

Curs 1

Introducere

Objective

- Deprinderea principalelor concepte ale programării în .NET
- Programarea aplicațiilor Web utilizând ASP.NET Core
- Programarea aplicațiilor multiplatformă utilizând .NET MAUI
- Programarea accesului la date
- Limbaj de programare C# (principiile de programare orientată obiect învățate în anul 2 la POO sunt aceleași)
- Toate materialele sunt disponibile pe Platforma Moodle:
<https://econ.elearning.ubbcluj.ro/moodle> (Medii de programare și dezvoltare_IE3_24-25)

Echipa:

- Lect. Dr. Florina Covaci, Mihai Negrisan, Gabriel Pop



Masterat

- Continuarea traiectoriei de studii la unul din cele 3 masterate ale departamentului IE:
 - Business Analytics si Managementul Informatiilor (cu accent pe data science, data engineering, big data)
<https://econ.ubbcluj.ro/programe/bami/>
 - EBusiness (cu accent pe tehnologii web și sisteme distribuite)
<https://econ.ubbcluj.ro/programe/eb/>
 - Business Modelling and Distributed Computing (în limba engleză, cu accent pe modelare de business si calcul distribuit cu componenta de cercetare)
<https://econ.ubbcluj.ro/programe/bmdc/>

Lucrare licenta

- [Ghidul lucrărilor de licență cu specific de Informatică Economică](#)

<https://econ.ubbcluj.ro/departamente/studenti.php?c=4>

Tipuri:

- A. Proiectare și implementare de aplicații (Web, desktop, mobile etc.) sau de componente software
 - B. Cercetare științifică orientată empiric (formularea de ipoteze de cercetare și validarea/testarea lor prin analiză non-trivială de date colectate sau experimente)
 - C. Studii de caz privind administrarea și configurarea de medii/platforme IT
- Sablon de redactare FSEGA
- https://econ.ubbcluj.ro/n1.php?id_m=14&m=Examen%20finalizare%20studii



Evaluare

- IE – Disciplina obligatorie
 - Probe examen:
 - Probă teoretică: 40%
 - Examen grila în sesiune (20 intrebari*0.45=9+1pct oficiu) – o singura varianta de raspuns corecta
 - Proiect : 40% (cerinte minime proiect – vezi Moodle)
 - Examinat prin sustinere în perioada 8-17 ian. 2025
 - Activitate laborator 20% - 1 data pe sapt; laboratorul cu sarcina efectuata se incarca in saptamana curenta! (in assignment dedicat)
- Promovare examen: **minim 5** la fiecare probă de examen
- !!Atentie – media ponderata ≥ 4.5 (ex. $0.4*5+0.4*5+0.2*1=4.2$)



Cerinte minime pentru proiect IE

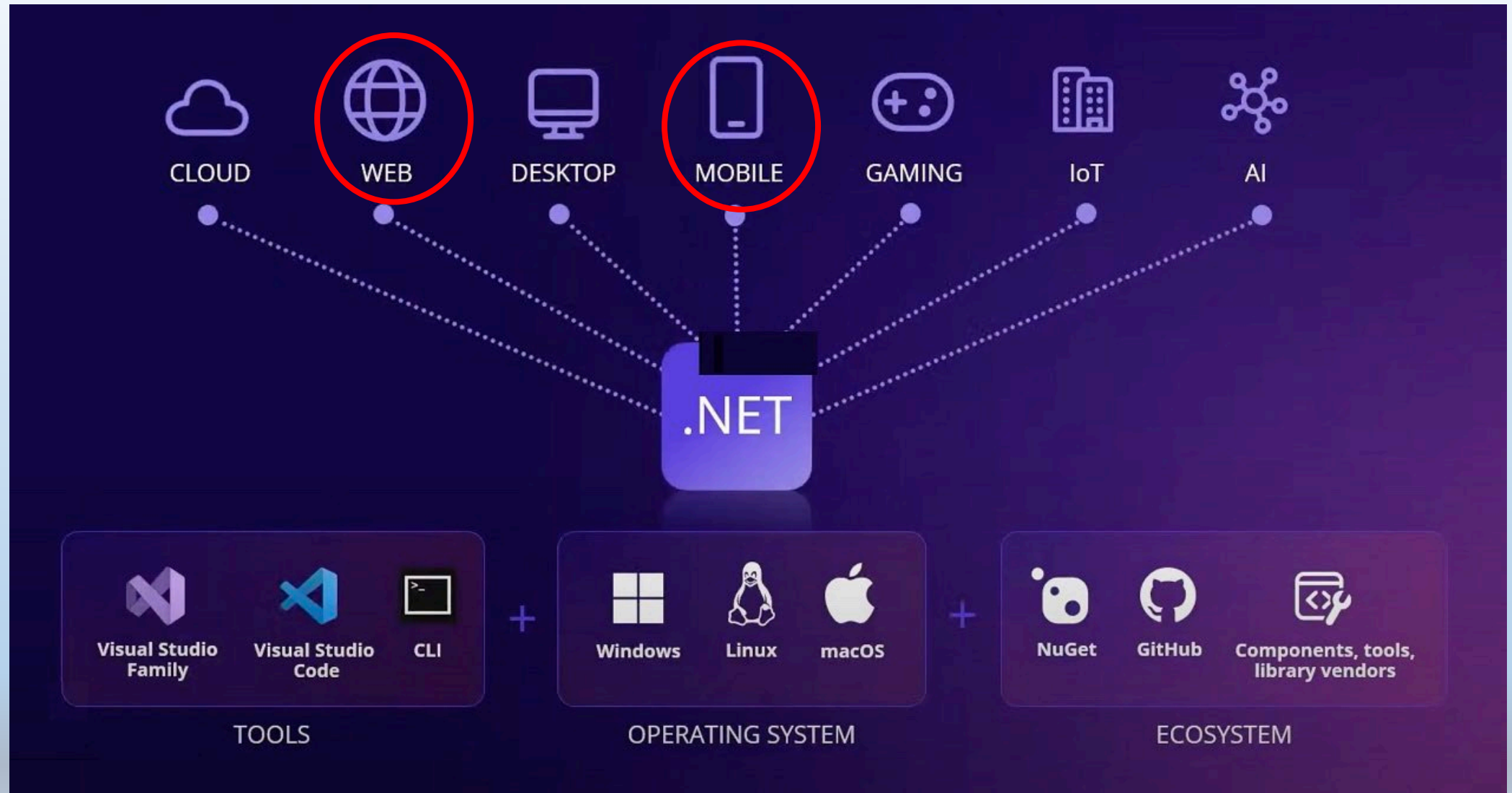
- 1. Realizarea unui sistem informatic (cu o tematica unitara – ex. Sistem informatic pentru o clinica stomatologica) care va contine doua proiecte de urmatoarele tipuri utilizand tehnologiile indicate:
 - I. aplicatie web (ASP.NET Core) ex. Programare pacienti; Review medic dupa efectuarea consultatiei/tratamentului
 - II. aplicatie mobila (.NET MAUI) ex. Programare pacienti; Trimitere notificare cu o zi inainte de data programata
- 2. Structura bazei de date pentru intreg sistemul informatic: existenta a cel putin 5 tabele cu relatii intre ele. Nu e obligatoriu ca cele 5 tabele sa fie utilizate toate in fiecare proiect ci se pot utiliza doar o parte din ele/proiect unele din tabele putand fi comune celor doua proiecte (minim 3 tabele/tip proiect)
- 3. Crearea interfetei pentru fiecare proiect si executarea de operatii CRUD (Create, Read, Update, Delete) asupra TUTUROR tabelelor din fiecare proiect conform relatiilor dintre aceste tabele -> adaugare, modificare, stergere, vizualizare a datelor
- 4. Validarea datelor de intrare pentru fiecare tip de proiect
- 5. Autentificare si autorizare
- Observatii:
 - 1. TEMATICA TREBUIE SA FIE DIFERITA DE CEA DIN LABORATOARE! (Atentie proiectul va fi notat cu nota 1 daca doar traducem numele de entitati din engleza in romana (ex. Books vs. Carti sau utilizam entitati foarte similare ex. Movies vs Books) !)
 - 2. Proiectul se realizeaza intr-o ECHIPA formata din DOUA persoane
 - 3. Pentru fiecare tip de aplicatie aveti exemple de implementari in fisierele aferente laboratoarelor.
 - 4. Implementarea nu trebuie neaparat sa fie identica cu cea din laboratoare, dar e bun punct de sprijin
 - 5. Proiectele copiate primesc automat nota 1



CONCURS : "Ghiceste pentru 10!"

- Studentii care participa activ la cursul de Medii de programare vor primi pe emailul institutional (stud.ubbcluj.ro) cate o litera pentru fiecare activitate notata in cadrul intalnirilor de la curs.
- Litera primita face parte dintr-un cuvant. Cuvantul reprezinta un termen/notiune invatata pe parcursul semestrului 1 la disciplina Medii de programare si dezvoltare.
- Primii 3(4) studenti (in functie de data si ora email-ului) care trimit raspunsul corect pe email florina.covaci@econ.ubbcluj.ro pana in Ian 2024 primesc nota 10 la proba teoretica pentru disciplina Medii de programare si dezvoltare.
- Pentru a gasi si trimite raspunsul puteti lucra individual sau in echipe de maxim 2 persoane. Pentru ca echipa sa fie valida, fiecare membru din echipa trebuie sa fii primit pe email minim o litera.
- Raspunsurile vor fi trimise utilizand email-ul institutional (stud.ubbcluj.ro). In cazul in care lucratii individual raspunsul corect va fi atribuit studentului caruia ii apartine adresa de email. In cazul in care lucratii in echipa, studentul care trimite emailul va mentiona in email si componenta echipei.
- Succes!





ASP.NET Core

- ASP.NET - O unealta Microsoft C# pentru dezvoltarea de aplicatii web
- Competitori: Java Spring, PHP Laravel, Node.js Express
- .NET Platforma de dezvoltare Microsoft din anul 2001
- ASP – Active Server Pages- pagini web dinamice, de cele mai multe ori conectate la o baza de date
- Core – open source, cross-platform



ASP.NET Core

- Necesitatea ca ASP.NET si Entity Framework sa ruleze pe alte SO in afara de Windows
- Arhitectura care usureaza testarea
- Integrarea de framework-uri client-side – Blazor –creare de UIs utilizand C# in loc de JavaScript.
- Suporta instalarea de versiuni diferite ale .Net Core – side by side versioning

Tehnologii Web

- HTML
- CSS
- JavaScript si TypeScript
- Librarii de scripting

HTML si CSS

- Cascade Style Sheets - definesc felul in care arata pagina web
- HTML tagul pentru list item definea modul in care este vizualizat-cerc,disc,patrat ->CSS
- Se pot utiliza template-uri Bootstrap - o colectie de conventii CSS si HTML care pot fi adaptate facil
- www.getbootstrap.com - documentatie si template-uri

JavaScript si TypeScript

- Putem modifica dynamic elemente client-side
- ECMAScript – standard care defineste functionalitatile curente si vitoare pentru JavaScript.
- Implementarea Microsoft pentru JavaScript se numeste JScript.
- TypeScript similar cu JavaScript. Sintaxa TypeScript se bazeaza pe JavaScript, dar aduce noi functionalitati (ex. Cod puternic tipizat, adnotari)
- Documentatie la : www.typescriptlang.org.

Librarii de scripting

- Pot fi utilizate server-side impreuna cu ASP.NET Core
- jQuery (<http://www.jquery.org>) –utilizata pentru a gestiona unitar modul in care diferite browsere gestioneaza evenimentele
- Angular (<https://angular.io>) librerie de la Google bazata pe patternul MVC creata pentru a simplifica dezvoltarea si testarea aplicatiilor web de tip single-page (Spre deosebire de ASP.NET MVC, Angular ofera patternul MVC in codul client-side)
- React (<https://reactjs.org>) librerie de la FaceBook care ofera functionalitati prin care se pot actualiza facil interfetele utilizator pe masura ce datele se modifica in background
- Visual Studio ofera template-uri pentru Angular si React

C# proprietati

- **C#** oferă un mecanism numit **proprietăți**, care simplifică modul în care se accesează și modifică câmpurile private dintr-o clasă. Proprietățile sunt definite folosind sintaxa get și set, spre deosebire de Java, unde accesul la câmpuri se face în general prin metode getter și setter separate.

```
//Java
```

```
public class Persoana {  
    private String nume;
```

```
  
    public String getNume() {  
        return nume;  
    }
```

```
  
    public void setNume(String nume) {  
        this.nume = nume;  
    }  
}
```

```
// C# - Proprietăți
```

```
public class Persoana {  
    public string Nume { get; set; } // Proprietate automată  
}
```

Namespace vs. Package

În **Java**, spațiile de nume sunt denumite **package** și sunt definite folosind cuvântul cheie package.

//Java

```
package com.exemplu;
```

```
public class Main {  
    public static void main(String[] args) {  
        System.out.println("Salut, Java!");  
    }  
}
```

În **C#**, spațiile de nume sunt denumite **namespace**. De asemenea, toate clasele sunt grupate în spații de nume și utilizatorii pot folosi using pentru a aduce aceste namespace-uri în domeniul de utilizare.

//C#:

```
using System;
```

```
namespace Exemplu {  
    class Program {  
        static void Main(string[] args) {  
            Console.WriteLine("Salut, C#!");  
        }  
    }  
}
```


Aplicatii Web cu ASP.NET Core

- De tip Server-side
 - Razor Pages
 - MVC
- De tip Client-side
 - Blazor
 - Angular
 - React