralizat sau al unei baze de date interne.
Functionalitate de tip built-in	Acopera majoritatea sarcinilor obisnuite necesare functionarii unei aplicatii Web. Poate accesa si librariile Java.	Are acces direct la librariile .NET, ceea ce ii da o functionalitate mare.
Expresii obisnuite	Suporta sintaxa expresiilor compatibile cu Perl 5, insa are si suport partial pentru Unicode.	Suporta expresii compatibile Perl, prin caracteristici suplimentare cum ar fi potrivirea stanga-dreapta, expresii precompilate, grupuri, suport unicode complet. Permite utilizatorului sa specifice o functie ce va fi apelata printr-o operatie de inlocuire.
Depanare	Include informatii de urmarire extensive, care pot fi incluse in pagina curenta sau in una noua. DreamWeaver MX si Homesite+ permit depanarea interactiva a paginilor in timpul executiei, desi este dificil de lucrat cu setarile initiale in oricare dintre mediile de dezvoltare integrate specificate.	Include informatii de urmarire extensive, care pot fi incluse in pagina curenta sau in una noua. Se face depanare interactiva, in timpul exeutiei paginii, prin Microsoft Visual Studio .NET, in plus fata de procesul de depanare pe partea clientului, dar si a procedurilor SQL Server.
Managementul erorilor	Suporta management structurat, specificand paginile cu erori specifice pentru diferite tipuri de erori nedescoperite	Suporta un management al exceptiilor (impreuna cu un bloc de cod care se executa indiferent daca eroarea a

		aparut), aplicarea de exceptii specifice, dar si specificarea paginilor eronate pentru diferite tipuri de erori nedescoperite.
Planificare	Suporta generarea de harti de baza in JPG, PNG sau format Flash	Nu are suport intern, insa se folosesc componente terte.
Manipularea Imaginilor	Nu are suport intern, insa se folosesc componente terte.	Detine catacteristici extensive de creare si manipulare a imaginilor (clase System.Drawing)
Refolosirea codului	Taguri specifice, functii definite pentru utilizator, Componente ColdFusion.	Comenzi de control pentru utilizatori, pentru Server, clase specifice.
Fir de executie	Fara suport, in afara de faptul ca cererile simultane de pagini se petrec in fire de executie separate.	Realizeaza cereri simultane pentru pagini in fire de executie separate, insa fiecare pagina poate executa propriile fire.
Memorarea in cache a datelor	Permite memorare bazata pe timp a rezultatelor date de interogari. Alte tipuri de date pot fi stocate in aplicatii sau variabile de sesiune, insa datele de expirare si blocarile vor fi introduse manual.	Include mecanisme cache API, pentru aproape toate tipurile de date, cu data de expirare in functie de timp, folosinta, dependenta de un fisier.
Internationalizare	Suporta siruri de date unicode si diferite codari de caractere, incluzand caracteristici de conversie numerara, conversie valutara si de date intr-un format specific. Nu are suport pentru localizarea dinamica a aplicatiilor.	Foloseste Unicode si diferite codificari de caractere. Functiile de date si numere isi modifica iesirea in functie de setarile de cultura. Suport pentru localizarea dinamica a aplicatiilor.
SMTP, HTTP, FTP, POP3	Suportat, chiar daca CFMAIL este vazut in general ca indisponibil.	Suportul SMTP depinde de Serviciul SMTP IIS. Suport bun pentru HTTP. Nu are suport pentru FTP, POP3 pe .NET, dar se gasesc in componente de la distribuitori terti.
Cautare full-text	Include o versiune de motor de cautare Verity K2. Poate folosi caracteristici interne furnizate de diferiti furnizori. Poate accesa alte motoare de cautare prin platformele COM si .NET API aferente.	Nu are motor de cautare intern, insa poate accesa MS Index Server pentru cautare full-text. Poate folosi caracteristici interne furnizate de diferiti furnizori. Serverul Verity K2 poate fi accesat prin platformele COM si .NET

		API, la fel ca si alte motoare de cautare.
Suport Server Web	Suporta IIS, Sun ONE, Netscape, IPlanet, Apache.	Suport pentru versiuni IIS si Apache 2.0.
Suport Sistem de Operare	Suporta Windows, Linux, Solaris, HP-UX	Suporta Windows 2000, Windows XP, Windows Server 2003

CAPITOLUL 5. MICROSOFT VISUAL STUDIO

5.1. NOȚIUNI GENERALE

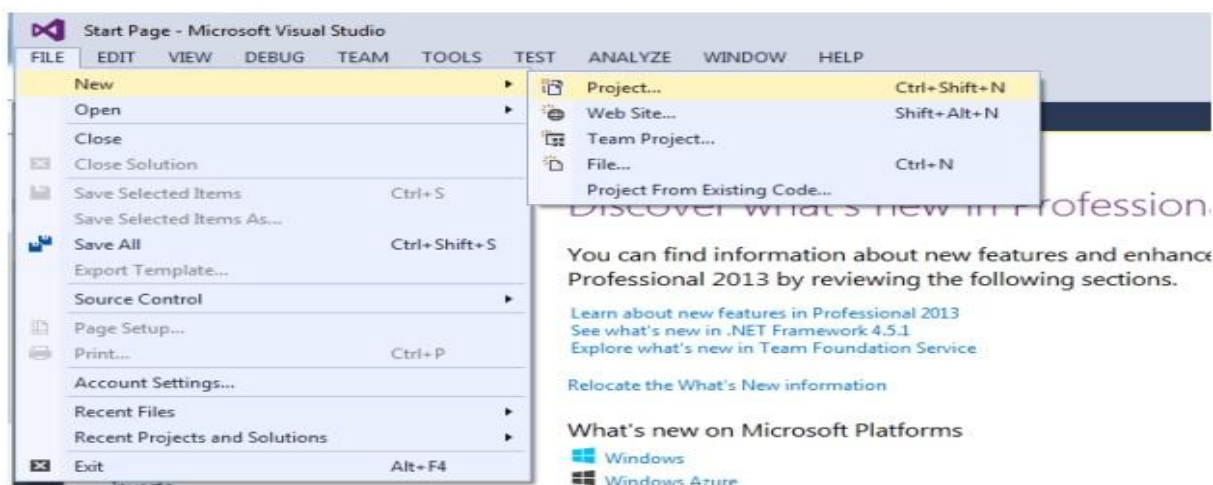
Microsoft Visual Studio este un mediu de dezvoltare integrat (integrated development environment - IDE) de la Microsoft. Acesta poate fi folosit pentru a dezvolta aplicații consolă și aplicații cu interfață grafică pentru toate platformele suportate de Microsoft Windows (ex. .NET Framework, Windows Mobile etc) [11].

Microsoft Visual Studio oferă editor, compilator/debugger și mediu de proiectare (designer) pentru mai multe limbaje de programare. Limbaje de programare incluse: **Microsoft Visual C++**, **Microsoft Visual C#**, **Microsoft Visual Basic**, **Microsoft Visual Web Developer**, **Team Foundation Server**.

5.2. CREAREA UNEI APLICAȚII DE TIP CONSOLĂ ÎN VISUAL STUDIO 2013

Pentru a crea o aplicație de tip consolă se vor parcurge următorii pași:

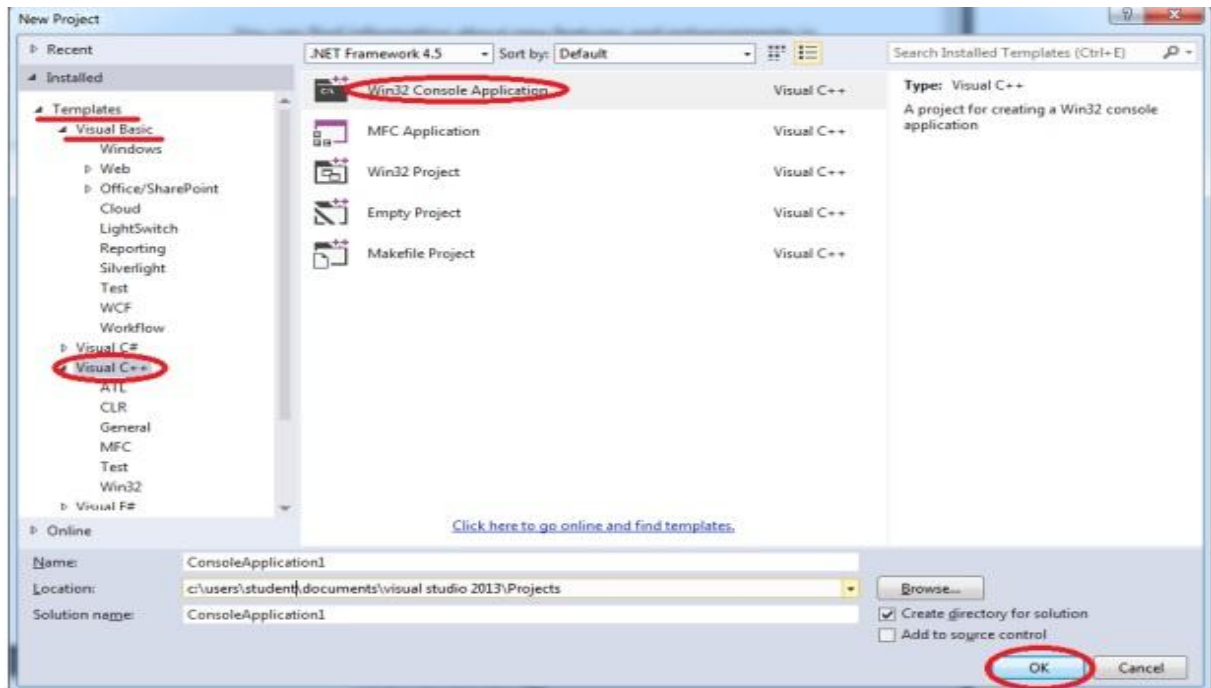
File-> New Project



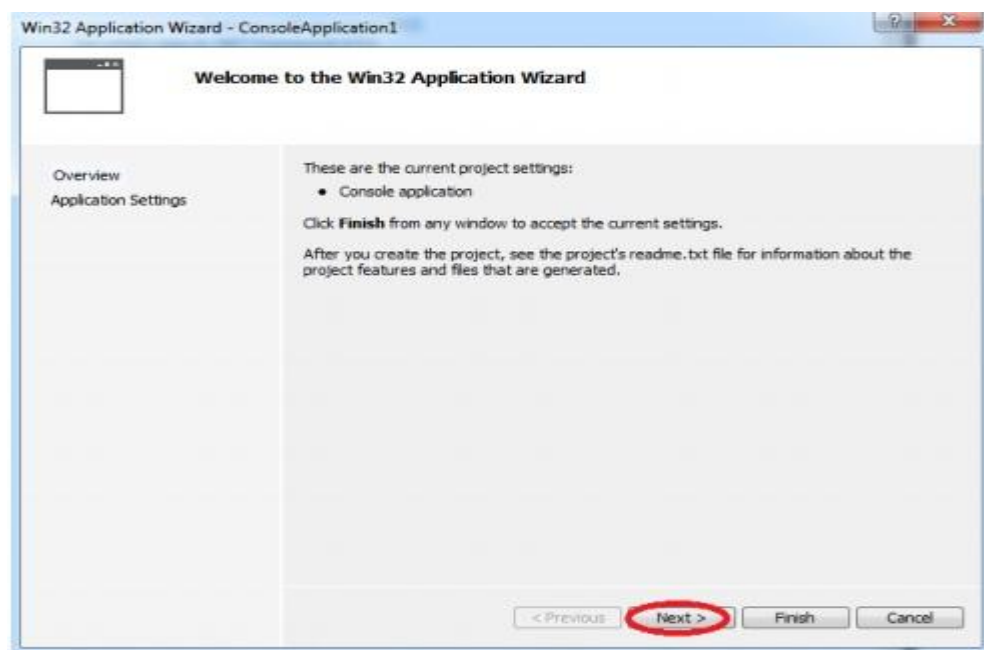
Din fereastra de dialog se va selecta Visual C++. Apoi din partea centrala a ferestrei se va selecta Win32 Console Application.

În partea de jos se va scrie un nume pentru aplicație și se va selecta locația unde să fie salvată această aplicație (o locație unde userul student are acces de citire, scriere și execuție).

După ce aceste informații au fost completate se va apăsa butonul OK.

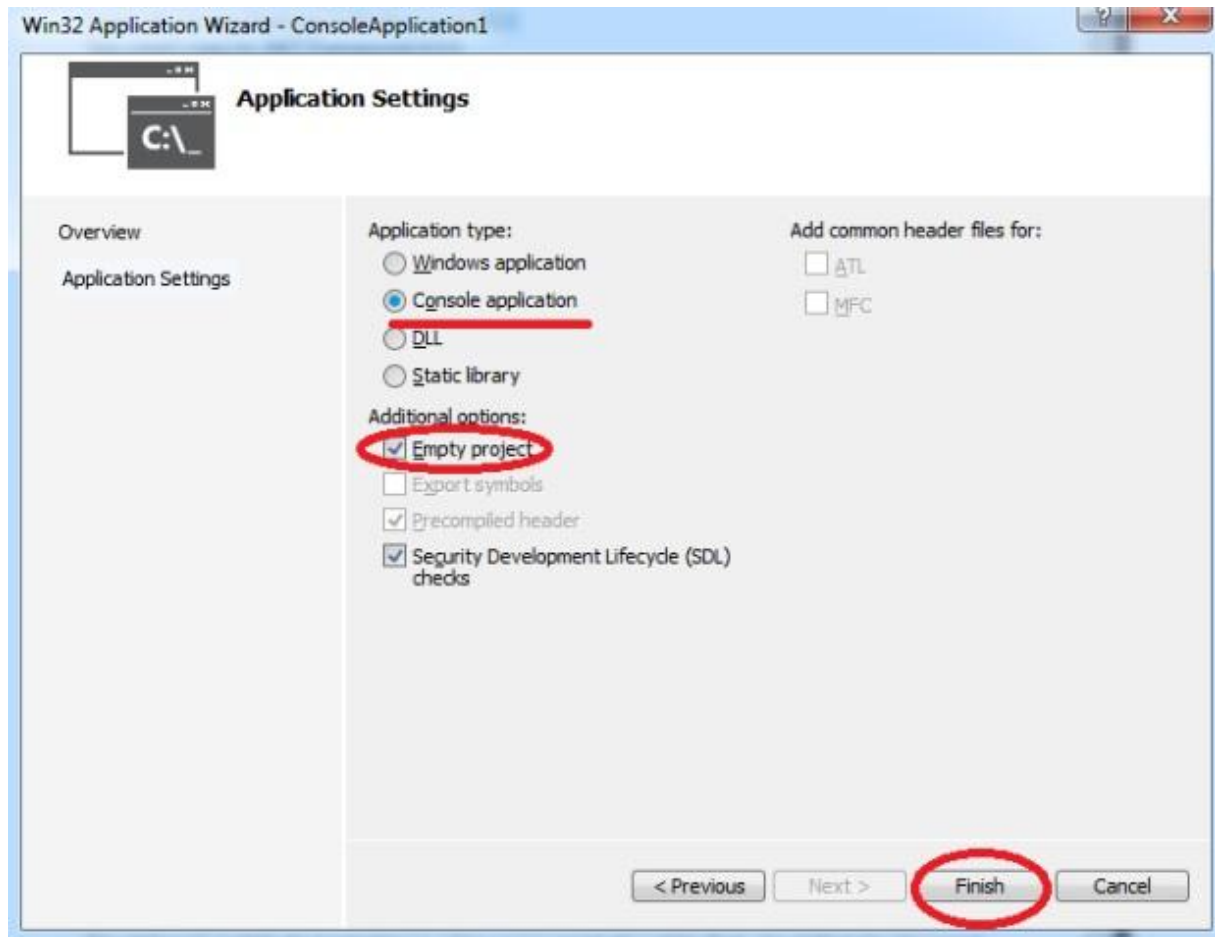


O nouă fereastră de dialog se va deschide, unde se va apăsa butonul Next:



O nouă fereastră de dialog se va deschide, unde Console application este selectat implicit și va rămâne așa.

Se va selecta Empty project și se va apăsa butonul Finish.



Mediul de dezvoltare se va deschide cu noul proiect salvat, dar care este gol.

BIBLIOGRAFIE

- [1] Tony Northrup and Mike Snell, “Web Applications Development with Microsoft .NET Framework 4”, SELF-PACED Training Kit, Microsoft Press, 2010
- [2] <http://www.inphpwetrust.com/arhitectura-client-server/>, accesat la data de 10.01.2015
- [3] “Programarea WEB cu Microsoft .NET”
- [4] <http://www.asp.net/>, accesat la data de 17.01.2015
- [5] Lucian Sasu, “Site-uri WEB dinamice in ASP.NET”, 5 octombrie 2009
- [6] <http://www.memorymanagement.org/glossary/g.html>, accesat la data de 16.01.2015
- [7] <http://msdn.microsoft.com/en-us/library/015103yb%28v=vs.140%29.aspx>, accesat la data de 17.01.2015
- [8] <http://www.4guysfromrolla.com/images/typeGraph.base.gif>, accesat la data de 17.01.2015
- [9] <https://www.udemy.com/blog/php-vs-asp-net/>, accesat la data de 21.01.2015
- [10] <https://msdn.microsoft.com/en-us/library/aa479002.aspx>, accesat la data de 27.01.2015
- [11] <http://www.math.uaic.ro/~mapetree/POO/>, accesat la data de 02.02.2015
- [12] <http://www.cs.upt.ro/sda/Tutorial%20VS.pdf>, accesat la data de 04.02.2015
- [13] Lucian Sasu, “Visual C#”, 30 Mai 2015
- [14] <https://msdn.microsoft.com/en-us/library/aa479305.aspx>, accesat la data de 04.02.2015