

# FIȘA DISCIPLINEI

## 1. Date despre program

|  |   |
|--|---|
| 1.1. Instituția de învățământ superior | Universitatea Petrol-Gaze din Ploiești  |
| 1.2. Facultatea                        | Inginerie Mecanică și Electrică         |
| 1.3. Departamentul                     | Automatică, Calculatoare și Electronică |
| 1.4. Domeniul de studii universitare   | Calculatoare și Tehnologia Informației  |
| 1.5. Ciclul de studii universitare     | Licență                                 |
| 1.6. Programul de studii universitare  | Calculatoare                            |

## 2. Date despre disciplină

|   |   |
|---|---|
| 2.1. Denumirea disciplinei                          | <b>Instrumente pentru dezvoltarea programelor</b> |
| 2.2. Titularul activităților de curs                | Șef lucr. dr. ing. Roșca Cosmina-Mihaela          |
| 2.3. Titularul activităților seminar/laborator      | Șef lucr. dr. ing. Roșca Cosmina-Mihaela          |
| 2.4. Titularul activității proiect                  | -   |
| 2.5. Anul de studiu                                 | 2   |
| 2.6. Semestrul *                                    | 3   |
| 2.7. Tipul de evaluare                              | Examen  |
| 2.8. Categoria formativă** / regimul*** disciplinei | DS/DOB  |

\* numărul semestrului este conform planului de învățământ;

\*\* DF - Discipline fundamentale; DS - discipline de specializare; DC - discipline complementare

\*\*\* obligatorie/impusă = DOB; opțională = DOP; facultativă = DFA

## 3. Timpul total estimat (ore pe semestru al activităților didactice)

|  |    |                     |    |                        |    |              |     |
|--|----|---------------------|----|------------------------|----|--------------|-----|
| 3.1. Număr de ore pe săptămână   | 4  | din care: 3.2. curs | 2  | 3.3. Seminar/laborator | 2  | 3.4. Proiect | -   |
| 3.5. Total ore din planul de învățământ  | 56 | din care: 3.6. curs | 28 | 3.7. Seminar/laborator | 28 | 3.8. Proiect | -   |
| 3.9. Total ore studiu individual (studiu după suport de curs, bibliografie și notițe, documentare suplimentară în bibliotecă, pe platformele electronice de specialitate, pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri) |    |                     |    |                        |    |              | 69  |
| 3.10. Total ore pe semestru  |    |                     |    |                        |    |              | 125 |
| 3.11. Numărul de credite   |    |                     |    |                        |    |              | 5   |

## 4. Condiții (acolo unde este cazul)

|   |   |
|---|---|
| 4.1. de curriculum                              | ➤ Programarea calculatoarelor și limbaje de programare; Proiectarea algoritmilor.   |
| 4.2. de desfășurare a cursului                  | ➤ Sală de curs multimedia, acces la platforme educaționale online.                  |
| 4.3. de desfășurare a seminarului/laboratorului | ➤ Sisteme de calcul dotate cu Visual Studio, Git, GitHub, Jira, conexiune Internet. |

## 5. Competențe specifice acumulate și rezultatele învățării\* care stau la baza acestora

| Competențe profesionale  | Rezultatele învățării*  |
|--|---|
| C3. Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor | C1 - Studentul descrie, identifică, sumarizează concepte și metode elementare privitoare la limbaje de programare, medii de programare, tehnici de programare, inteligență artificială și inginerie software și modul lor de aplicare în probleme concrete.<br>A1 - Studentul alege și explică concepte proprii specifice proiectării algoritmilor. |

|                                |  |
|--------------------------------|--|
|                                | RA1 - Studentul are o comportare onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei.<br>RA2 - Studentul selectează și utilizează surse bibliografice specifice domeniului. |
| <b>Competențe transversale</b> | <b>Rezultatele învățării*</b>  |
| N/A                            | -  |

\* C – cunoștințe; A – aptitudini; RA – responsabilitate și autonomie.

## 6. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

|  |  |
|--|--|
| 6.1. Obiectivul general al disciplinei | <ul style="list-style-type: none"> <li>➤ Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor prin cunoașterea etapelor necesare dezvoltării aplicațiilor complexe, alegerea tehnologiilor potrivite tipului de aplicație și utilizarea instrumentelor specifice lucrului în echipă în domeniul dezvoltării de conținut software.</li> </ul>   |
| 6.2. Obiectivele specifice             | <ul style="list-style-type: none"> <li>➤ Etapizarea dezvoltării unor aplicații complexe;</li> <li>➤ Proiectarea și implementarea de programe folosind tehnologii moderne;</li> <li>➤ Scrierea de programe capabile să se conecteze la servicii moderne prin intermediul API-urilor, incluzând API-uri ale LLM-urilor;</li> <li>➤ Utilizarea instrumentelor de testare a programelor;</li> <li>➤ Planificarea dezvoltării programelor cu ajutorul instrumentelor specifice;</li> <li>➤ Utilizarea instrumentelor de partajare a codului;</li> <li>➤ Realizarea de aplicații hibride (clasice și AI).</li> <li>➤ Analiza aspectelor etice și de securitate ale instrumentelor ML;</li> <li>➤ Dezvoltarea abilităților de lucru în echipă.</li> </ul> |

## 7. Conținuturi

| 7.1. Curs  | Nr.ore | Metode de predare   | Observații  |
|--|--------|---|---|
| 1.Instrumente clasice pentru dezvoltarea programelor<br>1.1. Instrumente CASE.<br>1.2. Limbaje de programare și medii de dezvoltare.<br>1.3. Framework-uri pentru dezvoltare conținut software și modele ML.<br>1.4. Template-uri.<br>1.5. Instrumente de interogare integrate (LINQ). | 3 ore  | Expunere, studii de caz, conversație, demonstrații, dezbateri, utilizare tehnologii multimedia                          |   |
| 2. Funcții asistate de AI în dezvoltarea programelor<br>2.1. Utilizarea modelelor LLM pentru asistarea în generarea de cod.<br>2.2. Generatoare de cod AI.   | 3 ore  |   |   |
| 3. Instrumente și API-uri AI/ML.<br>3.1. Prezentarea generală a API-urilor.<br>3.2. Explorarea celor mai utilizate API-uri AI/ML.<br>3.3. Etică și securitate în programare asistată de AI.  | 6 ore  | Expunere, studii de caz, conversație, demonstrații, evaluare bazată pe portofoliu digital                               | Fiecare student va avea ca temă integrarea într-o aplicație a unui API, la alegere, care va fi evaluat de către profesor                            |
| 4. Instrumente pentru testarea programelor (inclusiv modele ML).<br>4.1. Terminologie specifică unităților de testare.<br>4.2. Mecanisme pentru tratarea erorilor.<br>4.3. Instrumente pentru testarea unitară.<br>4.4. Analiza comparativă între instrumente clasice vs. AI-native.   | 6 ore  | Expunere, conversație, metoda "Flipped Classroom" (resurse video, MOOC-uri și materiale OER), demonstrații și dezbateri | Studentii aleg câte un instrument, pregătesc avantajele acasă și le prezintă în clasă. Colegii oferă feedback structurat folosind rubrici digitale. |

|   |                |  |  |
|---|----------------|--|--|
| 4.5. Instrumente pentru testarea automată a programelor și a pipeline-urilor ML.  |                |  |  |
| 5. Instrumente pentru planificarea dezvoltării programelor<br>5.1. Modele de dezvoltare software.<br>5.2. Clasificare instrumente pentru planificarea dezvoltării programelor și analiză comparativă.<br>5.3. Studiu de caz: Jira.  | 4 ore          | Design Thinking și Platforma digitală de colaborare (Jira)   | Se utilizează un panou Jira pentru gestionarea sarcinilor.   |
| 6. Instrumente pentru partajarea codului<br>6.1. Demonstrații Git și GitHub.<br>6.2. Funcții asistate AI pentru partajarea codului.<br>6.3. Bune practici pentru colaborarea asistată de AI   | 4 ore          | Livestream pe Google Meet cu Q&A interactiv  | Un expert invitat din industrie face o prezentare online privind utilizarea unei combinații de instrumente pe care le folosește în desfășurarea activității sale |
| Recapitulare.   | 2 ore          | Expunere   |  |
| <p><b>Bibliografie</b></p> <ol style="list-style-type: none"> <li>1. <b>Roșca, C.M., Instrumente pentru dezvoltarea programelor</b>, Ed. UPG, Ploiești, 2022.</li> <li>2. <b>Rosca, C.-M.</b>; Stancu, A.; Iovanovici, E.M. The New Paradigm of Deepfake Detection at the Text Level. Appl. Sci. 2025, 15, 2560. <a href="https://doi.org/10.3390/app15052560">https://doi.org/10.3390/app15052560</a></li> <li>3. <b>Roșca, C.M., Instrumente pentru dezvoltarea programelor folosind mediul Visual Studio</b>, Ed. UPG, Ploiești, 2022.</li> <li>4. <b>Rosca, C.-M.</b>; Stancu, A. Fusing Machine Learning and AI to Create a Framework for Employee Well-Being in the Era of Industry 5.0. Appl. Sci. 2024, 14, 10835. <a href="https://doi.org/10.3390/app142310835">https://doi.org/10.3390/app142310835</a></li> <li>5. Karatas, M., Developing AI Applications: An Introduction, Rheinwerk Computing, 2024</li> <li>6. Hinton, A., AI and ML for Coders: A Comprehensive Guide to Artificial Intelligence and Machine Learning Techniques, Tools, Real-World Applications, and Ethical Considerations for Modern Programmers (Kindle edition). AI Fundamentals, 2024</li> </ol> |                |  |  |
| <b>7.2. Seminar / laborator</b>   | <b>Nr. ore</b> | <b>Metode de predare</b>   | <b>Observații</b>  |
| Noțiuni de bază în elaborarea unui program și utilizarea AI IntelliCode   | 2              | Interactivă și convențională, centrată pe student. Aplicații practice și teme pentru verificarea gradului de înțelegere privind noțiunile studiate folosind platforma Moodle |  |
| Sistemul de fișiere în C#. Șiruri de caractere în C#. Clase în C#.  | 4              |  |  |
| Instrumentul LINQ   | 2              |  |  |
| Utilizarea API-uri și integrare cu instrumente de AI  | 6              |  |  |
| LINQ, Windows Presentation Foundation, Entity Framework și instrumentul ML.NET  | 4              |  |  |
| Instrumente pentru testarea unitară (MSTest)  | 4              |  |  |
| Instrumente pentru planificarea dezvoltării programelor (Jira)  | 2              |  |  |
| Instrumente pentru partajarea codului (Git și GitHub)   | 2              |  |  |
| Colocviu  | 2              |  |  |
| <p><b>Bibliografie</b></p> <ol style="list-style-type: none"> <li>1. <b>Roșca, C.M., Instrumente pentru dezvoltarea programelor folosind mediul Visual Studio</b>, Ed. UPG, Ploiești, 2022.</li> <li>2. <b>Roșca, C.M.</b>, Rădulescu G., Instrumente software pentru studiul sistemelor de operare avansate, Editura Universității Petrol-Gaze din Ploiești, 2021, ISBN 978-973-719-835-8, 283 pag</li> <li>3. Hinton, A., AI and ML for Coders: A Comprehensive Guide to Artificial Intelligence and Machine Learning Techniques, Tools, Real-World Applications, and Ethical Considerations for Modern Programmers (Kindle edition). AI Fundamentals, 2024</li> </ol>  |                |  |  |

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

- Competențele dobândite de studenți în urma participării la acest curs le vor asigura acestora capacitatea de a realiza aplicații informatice complexe și de a se integra cu ușurință în echipele de dezvoltare software care utilizează instrumente specifice colaborării. Prin înțelegerea și utilizarea instrumentelor CASE, studenții vor fi capabili să proiecteze și să dezvolte soluții software avansate, beneficiind de funcționalități precum modelarea arhitecturii și generarea automată a codului prin instrumente AI, înțelegând implicațiile utilizării acestora.
- Cunoașterea și utilizarea instrumentelor de interogare integrate vor oferi studenților abilitatea de a lucra cu surse de date variate. Utilizarea instrumentelor AI le va permite să se integreze în contextul modern al dezvoltărilor de software.
- Învățarea și utilizarea instrumentelor de testare a programelor și a instrumentelor AI reprezintă competențe necesare viitorilor dezvoltatori software pentru a se adapta cerințelor de piață.
- Prin familiarizarea cu instrumentele pentru planificarea dezvoltării programelor, cum ar fi modelele de dezvoltare software și instrumentele de gestionare a proiectelor precum Jira, studenții vor fi pregătiți să planifice și să organizeze sarcinile și resursele în cadrul unui proiect de dezvoltare software, respectând termenele limită ale proiectului.
- Prin învățarea și utilizarea instrumentelor pentru partajarea codului, cum ar fi Git, studenții vor putea lucra în mod colaborativ, partajând și gestionând codul sursă, asigurând versionarea codului.

## 9. Evaluare

| Tip activitate  | 9.1. Criterii de evaluare | 9.2. Metode de evaluare               | 9.3. Pondere din nota finală |
|---|---------------------------|---------------------------------------|------------------------------|
| 9.4. Curs   | Examinare finală          | Subiecte teoretice grilă și aplicații | 60%                          |
|   | Activitate curs           | Teme                                  | 10%                          |
| 9.5. Seminar/laborator  | Activitate de laborator   | Aplicații                             | 30%                          |
| 9.7. Standard minim de performanță  |                           |                                       |                              |
| ➤ Însușirea unor concepte de bază ale disciplinei instrumente pentru dezvoltarea programelor. |                           |                                       |                              |

Data completării

25.09.2025

Semnătura titularului de curs

Șef lucr. dr. ing.  
Roșca Cosmina-Mihaela

Semnătura titularului de seminar/laborator

Șef lucr. dr. ing.  
Roșca Cosmina-Mihaela

Semnătura titularului de proiect

\_\_\_\_\_

Data avizării în departament

26.09.2025

Director de departament  
Conf. dr. ing. Pricop Emil

\_\_\_\_\_

Decan

Conf. dr. ing. Bădicioiu Marius

\_\_\_\_\_