

ASPECTE PRIVIND ÎNVĂȚAREA PROGRAMĂRII FOLOSIND INTELIGENȚA ARTIFICIALĂ

Teodora VASCAN, dr., conf. univ.

<https://orcid.org/0000-0002-6828-5343>

Catedra ITI, UPSC

Rezumat. În articol se investighează rolul crucial pe care Inteligența Artificială îl are în procesul de învățare a programării. Studiul examinează multiple aplicații digitale bazate pe Inteligența Artificială, evaluând eficacitatea și relevanța acestora pentru cei care doresc să învețe programarea. Sunt prezentate și selectate aplicații recomandate, evidențiind modul în care IA poate personaliza și optimiza experiența de învățare, facilitând înțelegerea conceptelor complexe și dezvoltarea abilităților de programare.

Cuvinte cheie: AI, DeepCode, OpenAI Codex, Tabnine, GitHub Copilot, Codiga AI, Study Buddy.

ASPECTS OF LEARNING PROGRAMMING USING ARTIFICIAL INTELLIGENCE

Abstract. The article investigates the crucial role that Artificial Intelligence plays in the learning process of programming. The study examines multiple digital applications based on Artificial Intelligence, evaluating their effectiveness and relevance for those who want to learn programming. Recommended applications are presented and selected, highlighting how AI can personalize and optimize the learning experience, facilitating the understanding of complex concepts and the development of programming skills.

Keywords: AI, DeepCode, OpenAI Codex, Tabnine, GitHub Copilot, Codiga AI, Study Buddy.

Introducere

Într-o lume în continuă evoluție, unde se vede prezentă Inteligența Artificială în toate sferile activității umane, educația nu este o excepție. Inteligența Artificială devine din ce în ce mai esențială în procesul educațional, în special pentru învățarea programării. Utilizarea Inteligenței Artificiale în educație oferă oportunități unice pentru a revoluționa modul de învățare, în special pentru studenții care vor să învețe programarea, făcând procesul de învățare mai accesibil, interactiv și personalizat [1]. În domeniul educației, inteligența artificială și programarea se completează perfect. Algoritmii cu Inteligență Artificială pot evalua modelele individuale de învățare, punctele forte și punctele slabe, deschizând calea pentru experiențe de învățare personalizate pentru dezvoltatorii de programe începători. Această abordare personalizată nu numai că crește implicarea, ci și accelerează învățarea.

Una dintre principalele avantaje ale Inteligenței Artificiale în învățarea programării este capacitatea de a oferi feedback instantaneu. Prin evaluarea codului scris de studenți în timp real, Inteligența Artificială poate identifica erori, sugera și oferi explicații detaliate. Acest feedback rapid ajută la corectarea greșelilor mai repede, consolidând astfel învățarea și încurajând dezvoltarea abilităților de programare.

Inteligența Artificială poate personaliza experiența de învățare pentru fiecare student, adaptându-se la nevoile și ritmul specific. Prin analizarea progresului și a dificultăților întâmpinate, Inteligența Artificială poate recomanda resurse suplimentare, exerciții personalizate și provocări specifice care să stimuleze și să dezvolte competențe individuale. De asemenea, Inteligența Artificială poate oferi simulări și medii de programare interactive, care permit studenților să experimenteze și să învețe prin practică. Platformele bazate pe AI pot crea scenarii realiste și proiecte de programare care vor ajuta la aplicarea teoriei în contexte practice, pregătindu-i pe studenți pentru provocările din lumea reală. Un alt avantaj al Inteligenței Artificiale în procesul educațional este posibilitatea de a facilita învățarea colaborativă, conectând studenții cu mentorii, cu colegii și experții în domeniu. Prin platformele online și comunitățile virtuale, studenții pot lucra împreună, pot împărtăși cunoștințe și pot primi îndrumare, ceea ce contribuie la un mediu de învățare mai dinamic și mai interactiv.

În general, integrarea Inteligenței Artificiale în procesul de învățare a programării aduce semnificative beneficii. De la feedback instantaneu și personalizare, până la simulări interactive și colaborare online, Inteligența Artificială transformă modul în care învățăm programarea, pregătindu-ne pentru un viitor digital în continuă schimbare.

Rolul Inteligenței Artificiale în studierea programării

Un rol important în studierea programării îl au *platformele de învățare alimentate de Inteligență Artificială*. Apariția platformelor de învățare bazate pe Inteligență Artificială a schimbat jocul în învățarea programării. Aceste platforme folosesc algoritmi de Inteligență Artificială pentru a oferi cursuri personalizate bazate pe mediile și obiectivele studenților. Analizând cantități uriașe de date, Inteligența Artificială poate recomanda cele mai potrivite cursuri, proiecte și resurse pentru fiecare persoană. În plus, platformele de învățare bazate pe Inteligență Artificială se adaptează în mod constant la progresul studenților și fac ajustări în timp real pentru a asigura o drum optim de învățare. Această abordare dinamică îi ajută pe studenți să rămână motivați, încrezători și concentrați pe măsură ce își dezvoltă și formează abilitățile de programare.

Un alt aspect important pe care îl are Inteligența Artificială în studierea programării este *ajutorul oferit privind scrierea codului*. Scrierea codului este o parte integrantă a programării, iar Inteligența Artificială a îmbunătățit foarte mult acest proces cu asistență inteligentă pentru codare. Editorii de cod bazați pe Inteligență Artificială și mediile de dezvoltare integrate (IDE) analizează codul care este scris și oferă sugestii utile, funcții automate și detectarea erorilor în timp real. Beneficiile Inteligenței Artificiale în acest sens depășesc eficiența și acuratețea. Aceste instrumente pot ajuta dezvoltatorii, în special începătorii, să învețe cele mai bune practici, standarde de cod și să își optimizeze codul

pentru performanță. Datorită acestui fapt, dezvoltatorii pot petrece mai puțin timp depanând și mai mult timp creând soluții inovatoare.

Inteligența Artificială a făcut, de asemenea, progrese semnificative în *automatizarea proceselor de testare și depanare*. Folosind algoritmi de învățare automată, Inteligența artificială poate detecta potențialele erori, vulnerabilități și erori în cod cu o acuratețe remarcabilă. Testarea automată economisește timp și resurse, dar asigură și livrarea de software de înaltă calitate. În plus, îmbunătățirea continuă a instrumentelor de depanare bazate pe Inteligența Artificială promite să reducă semnificativ timpul de dezvoltare și să facă programarea mai eficientă.

Într-un sistem de învățământ tradițional, studenții urmează adesea un curriculum specific, indiferent de cunoștințele sau interesele existente. Cu toate acestea, Inteligența Artificială are potențialul de a schimba acest peisaj. Analizând performanța studenților, preferințele și obiectivele, Inteligența Artificială poate *crea căi de învățare individualizate*. Personalizarea căii de învățare bazată pe Inteligența Artificială îi ajută pe studenți să se concentreze asupra zonelor lor slabe și oferă provocări care se potrivesc cu nivelul lor de calificare. Această adaptabilitate asigură că studenții nu au nici oboseală, nici plictiseală, ceea ce promovează creșterea continuă a abilităților lor de programare.

Un alt aspect interesant al impactului Inteligenței Artificiale în studierea programării este *integrarea procesării limbajului natural* (NLP). NLP permite studenților să interacționeze cu codul și conceptele de programare folosind limbajul natural, făcând învățarea mai intuitivă și mai ușor de utilizat. NLP poate fi folosit pentru a transforma documentația de programare și tutorialele într-un limbaj ușor de utilizat, de care pot beneficia studenții de toate nivelurile. Această abordare incluzivă a învățării codificării permite oamenilor din toate mediile să exploreze lumea codificării fără bariere inutile.

Jocurile sunt recunoscute de mult timp ca un mijloc eficient de implicare și motivare. Combinarea Inteligenței Artificiale cu platformele de învățare bazate pe jocuri duce experiența la un nivel cu totul nou. Algoritmii Inteligenței Artificiale pot evalua progresul, preferințele și stilul de învățare ale unei persoane și pot adapta experiențe de joc care provoacă și inspiră studenții. Învățarea variată folosește puterea competiției, recompenselor și realizărilor pentru a încuraja învățarea și îmbunătățirea continuă. Făcând procesul de învățare distractiv și interactiv, jocurile îmbunătățite cu Inteligența Artificială implică studenții și îi face să își urmărească obiectivele programului cu entuziasm.

Pe măsură ce valorificăm potențialul Inteligenței Artificiale în studierea programării, este important să luăm în considerare aspectele etice. Corectitudinea, transparența și protecția datelor trebuie să fie prioritare atunci când se utilizează algoritmi de Inteligență Artificială în luarea deciziilor educaționale. În plus, este important să ne asigurăm că Inteligența Artificială nu conține părtiniri și că datele studenților sunt gestionate în mod responsabil. Posibilități viitoare ale Inteligenței Artificiale în studierea programării sunt

nelimitate. Pe măsură ce tehnologia Inteligenței Artificiale avansează, oportunitățile de inovare perturbatoare devin din ce în ce mai clare.

Aplicații cu Inteligență Artificială recomandate pentru studierea programării

Tehnologiile moderne de Inteligență Artificială, cum ar fi învățarea automată și rețelele neuronale, au avut un impact revoluționar asupra procesului de învățare a programării. Învățarea automată permite computerelor să extragă cunoștințe din date fără a programa în mod explicit fiecare pas. Rețelele neuronale pot detecta modele și pot efectua diverse sarcini, cum ar fi traducerea textului sau recunoașterea imaginilor.

Un exemplu excelent de astfel de tehnologie este rețeaua neuronală Chat GPT a OpenAI, care este capabilă să analizeze și să genereze codul de program, oferind dezvoltatorilor posibilitatea de a-și automatiza și accelera munca.

Printre cele mai remarcabile inovații din acest sector sunt două proiecte majore: *DeepCode* și *OpenAI Codex*.

DeepCode, care a început ca startup în Zurich, este un exemplu de utilizare eficientă a învățării automate pentru analiza și optimizarea codului de program. Acest instrument, similar cu un examinator inteligent de cod, scanează codul pentru a identifica erorile și vulnerabilitățile și practicile de programare ineficiente. Alimentat de o bază de date extinsă de coduri colectate din surse deschise, cum ar fi *GitHub*, *DeepCode* învață din milioane de proiecte pentru a deveni un instrument eficient care oferă ajustări utile pentru optimizarea codului.

Al doilea instrument, *OpenAI Codex*, dezvoltat de *OpenAI*, este un sistem puternic de Inteligență Artificială capabil să genereze cod din limbajul natural. Fiind o versiune avansată a modelului de limbaj *GPT-3*, *Codex* impresionează prin capacitatea sa de a înțelege descrierile textuale ale sarcinilor și de a le transforma în cod de program de lucru.

Acest lucru deschide oportunități uimitoare nu numai pentru dezvoltatorii experimentați, care pot accelera procesul de codificare, ci și pentru începători, pe care *Codex* îi ajută chiar și în procesul de învățare a programării [2].

La rândul lor, specialiști de top în tehnologia informației de la United Airlines, Johnson & Johnson, Visa, Cardinal Health și Goldman Sachs sunt încântați de potențialul Inteligenței Artificiale generative în contextul automatizării procesului de codificare. Ei îl văd ca pe un instrument care poate crește semnificativ productivitatea. Cu toate acestea, aceștia își exprimă îngrijorarea că adoptarea Inteligenței Artificiale ar putea duce la scăderea barierelor la intrare, la niveluri crescute de complexitate, la datoria tehnică și la confuzie generală. În acest sens, ei vor trebui să rezolve problemele asociate cu gestionarea acestui balon software „umflat”.

„Problema creșterii datoriei tehnice și a codului mort va fi întotdeauna o problemă”, a declarat Tracey Daniels, director de date la compania de servicii financiare Truist. „Există

riscul să fim inundați cu un volum mare de cod defect generat de mașini”, adaugă Armando Solar-Lezama, profesor la Laboratorul de Informatică și Inteligență Artificială al MIT.

Adevărat, nu toată lumea este de acord cu un scenariu atât de pesimist. Cel puțin, dacă te uiți la capacitățile actuale de generare automată a codului în *Copilot* și *ChatGPT*, acesta are încă o mulțime de erori și nu poate fi complet de încredere.

De exemplu, conform statisticilor *Github*, utilizatorii acceptă aproximativ 30% din codul oferit de *Copilot*. Interesant este că rata de adoptare crește odată cu timpul de utilizare a acestui instrument [3].

Generarea automată a codului este un instrument puternic care automatizează scrierea codului în diferite limbaje de programare, cum ar fi *JavaScript*, *Python*, *SQL*, *JSON*, *HTML* și altele. Acest instrument va fi util pentru dezvoltatorii care nu sunt familiarizați cu un anumit limbaj de programare sau care au nevoie de asistență atunci când creează cod standard. Codul generat este ușor de modificat și adaptat la cerințele specifice ale utilizatorului.

Pentru a utiliza funcțiile de generare automată a codului în diferite domenii, inclusiv proprietățile componentelor, editorii de cod și configurarea pașilor, se va introduce un indiciu sau o descriere a codului de care aveți nevoie și codul va fi generat automat. În plus, în timpul procesului de generare, Inteligența Artificială oferă posibilitatea de a verifica și edita codul generat înainte de a-l introduce. Această abordare oferă flexibilitatea de a face orice modificări de cod necesare pentru a se potrivi cerințelor unice ale proiectelor [4].

Pentru a facilita generarea apelurilor API, se furnizează instrucțiuni sau o descriere a acțiunii pe care se dorește să o efectueze aplicația. De exemplu, dacă trebuie de creat o interogare în *Salesforce* pentru a obține o listă de oportunități de venituri, va fi descrisă în interogare. Inteligența Artificială înțelege în mod independent cererea și generează automat apelul corect *Salesforce* cu punctul final, parametrii, anteturile și corpul cererii corespunzătoare [5].

Se recomandă de a profita de caracteristica de aspect pentru a genera date unice de aspect pentru un proces de dezvoltare API eficient și fără probleme. Caracteristica *Editare* permite de a modifica codul existent prin evidențierea acestuia. Utilizatorii pot vedea o reprezentare încorporată a diferențelor în previzualizarea lor. Această funcționalitate permite de a compara cu ușurință modificările și de a urmări modificările aduse codului [6]. Vom continua cu o descriere a instrumentelor cu Inteligență Artificială recomandate pentru învățarea programării:

Tabnine este un software inovator de dezvoltare a Inteligenței Artificiale conceput pentru a face procesul de codificare mai ușor, mai eficient și mai precis (fig. 1). *Tabnine* oferă o varietate de funcționalități, inclusiv completarea precisă a codului, generarea de funcții, precum și căutarea numelui variabilelor și sugestii de refactorizare.

În plus, *Tabnine* analizează depozitele de cod pentru a oferi idei noi pentru îmbunătățirea codului. Depanatorul încorporat *Tabnine* asigură detectarea rapidă a erorilor în timpul procesului de codare, îmbunătățind productivitatea utilizatorului. Utilizarea algoritmilor de inteligență artificială combinată cu un sistem patentat de recunoaștere a cuvintelor cheie face din *Tabnine* unul dintre cele mai precise instrumente de inteligență artificială din lume.

Prin dezvoltarea continuă a capacităților sale, *Tabnine* continuă să se îmbunătățească, rămânând în fruntea instrumentelor de lucru cu Inteligența Artificială. Permite utilizarea gratuită timp de 30 de zile.

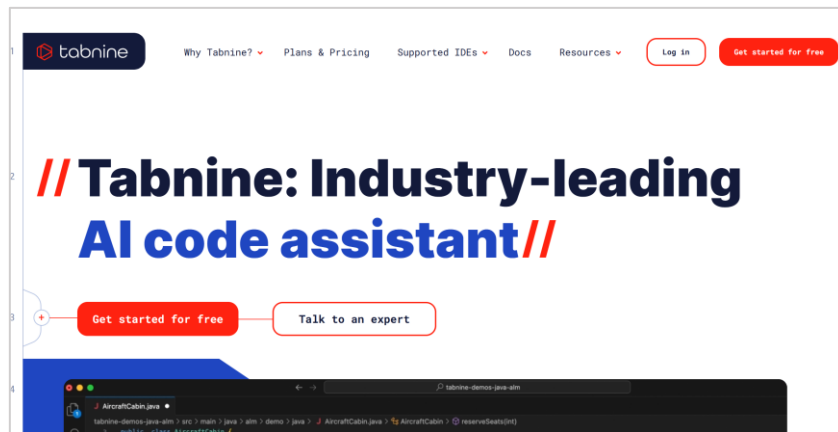


Figura 1. Fereastra aplicației Tabnine

OpenAI Codex (fig. 2) - este un software inovator care oferă un set extins de instrumente pentru crearea inteligenței artificiale. Utilizatorii au acces la o interfață intuitivă, precum și la funcționalități, așa ca procesarea limbajului natural (NLP) și codul sursă deschis (OSC).

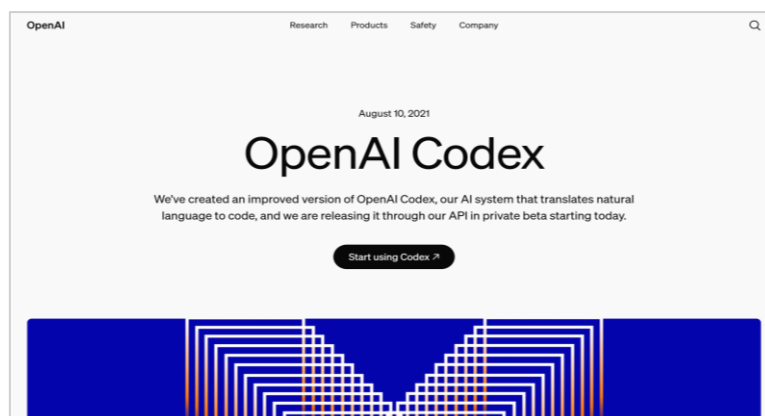


Figura 2. Interfața aplicației OpenAI Codex

În plus, *OpenAI Codex* oferă instrumente de depanare vizuale puternice care permit dezvoltatorilor să verifice rapid logica internă a codului lor. Acest lucru accelerează și simplifică semnificativ procesele de dezvoltare și asigură, de asemenea, o colaborare eficientă între programatori pe proiecte comune.

OpenAI Codex încurajează crearea de proiecte cu adevărat inovatoare, reducând la minimum necesitatea unei codări complexe. Platforma facilitează, de asemenea, programarea prin promovarea unei mai bune înțelegeri a conceptelor mai profunde prin reprezentări vizuale ale componentelor complexe.

Un instrument esențial pentru fiecare profesionist în inteligență artificială, *OpenAI Codex* ajută îmbunătățirea proiectelor făcând programarea mai accesibilă și mai intuitivă.

GitHub Copilot (fig. 3) este un instrument inovator care oferă soluții pentru scrierea codului în diferite limbaje de programare bazate pe introducerea în limbaj natural. Antrenat pe un volum uriaș de cod, *GitHub Copilot* este însoțitorul ideal pentru dezvoltatorii de toate nivelurile, permițând construirea și implementarea rapidă de aplicații fără griji cu privire la erorile de sintaxă sau problemele de compilare.

Acest instrument este excelent și pentru depanarea codului, oferind feedback în timp real asupra potențialelor probleme. Integrarea excepțională cu fluxul de lucru actual face *GitHub Copilot* convenabil și eficient, accelerând procesul de codare.

Ceea ce face ca *GitHub Copilot* să fie special este rolul său de „Programator al perechii tale”. În ciuda acestui fapt, rularea instrumentului va necesita utilizarea unui software plătit, cum ar fi Microsoft Visual Studio Code Editor [7].

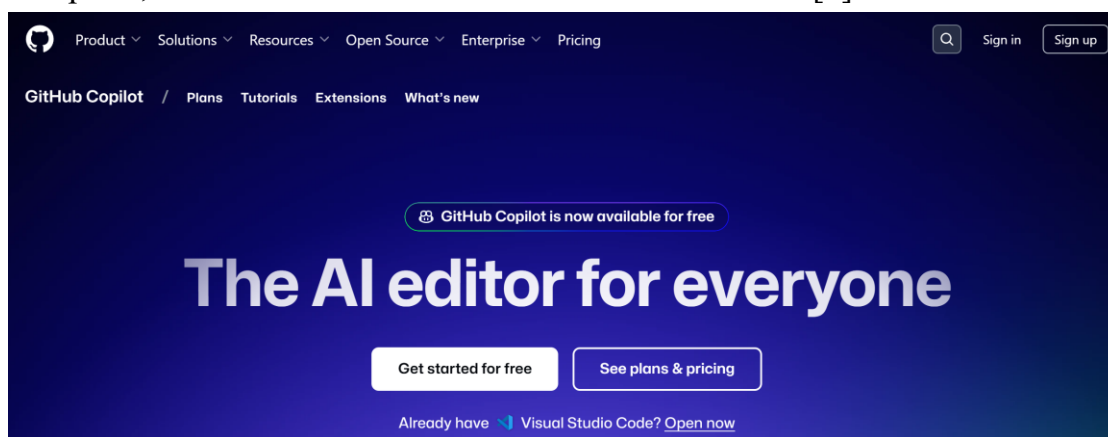


Figura 3. Fereastra aplicației GitHub Copilot

Codiga AI (fig. 4) - este un software inovator de codare care oferă instrumente ușor de utilizat pentru dezvoltarea, depanarea și optimizarea codului. Revizuirea automată a codului *Codiga* permite utilizatorilor să verifice acuratețea codului lor înainte de a-l trimite. Cu tehnologia cu Inteligență Artificială avansată *Codiga*, dezvoltatorii își pot testa rapid codul pentru erori și își pot îmbunătăți performanța.

Software-ul vine cu un mediu de dezvoltare integrat (IDE) avansat, care compilează funcții complexe în diferite limbaje de programare, cum ar fi C++, Java, Python, fără a fi nevoie de plugin-uri sau extensii suplimentare. În plus, varietatea de instrumente interactive de depanare *Codiga AI* permite dezvoltatorilor de la toate nivelurile de experiență să identifice și să rezolve rapid problemele de programare.

Extrem de personalizabil și capabil să se integreze cu fluxul de lucru actual, *Codiga* a devenit o platformă de încredere și versatilă pentru crearea de soluții de programare remarcabile, cu o viteză și flexibilitate incredibile.

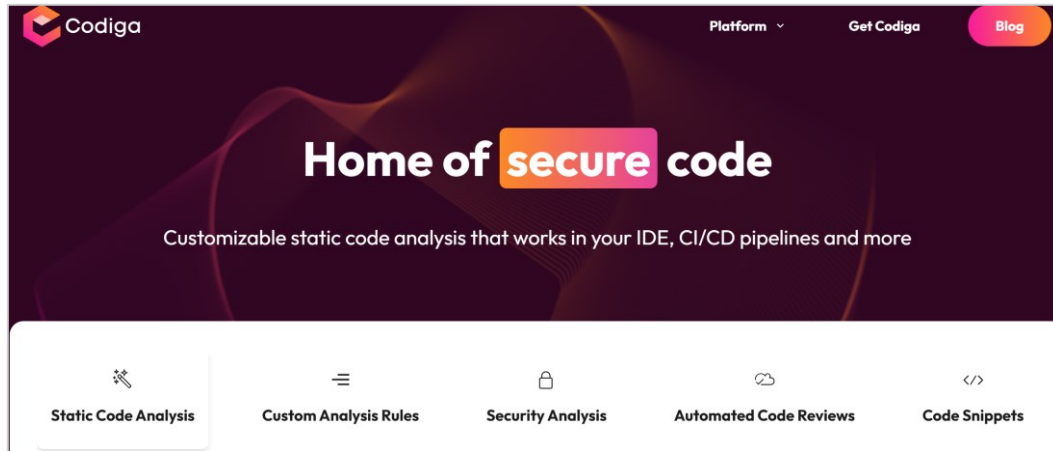


Figura 4. Interfața aplicației Codiga

Study Buddy (fig. 5) este un partener indispensabil în lumea programării pentru dezvoltarea aplicațiilor Android. Această colaborare inovatoare în cadrul Android Studio își propune să sporească productivitatea, răspunzând rapid la solicitările dezvoltatorilor Android. Cu ajutorul Inteligenței Artificiale, *Study Buddy* este capabil să înțeleagă limbajul natural, permițând exprimarea întrebărilor într-o engleză simplă. *Study Buddy* sprijină dezvoltatorii Android în generarea de cod, găsirea de resurse relevante, învățarea celor mai bune practici și economisirea timpului [8].

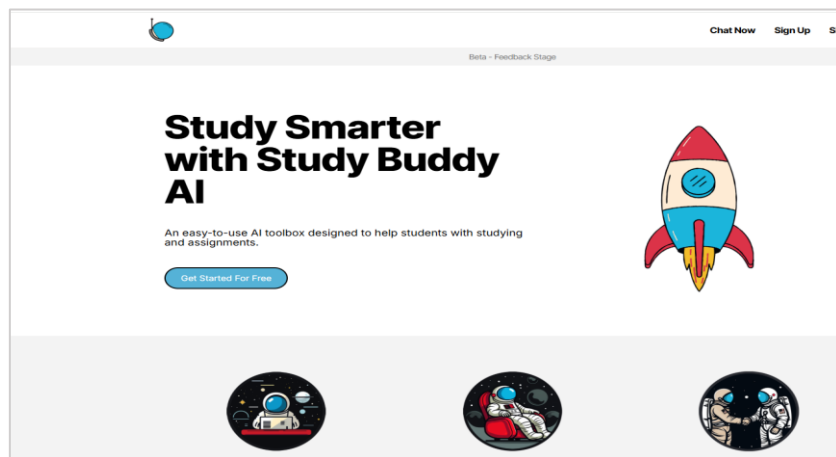


Figura 5. Interfața aplicației Study Buddy

Generalizând cele relatate mai sus putem spune că rețelele neuronale pot simplifica semnificativ viața unui începător programator sau dezvoltator obișnuit prin următoarele oportunități oferite:

- ✓ *Descompunerea sarcinilor.* Inteligența artificială este capabilă să împartă sarcini complexe în sarcini mici, să le explice, să ofere sfaturi despre cum să le finalizeze și să sugereze soluția optimă.

- ✓ *Generarea codului.* Un model inteligent poate primi o sarcină de programare specifică ca intrare, iar rezultatul va fi o bucată de cod gata făcută. Inteligența Artificială este, de asemenea, capabilă să analizeze codul altor persoane, să-l explice, să adauge comentarii, să cerceteze tehnologii de programare etc.
- ✓ *Revizuirea codului și refactorizarea.* Se poate economisi timp având încredere în Inteligența Artificială pentru a face modificările necesare: scanarea și optimizarea codului, eliminarea fragmentelor redundante, îmbunătățirea conciziei, a algoritmilor etc.
- ✓ *Teste de scriere.* Dezvoltatorilor nu prea le place să facă teste unitare, deși este o parte importantă a asigurării calității software-ului. Din fericire, inteligența artificială poate genera cazurile de testare necesare acoperind diverse scenarii și, de asemenea, poate automatiza execuția acestor teste.
- ✓ *Tratarea pro activă a erorilor.* Inteligența Artificială prezice erori potențiale și recomandă remedieri înainte ca acestea să apară, reducând timpul de depanare a codului.
- ✓ *Analiză puternică a datelor.* Asistenții artificiali sunt capabili să analizeze cantități mari de date relativ rapid pentru a identifica relațiile cauză-efect, pentru a identifica modele sau probleme potențiale, erori și altele asemenea. Destul de relevant pentru specialiștii IT care țin de Data Science, Machine Learning, Big Data etc.
- ✓ *Crearea documentației.* Inteligența Artificială poate analiza codul și genera automat documentație cuprinzătoare care explică funcționalitatea, instrucțiunile de utilizare și linkurile API și multe altele.
- ✓ *Prognoza.* Predicția și modelarea Inteligența Artificială a comportamentului utilizatorilor și a performanței software le permite dezvoltatorilor să planifice scalabilitate, să identifice potențiale blocaje la începutul dezvoltării și să le abordeze în mod proactiv pentru a asigura operațiuni bune în viitor.

Lista poate fi extinsă cu multe alte articole, în funcție de specificul direcției de dezvoltare. Folosind rețelele neuronale, dezvoltatorii își pot crește semnificativ eficiența și productivitatea. Cu toate acestea, este important să ne amintim că AI este un facilitator astăzi, iar expertiza umană rămâne esențială pentru luarea unor decizii importante, rezolvarea problemelor în mod creativ și asigurarea calității și viziunii generale a software-ului.

Concluzii

Utilizarea Inteligenței Artificiale pentru a învăța programarea este o experiență transformatoare în viitorul educației. Cu platforme de învățare bazate pe inteligență artificială, asistență pentru codificare inteligentă, testare automată și căi de învățare personalizate, dezvoltatorii în devenire sunt echipați cu instrumentele necesare pentru a

reuși în profesia lor. Impactul Inteligenței Artificiale asupra educației în studierea programării depășește eficiența și comoditatea, dând putere studenților, cu diferite nivele de pregătire, să îmbrățișeze lumea codificării și să devină creatori de soluții inovatoare care ne modelează viitorul. Integrarea Inteligenței Artificiale în studierea programării deschide o lume de posibilități și calea pentru o învățare mai incluzivă, dinamică și eficientă.

Articol realizat în cadrul proiectului de cercetări științifice „Metodologia implementării TIC în procesul de studiere a științelor reale din perspectiva conceptului STEAM și Inteligenței Artificiale”, codul 040101, din cadrul Programului instituțional de cercetare (2024-2027), aprobat prin Ordin MEC nr. 102 din 01.02.2024

Bibliografie

1. VASCAN, T. Beneficiile și provocările inteligenței artificiale în educație. In: *Proceedings of The 30nd Conference on Applied and Industrial Mathematics dedicated to the memory of Academician Mitrofan M. Choban – CAIM 2023*, 14th–17th september, 2023, “Alexandru Ioan Cuza” University of Iași, Romania, Editura Pim, ISBN 978-606-13-7848-7, pp.140-146.
2. Cum rescrie inteligența artificială regulile de codare – resursă digitală disponibilă pe: <https://gb.ru/blog/kak-iskusstvennyj-intellekt-perepisyvaet-pravila-napisaniya-koda/>, accesată la data 21.11.2024.
3. Viitorul programării cu AI - primele exemple. - resursă digitală disponibilă pe: <https://habr.com/ru/companies/timeweb/articles/745442/>, accesată la data 22.11.2024.
4. How You Can Use AI to Improve Your Code Quality. - resursă digitală disponibilă pe: <https://www.freecodecamp.org/news/how-to-use-ai-to-improve-code-quality>, accesată la data 23.11.2024.
5. 7 rețele neuronale pentru programatori: cum să scrieți codul mai rapid și mai bine. - resursă digitală disponibilă pe: <https://skillbox.ru/media/code/7-neyrosetey-dlya-programmistov-kak-pisat-kod-bystree-i-luchshe/>, accesată la data 23.11.2024.
6. Writing Code with AI. - resursă digitală disponibilă pe: <https://docs.superblocks.com/generativeai/writing-code-with-ai>, accesată la data 23.12.2024.
7. AI That Writes Code: 7 Best & Popular AI Coding Tools 2024 - resursă digitală disponibilă pe: <https://www.bytracyjackson.com/ai-that-writes-code/>, accesată la data 23.12.2024.
8. Meet Studio Bot - resursă digitală disponibilă pe: <https://developer.android.com/studio/preview/studio-bot>, accesată la data 23.12.2024.