

FIȘA DISCIPLINEI

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea „Tibiscus” din Timișoara
1.2. Facultatea	Calculatoare și Informatică Aplicată
1.3. Departamentul	Informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclul de studii	Licență
1.6. Programul de studii/Calificarea	Informatică

2. Date despre disciplină

2.1. Denumirea/codul disciplinei	Tehnologii WEB (TWeb) - LIN 218						
2.2. Titularul activității de curs	Conf. univ.dr. Florentina Anica PINTEA						
2.3. Titularul activității de seminar	Inf. Tiberiu-Gabriel DRAGOTĂ						
2.4. Anul de studiu	2	2.5. Semestrul	1	2.6. Tipul de evaluare	E	2.7. Regimul disciplinei	Conținut ¹ =DS Obligatorietate ² =DOP

3. Timpul total estimat

3.1. Numărul de ore pe săptămână	2	din care 3.2. curs	1	3.3. laborator	2
3.4. Total ore din planul de învățământ	42	din care 3.5. curs	14	3.6. laborator	28
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					14
Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate și pe teren					14
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					14
Tutoriat					6
Examinări					6
Alte activități					4
3.7. Total ore studiu individual	58				
3.8. Total ore pe semestru	100				
3.9. Numărul de credite	4				

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Programare 1, Baze de date, Rețele de calculatoare, Elemente de web design
4.2. de competențe	Analiza cerințelor, documentarea, modelarea, proiectarea, dezvoltarea, testarea și evaluarea unui proiect informatic

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sală de curs dotată cu tablă albă și videoproiector, calculator, Google Classroom
5.2. de desfășurare a seminarului/laboratorului	Sală de laborator dotată cu tablă, laptop/proiector, calculatoare, rețea, legătură internet, software adecvat, Google Classroom Prezență la orele de laborator minim 50%. Termenele predării temelor de laborator sunt stabilite de titular de comun acord cu studenții.

6. Obiectivele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei

Cunoștințe	• Înțelegerea noțiunilor elementare despre paginile WEB statice și dinamice • Dobândirea informațiilor despre tehnologiile web necesare realizării, gestionării și întreținerii paginilor web • Explicarea unor aplicații soft existente, pe niveluri de abstractizare (arhitectură, pachete, clase, metode) utilizând în mod adecvat cunoștințele de bază
------------	--

Abilități	Proiectează, planifică, construiește, dezvoltă aplicații software scalabile și utilizează eficient resursele hardware și software. Proiectează cod valid și semantic pentru a structura conținutul unei pagini web. Inițiază, pregătește, realizează, propune metode de dezvoltare a proiectelor informatice complexe
Responsabilitate și autonomie	Folosește gândirea logică, analizează enunțul problemelor, selectează metoda specifică de rezolvare a acestora și utilizează scheme logice și diagrame de lucru în rezolvarea problemelor din tematica parcursă la disciplinele din curiculă. Formulează soluții de rezolvare adecvate și generează idei inovative. Capacitatea de planificare și organizare a etapelor de dezvoltare a unei aplicații web, lucrând atât individual, cât și în echipă.

7. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1. Obiectivul general al disciplinei	- Disciplina acoperă noțiunile de bază privind tehnologiile web necesare realizării , gestionării și întreținerii paginilor web. Un obiectiv important al acestui curs îl reprezintă înțelegerea și realizarea aplicațiilor web de tipul client – server.
7.2. Obiectivele specifice	- Lucrările practice se bazează pe dezvoltarea de aplicații în diverse limbaje destinate aplicațiilor web cum ar fi HTML, Java Script - Aprofundarea cunoștințelor referitoare la de programare în JavaScript

8. Conținuturi

8.1. Curs – 14 ore	Metode de predare	Observații
1. Introducere în tehnologiile de programare web Noțiuni fundamentale despre web, front-end și back-end	Expunerea interactivă, documentarea pe web, exemplificarea, problematizarea, conversația euristică	Predarea se bazează pe folosirea videoproietorului Materialele de curs sunt: notele și prezentările de curs (disponibile și pe Google Classroom)
2. HTML și CSS – Crearea și formatarea paginilor web Elemente HTML esențiale. Structura și formatare		
3. Introducere în JavaScript. Sintaxă de bază, variabile, funcții, structuri de control. Manipulare DOM	Expunerea interactivă, documentarea pe web, exemplificarea, problematizarea, conversația euristică	
4. Crearea și formatarea paginilor web utilizând concepte Javascript avansate. Obiecte, array-uri, funcții anonime, scope		
5. Funcții HOF (High-Order Functions), closures și arrow functions		
6. Validare și interacțiune cu utilizatorul în JavaScript Formulare, evenimente, mesaje de eroare și confirmare	Expunerea interactivă, documentarea pe web, exemplificarea, problematizarea, conversația euristică	
7. Funcția de conectare (persoană, verifyFunction) Autentificare de bază pe partea clientului		
8. Programare asincronă în JavaScript Promises, async/await, gestionarea evenimentelor		
9. Introducere în Node.js Mediul de execuție, instalare, module de bază		
10. Introducere în NPM (Node Package Manager) Instalarea, utilizarea și gestionarea pachetelor		
11. Sisteme de gestionare a pachetelor (Package Management System)		
12. Crearea unui server web cu Node.js Modulele http și express	Expunerea interactivă, documentarea pe web, exemplificarea, problematizarea, conversația euristică	
13. Introducere în bazele de date nerelaționale – MongoDB și Mongoose Concepte de bază, conectarea la baza de date Modele și scheme de date		
14.REST API in NodeJs, folosind MongoDB ca baza de date nerelationala si Mongoose ca si ORM pentru a lucra cu baza de date		
Bibliografie		
1. Eduard Pintilie, Realizarea de elemente vizuale cu JavaScript, 2023 - Programare creativă: realizarea d		

elemente vizuale cu JavaScript / https://www.academia.edu/97235670/Programare_creativ%C4%83_realizarea_de_elemente_vizuale_cu_JavaScript 2. A. Silberschatz, P. Baer Galvin, G. Gagne, Operating Systems Concepts, 9th Edition / https://os-book.com/OS9/index.html 3. Andrew S. Tanenbaum, Modern operating systems, Second edition, Publisher: Prentice Hall, ISBN 0-13-031358-0 / https://www.dc16iupat.org/wp-content/uploads/2020/04/Modern.Operating.Systems.2nd.Ed_.by_.Tanenbaum-not-scanned-1-1.pdf 4. Programare asincrona in javascript, https://www.youtube.com/watch?v=XzhuuaE8R1w 5. https://nodejs.org/en/ 6. Un mini-tutorial JavaScript, https://www.slideshare.net/slideshow/un-minitutorial-javascript/38241492 7. Web technologies tutorial http://www.w3schools.com/ 8. JavaScript Resources - https://www.javascript.com/resources		
8.2. Laborator (teme de laborator, proiecte etc.) – 28 ore	Metode de predare/învățare	Observații
1. Introducere în dezvoltarea Frontend cu JavaScript Familiarizarea cu mediul de lucru	Rezolvare aplicații practice împreună cu studenții. Discuții. Lucrul în grup organizat pe calculator/laptop Învățare prin colaborare și descoperire	Predarea se bazează pe parcurgerea punctelor esențiale din platformele de laborator. Studenții implementează și evaluează probleme. Materialele didactice sunt disponibile și pe Google Classroom.
2. Funcții în JavaScript – Definire, Hoisting și Contextul „this”		
3. Closures și gestionarea memoriei Closures: definiție, avantaje, riscuri		
4. Funcții de ordin superior (HOF) și funcții ca obiecte map(), filter(), reduce() – exemple practice		
5. Programare asincronă în JavaScript Callbacks, Promises, async/await Exemple de cereri HTTP cu fetch()	Rezolvare aplicații practice împreună cu studenții. Discuții. Lucrul în grup organizat pe calculator/laptop Învățare prin colaborare și descoperire	Predarea se bazează pe parcurgerea punctelor esențiale din platformele de laborator. Studenții implementează și evaluează probleme. Materialele didactice sunt disponibile și pe Google Classroom.
6. Operatorii în JavaScript și exemple practice Operatorii aritmetici, logici, de atribuire și comparație Operatorii de destructurare, spread/rest Exercițiu: mini-calculator + manipulare array și obiecte		
7. Introducere în Node.js Ce este Node.js și cum se folosește. Rulare cod JS în afara browserului Exercițiu: script Node.js care citește și scrie fișiere		
8. Node Package Manager (NPM) Instalarea și gestionarea pachetelor. Pachete utile (lodash, axios, express)		
9. Fișierul package.json și managementul proiectului Crearea și configurarea unui proiect cu npm init. Scripturi, versiuni, dependențe	Rezolvare aplicații practice împreună cu studenții. Discuții. Lucrul în grup organizat pe calculator/laptop Învățare prin colaborare și descoperire	Predarea se bazează pe parcurgerea punctelor esențiale din platformele de laborator. Studenții implementează și evaluează probleme. Materialele didactice sunt disponibile și pe Google Classroom.
10. Crearea unui server web cu Express.js Rute, middleware, gestionarea cererilor și răspunsurilor		
11. Conectarea la o bază de date MongoDB Introducere în MongoDB și Mongoose		
12. Framework React – Noțiuni de bază Structura unui proiect React, componente, JSX		
13. Framework Angular – Principii de funcționare Instalare Angular CLI, componente, data binding Exercițiu: mini-aplicație CRUD	Rezolvare aplicații practice împreună cu studenții. Discuții. Lucrul în grup organizat	Predarea se bazează pe parcurgerea punctelor esențiale din platformele de laborator. Studenții implementează și evaluează probleme.

simplificată	pe calculator/laptop Învățare prin colaborare și descoperire	Materialele didactice sunt disponibile și pe Google Classroom.
14. Framework Vue.js – Introducere și comparație Concepte de bază: instanțe, directive, reactivitate Diferențe față de React și Angular Exercițiu: aplicație mică cu input-uri reactive		
Bibliografie Aceleași ca la curs		

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

Acest curs, prin competențele pe care le dezvoltă și prin extinderea orizontului de cunoaștere pe care o presupune, constituie un pas esențial în formarea viitorilor programatori sau analiști din domeniile dinamice ale industriei software – aplicații distribuite, servicii web și aplicații la nivel de întreprindere.

Conținutul este în mare măsură aliniat cu structura cursurilor similare din alte universități și acoperă o arie tematică fundamentală pentru înțelegerea domeniilor de aplicabilitate, precum și pentru proiectarea și implementarea soluțiilor informatice moderne.

Activitățile teoretice și practice (de laborator) vizează dobândirea cunoștințelor și competențelor necesare pentru proiectarea și realizarea aplicațiilor web utilizând limbajul JavaScript, pornind de la cerințe concrete.

10. Evaluare

Tip de activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
10.1 Curs	Înțelegerea noțiunilor elementare despre paginile WEB statice și dinamice Dobândirea informațiilor despre tehnologiile web necesare realizării, gestionării și întreținerii paginilor web Capacitatea de a implementa aplicații web prin utilizarea framework-ului web studiat.	Examen scris în sesiunea de examene	40%
10.2 Seminar / laborator	Proiectează, planifică, construiește, dezvoltă aplicații software scalabile și utilizează eficient resursele hardware și software. Proiectează cod valid și semantic pentru a structura conținutul unei pagini web. Inițiază, pregătește, realizează, propune metode de dezvoltare a proiectelor informatice complexe Realizarea unei aplicații web Capacitatea de a proiecta, implementa și testa un proiect pentru o aplicație în JavaScript	Probă practică Evaluare individuală Implementarea și susținerea unui proiect de echipă (maxim 3 studenți în echipă).	60%
10.3. Standard minim de performanță			
Pentru nota 5 - Capacitatea de a scrie cod în JavaScript după cerință Pentru nota 10 - Proiectarea, dezvoltarea, testarea și explicarea unei aplicații web în JavaScript. Deployng web. Testing Nota finală se calculează ca medie ponderată a notelor acordate pentru cele două componente (curs + laborator). Examenul se consideră promovat dacă media este cel puțin 5 (nu e necesar ca fiecare notă să fie mai mare de 5). La fiecare dintre sesiunile de examen (inclusiv cele de restanță și măriri) nota se calculează după aceeași regulă.			

Notă:

- 1) Regimul disciplinei (conținut) - pentru nivelul de licență se alege una din variantele: DF (disciplină fundamentală) / DS (disciplină de specialitate) / DC (disciplină complementară).
- 2) Regimul disciplinei (obligativitate) - se alege una din variantele: DO=discipline obligatorii (impuse) / DOP=discipline opționale/ DFc (disciplină facultativă).

Data completării
20.09.2025

Titular de disciplină
Conf. univ.dr. Florentina Anica PINTEA

Data avizării în departament
26.09.2025

Director de departament
Conf.univ.dr. Victoria IORDAN