

# REVOLUCIJA U RAZVOJU SOFTVERA: UTICAJ VEŠTAČKE INTELIGENCIJE NA PRODUKTIVNOST I KVALITET PROGRAMIRANJA

Miloš Cetina  
Subotica  
miloscetina@gmail.com

Stručan rad  
UDK : 001(0/004.8):004.4/004.4`2/`6  
<https://doi.org/10.59417/nir.2024.25.90>

## Apstrakt

Razvoj softvera prolazi kroz suštinsku transformaciju zahvaljujući integraciji veštačke inteligencije (AI). Ova teza istražuje kako VI redefiniše softversko inženjerstvo kroz automatizaciju kodiranja, optimizaciju razvoja, poboljšanje testiranja i unapređenje održavanja softverskih sistema. Analizira se uticaj mašinskog učenja, generativnih modela i inteligentnih alata na produktivnost programera, kvalitet softverskih proizvoda i efikasnost razvojnih procesa. Posebna pažnja posvećena je etičkim izazovima, sigurnosnim rizicima i promenama u tradicionalnim ulogama unutar softverske industrije. Na osnovu analize najnovijih tehnologija i studija slučaja, rad zaključuje da VI ne samo da ubrzava razvoj softvera već i postavlja temelje za novu paradigmu u ovoj oblasti.

**Ključne reči:** veštačka inteligencija, razvoj softvera, automatizacija, mašinsko učenje, generativni modeli, produktivnost, testiranje, etički izazovi, sigurnost, softverska industrija.

## UVOD

Veštačka inteligencija (AI) postala je jedan od najvažnijih fenomena savremenog doba, značajno utičući na brojne aspekte ljudskog života, uključujući i oblast razvoja softvera. Razvoj softverskog inženjeringa tokom decenija prolazio je kroz mnoge faze, od ručnog kodiranja do automatizacije procesa, ali dolazak veštačke inteligencije označio je revolucionarni trenutak u ovoj industriji. Sve veća primena AI tehnologija u razvoju softverskih rešenja ne samo da menja način na koji se softver razvija, već i redefiniše ulogu samih programera i njihovih alata. U trenutku kada brzina i kvalitet isporučenog koda postaju ključni faktori u tržišnoj konkurenciji, veštačka inteligencija se nameće kao ključni saveznik u unapređenju produktivnosti i kvaliteta softverskih rešenja.

Razumevanje značaja AI-a u softverskom razvoju počinje od uvida u njenu sposobnost da transformiše procese koji su do sada bili zasnovani isključivo na ljudskoj intervenciji. Veštačka inteligencija, oslanjajući se na tehnike poput mašinskog učenja, dubokog učenja i obrade prirodnog jezika, omogućava automatizaciju zadataka koji su ranije zahtevali značajno ljudsko angažovanje. Ovo uključuje pisanje, testiranje i optimizaciju koda, kao i otkrivanje grešaka i predlaganje rešenja u realnom vremenu. AI alati ne samo da olakšavaju rad programerima već i omogućavaju timovima da se fokusiraju na složenije aspekte razvoja, povećavajući kreativnost i inovativnost. Ova promena ima dubok uticaj na produktivnost, jer se vreme potrebno za realizaciju kompleksnih softverskih projekata značajno smanjuje.

Istovremeno, uloga veštačke inteligencije u poboljšanju kvaliteta softverskih rešenja ne može se zanemariti. Tradicionalni procesi testiranja i verifikacije koda često su bili dugotrajni i podložni greškama usled ljudskog faktora. AI alati, međutim, pružaju mogućnost detaljnije analize koda, prepoznavanja potencijalnih sigurnosnih ranjivosti i predviđanja problema pre nego što oni postanu ozbiljan izazov. Na taj način se ne samo smanjuje broj grešaka u finalnim proizvodima već i unapređuje ukupni kvalitet isporučenog softvera, što ima direktan uticaj na korisničko iskustvo i reputaciju softverskih kompanija.

Razvoj veštačke inteligencije u softverskom inženjeringu nije samo tehnološki fenomen, već i društveni, ekonomski i etički izazov. Pitanja koja se postavljaju u vezi sa sigurnošću i transparentnošću AI sistema, kao i njihovim uticajem na tržište rada, zahtevaju duboko promišljanje. Dok AI alati omogućavaju programerima da postanu produktivniji, postavlja se pitanje kakva će biti njihova uloga u budućnosti, kada se algoritmi razvijaju do tačke gde će moći da preuzmu značajan deo njihovih trenutnih zadataka. Ova dilema otvara prostor za raspravu o tome kako obrazovni sistemi, industrija i društvo mogu da se prilagode novim realnostima, gde će veštačka inteligencija igrati sve centralniju ulogu.

Osim tehničkih i društvenih aspekata, važan segment istraživanja odnosi se i na etiku u razvoju AI alata za programiranje. Automatizacija koda i primena algoritama za generisanje softverskih rešenja postavljaju pitanje odgovornosti za eventualne greške ili zloupotrebe takvih sistema. Dok AI alati omogućavaju generisanje koda u rekordnom vremenu, postavlja se pitanje kvaliteta i pouzdanosti tog koda, posebno kada se koristi u kritičnim aplikacijama, poput finansijskih sistema, zdravstvene industrije ili bezbednosnih sistema. Stoga se sve više pažnje posvećuje razvoju pravila, standarda i regulativa koje bi obezbedile odgovorno korišćenje AI tehnologija u softverskom inženjeringu.

## POČETAK KORIŠĆENJA VEŠTAČKE INTELIGENCIJE

Razvoj softverskog inženjeringa tokom poslednjih decenija bio je obeležen sve većim oslanjanjem na automatizaciju, kao ključni faktor u unapređenju efikasnosti i pouzdanosti procesa razvoja softvera. Uvođenje automatizacije obeležilo je prelazak sa tradicionalnih, manuelnih metoda rada na korišćenje alata i tehnologija koje mogu preuzeti repetitivne i mehaničke zadatke, čime su se softverski inženjeri oslobodili da se fokusiraju na kreativne i složenije aspekte razvoja. Istovremeno, početak primene veštačke inteligencije u softverskom inženjeringu označio je novu eru, u kojoj su mašine počele da igraju aktivnu ulogu u generisanju, testiranju i optimizaciji koda, otvarajući vrata inovacijama koje su do tada bile nezamislive.

Automatizacija u razvoju softvera prvi put je značajnije primenjena kroz alate za upravljanje verzijama i kontrolu izvornog koda, koji su omogućili programerima da efikasnije sarađuju u timovima i izbegnu konflikte prilikom integracije različitih delova koda (Alowais et al., 2023). Sistemi poput Git-a i Subversion-a postali su standard u industriji, obezbeđujući ne samo konzistentnost koda, već i mogućnost praćenja istorije promena i lakog povratka na ranije verzije u slučaju grešaka. Pored toga, alati za automatizaciju kompilacije i izgradnje softvera, kao što su Maven i Gradle, značajno su ubrzali proces pripreme koda za testiranje i implementaciju, smanjujući potencijal za ljudske greške i omogućavajući veću pouzdanost.

Sredinom 2000-ih godina, razvoj softvera ulazi u novu fazu sa širenjem praksi kao što su kontinuirana integracija (CI) i kontinuirana isporuka (CD). Ove prakse, podržane alatima poput Jenkins-a, Travis CI-a i CircleCI-a, omogućile su timovima da automatski integrišu promene u kodu, izvršavaju testove i implementiraju ažuriranja u realnom vremenu. Automatizacija ovih procesa ne samo da je ubrzala cikluse razvoja, već je i značajno smanjila rizik od grešaka koje proizlaze iz manualnog upravljanja timovima i sistemima. Ova faza automatizacije označila je prvi korak ka inteligentnijim sistemima, jer su ovi alati počeli da uključuju osnovne forme analitike i predikcije kako bi unapredili proces donošenja odluka (Ford, 2018).

Početak korišćenja veštačke inteligencije u softverskom inženjeringu obeležen je razvojem alata koji koriste mašinsko učenje za rešavanje specifičnih problema. Prvi značajni primeri uključuju alate za statičku analizu koda, koji koriste algoritme za identifikaciju potencijalnih grešaka i ranjivosti u kodu pre nego što se izvrši njegovo testiranje. Ovi alati, poput SonarQube-a, omogućili su programerima da unaprede kvalitet koda, identifikujući probleme koji bi inače mogli proći nezapaženo tokom manuelne revizije. Dodatno, AI algoritmi su se pokazali korisnim u optimizaciji performansi aplikacija, analizirajući obrasce upotrebe i predlažući prilagođavanja kako bi se postigla optimalna efikasnost.

Jedan od najvažnijih koraka u razvoju veštačke inteligencije u softverskom inženjeringu bilo je uvođenje alata za generisanje koda zasnovanih na mašinskom učenju. Ovi alati, poput GitHub Copilot-a i OpenAI Codex-a, koriste velike jezičke modele trenirane na obimnim skupovima podataka kako bi generisali kod na osnovu tekstualnih zahteva programera. Umesto ručnog pisanja linija koda, programeri sada mogu da opišu funkcionalnosti koje žele, dok AI generiše odgovarajući kod u realnom vremenu (Kaplan & Haenlein, 2020). Ova tehnologija značajno smanjuje vreme potrebno za razvoj aplikacija i omogućava programerima da se fokusiraju na dizajn i arhitekturu sistema, dok repetitivni zadaci postaju automatizovani.

Primena veštačke inteligencije takođe je transformisala proces testiranja softvera, koji je tradicionalno bio jedan od najdugotrajnijih aspekata razvoja. AI alati za automatizaciju testiranja, poput AppliTools i Testim.io, koriste algoritme mašinskog učenja za identifikaciju potencijalnih problema u korisničkom interfejsu, performansama i funkcionalnostima aplikacije. Ovi alati su sposobni da uče iz prethodnih iteracija testiranja i da se prilagođavaju promenama u aplikaciji, omogućavajući dinamičko testiranje koje je znatno efikasnije od tradicionalnih metoda. Pored toga, veštačka inteligencija se koristi za simulaciju korisničkog ponašanja, pružajući uvide u to kako će se aplikacija ponašati u stvarnom svetu, što dodatno unapređuje kvalitet i pouzdanost softverskih proizvoda (Kaplan & Haenlein, 2020).

Početak korišćenja veštačke inteligencije u razvoju softvera ne donosi samo tehnička unapređenja, već postavlja i nova pitanja i izazove. Jedno od ključnih pitanja odnosi se na etiku i odgovornost za kod generisan od strane AI sistema. Dok AI alati mogu generisati funkcionalan kod u rekordnom roku, postavlja se pitanje njegove pouzdanosti, sigurnosti i potencijalnih posledica grešaka u kritičnim aplikacijama. Osim toga, uticaj veštačke inteligencije na ulogu programera izaziva zabrinutost u vezi sa mogućnošću smanjenja potrebe za određenim vrstama poslova u ovoj oblasti.

Uvođenje automatizacije i veštačke inteligencije u softverski inženjering promenilo je način na koji se razvijaju, testiraju i implementiraju softverski sistemi. Ove tehnologije omogućavaju ne samo veće uštede vremena i resursa, već i značajno povećanje kvaliteta i sigurnosti softverskih rešenja. Iako se industrija još uvek nalazi u fazi prilagođavanja ovim transformacijama, postaje jasno da će automatizacija i veštačka inteligencija igrati sve značajniju ulogu u budućnosti razvoja softvera.

## KONCEPT PRIMENE VEŠTAČKE INTELIGENCIJE U SOFTVERSKOM INŽENJERINGU

Veštačka inteligencija (AI) predstavlja naučnu disciplinu koja se bavi razvojem sistema sposobnih da obavljaju zadatke koji zahtevaju inteligenciju sličnu ljudskoj, poput učenja, zaključivanja,

rešavanja problema i donošenja odluka. U kontekstu softverskog inženjeringa, veštačka inteligencija postaje ključna tehnologija koja transformiše tradicionalne procese razvoja, omogućavajući programerima i timovima da prevaziđu ograničenja manuelnog rada, unaprede produktivnost i poboljšaju kvalitet isporučenih softverskih rešenja. Osnove veštačke inteligencije u ovoj oblasti zasnivaju se na razumevanju njenih osnovnih principa, tehnika i potencijalnih primena.

Veštačka inteligencija (AI) je multidisciplinarna oblast računarstva koja ima za cilj razvoj sistema sposobnih da obavljaju zadatke koji tradicionalno zahtevaju ljudsku inteligenciju, kao što su učenje, zaključivanje, rešavanje problema, razumevanje jezika i donošenje odluka. Iako definicije mogu varirati, suština veštačke inteligencije leži u stvaranju mašina koje mogu oponašati ili čak prevazići ljudske kognitivne sposobnosti u specifičnim domenima. AI se oslanja na algoritme i tehnologije koje omogućavaju računarima da uče iz podataka, prepoznaju obrasce, prilagođavaju se novim informacijama i donose autonomne odluke.

Jedna od najčešće korišćenih definicija veštačke inteligencije opisuje je kao sposobnost mašina da interpretiraju spoljne podatke, uče iz tih podataka i koriste stečena znanja za ostvarivanje ciljeva sa visokim stepenom autonomije. AI sistemi su dizajnirani tako da simuliraju procese razmišljanja i zaključivanja, bilo kroz rešavanje složenih matematičkih problema, prepoznavanje vizuelnih obrazaca, ili analiziranje prirodnog jezika (Wang et al., 2023). Osnovni koncepti veštačke inteligencije obuhvataju nekoliko ključnih tehnologija i metoda (AI Kuwaiti et al., 2023):

1. Mašinsko učenje (Machine Learning): Mašinsko učenje je podskup veštačke inteligencije koji se fokusira na razvijanje algoritama koji omogućavaju računarima da uče iz podataka i unapređuju svoje performanse bez eksplicitnog programiranja. Algoritmi mašinskog učenja dele se na tri glavne kategorije: nadgledano učenje (supervised learning), nenadgledano učenje (unsupervised learning) i učenje pojačanjem (reinforcement learning). Nadgledano učenje koristi označene skupove podataka za obuku modela, dok nenadgledano učenje analizira neoznačene podatke kako bi identifikovalo obrasce i strukture. Učenje pojačanjem, s druge strane, koristi sistem nagrada i kazni za donošenje optimalnih odluka u dinamičnim okruženjima.
2. Duboko učenje (Deep Learning): Duboko učenje je napredna grana mašinskog učenja koja koristi višeslojne neuronske mreže za obradu velikih i kompleksnih skupova podataka. Ove mreže su inspirisane načinom na koji funkcioniše ljudski mozak, pri čemu se informacije prolaskom kroz različite slojeve mreže postupno transformišu i apstrahuju. Duboko učenje je osnova mnogih savremenih AI tehnologija, uključujući prepoznavanje govora, obrade slike i generisanje prirodnog jezika.
3. Obrada prirodnog jezika (Natural Language Processing - NLP): NLP omogućava računarima da razumeju, interpretiraju i generišu ljudski jezik. Ključni koncepti u NLP-u uključuju analizu sintakse, semantike i pragmatike jezika, što omogućava primenu u alatima poput virtuelnih asistenata, sistema za prevod i automatskog generisanja tekstova. Tehnologije NLP-a koriste se u softverskom inženjeringu za razvoj pametnih alata za kodiranje, analiza dokumentacije i automatizaciju korisničkih interakcija.
4. Računarska vizija (Computer Vision): Ova oblast omogućava računarima da analiziraju i interpretiraju vizuelne informacije iz sveta, poput slika i video sadržaja. Tehnologije računarske vizije koriste se u softverskom inženjeringu za automatizaciju testiranja korisničkog interfejsa, prepoznavanje vizuelnih obrazaca i analizu vizuelnih podataka.

5. Planiranje i optimizacija: Ovaj segment AI-a odnosi se na metode koje omogućavaju računarima da analiziraju složene sisteme, predviđaju potencijalne ishode i optimizuju resurse za postizanje definisanih ciljeva. Tehnologije planiranja i optimizacije koriste se u softverskom inženjeringu za upravljanje projektima, alokaciju resursa i optimizaciju performansi softverskih sistema.

## ZAKLJUČAK

AI tehnologije donose neosporne prednosti, poput ubrzavanja procesa razvoja, automatizacije repetitivnih zadataka, unapređenja kvaliteta koda i poboljšanja produktivnosti programera. Međutim, istovremeno, one postavljaju izazove u vezi sa pouzdanošću generisanog koda, bezbednošću, privatnošću, pristrasnošću, odgovornošću i uticajem na radnu snagu.

Primena veštačke inteligencije u razvoju softvera omogućava timovima da brže i efikasnije isporučuju rešenja, smanjujući troškove i vreme potrebno za implementaciju. Automatizacija zadataka poput pisanja osnovnog koda, testiranja, otklanjanja grešaka i dokumentacije omogućava programerima da se fokusiraju na kreativnije i strateške aspekte razvoja. AI alati postaju dragoceni saveznici u otkrivanju i eliminaciji grešaka u ranim fazama razvoja, čime se poboljšava ukupni kvalitet softverskih rešenja. Pored toga, automatizovana analiza koda i predikcija problema omogućavaju proaktivno održavanje i skalabilnost sistema.

Potencijal AI tehnologija dolazi sa značajnim rizicima koji zahtevaju pažljivo upravljanje. Generisanje koda na osnovu obimnih skupova podataka može dovesti do replikacije zastarelih praksi, ranjivosti i pravnih problema vezanih za autorska prava. Nedostatak transparentnosti u AI modelima otežava razumevanje odluka koje donose, što može negativno uticati na poverenje programera i organizacija u kvalitet generisanih rešenja. Etika korišćenja AI alata, uključujući odgovornost za greške, zaštitu privatnosti i sigurnost podataka, ostaje ključno pitanje koje zahteva regulatorne smernice i standarde.

AI tehnologije takođe redefinišu ulogu programera, transformišući način na koji rade i zahtevajući nove veštine. Programeri će sve više preuzimati uloge supervizora i kreatora strategija, dok se osnovni zadaci automatizuju. Ovo pruža priliku za unapređenje profesionalnih veština, ali istovremeno može izazvati nesigurnost kod manje iskusnih programera i promeniti dinamiku na tržištu rada.

Kako AI tehnologije nastavljaju da evoluiraju, ključni izazov leži u postizanju balansa između automatizacije i ljudske intervencije. AI treba posmatrati kao alat koji pojačava ljudske sposobnosti, a ne kao zamenu za njih. Dalji razvoj i primena AI-a zahtevaće interdisciplinarni pristup, uključujući tehničku inovaciju, etička razmatranja i pravnu regulaciju kako bi se obezbedila odgovorna, transparentna i bezbedna upotreba ovih tehnologija.

Veštačka inteligencija predstavlja revoluciju u razvoju softvera koja donosi nove mogućnosti, ali i izazove koji zahtevaju kontinuirano prilagođavanje. Budućnost AI-a u programiranju leži u njegovoj sposobnosti da obezbedi fleksibilnija, pouzdanija i sigurnija softverska rešenja, dok istovremeno osnažuje programere i organizacije da odgovore na kompleksne izazove modernog tehnološkog doba. Istraživanje na temu revolucija u razvoju softvera: uticaj veštačke inteligencije na produktivnost i kvalitet programiranja ukazuje da sa pažljivim upravljanjem, AI ima potencijal da postane ključni stub inovacija i napretka u softverskoj industriji.

## LITERATURA

- Al Kuwaiti, A., Nazer, K., Al-Reedy, A., Al-Shehri, S., Al-Muhanna, A., Subbarayalu, A. V., ... & Al-Muhanna, F. A. (2023). A review of the role of artificial intelligence in healthcare. *Journal of personalized medicine*, 13(6), 951.
- Alowais, S. A., Alghamdi, S. S., Alsuhebany, N., Alqahtani, T., Alshaya, A. I., Almohareb, S. N., ... & Albekairy, A. M. (2023). Revolutionizing healthcare: the role of artificial intelligence in clinical practice. *BMC medical education*, 23(1), 689.
- Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete Problems in AI Safety. arXiv preprint arXiv:1606.06565.
- Bharadiya, J. P. (2023). A comparative study of business intelligence and artificial intelligence with big data analytics. *American Journal of Artificial Intelligence*, 7(1), 24.
- Bostrom, N. (2014). *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press.
- Brynjolfsson, E., & McAfee, A. (2014). *The second machine age: Work, progress, and prosperity in a time of brilliant technologies*. WW Norton & Company.
- Chollet, F. (2021). *Deep learning with Python*. Simon and Schuster.
- Clark, J., & Amodei, D. (2016). Faulty Reward Functions in the Wild. OpenAI.
- Dastin, J. (2018). Amazon Scraps Secret AI Recruiting Tool That Showed Bias Against Women. Reuters.
- Estella, A. (2023). Trust in Artificial Intelligence: Analysis of the European Commission Proposal for a Regulation of Artificial Intelligence. *Ind. J. Global Legal Stud.*, 30, 39.
- Floridi, L. (2014). *The Fourth Revolution: How the Infosphere is Reshaping Human Reality*. Oxford University Press.
- Ford, M. (2018). *Architects of Intelligence: The Truth about AI from the People Building It*. Packt Publishing.
- Gomes, B., & Ashley, E. A. (2023). Artificial intelligence in molecular medicine. *New England Journal of Medicine*, 388(26), 2456-2465.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Regularization for deep learning. *Deep learning*, 216-261.
- Hunter, D. J., & Holmes, C. (2023). Where medical statistics meets artificial intelligence. *New England Journal of Medicine*, 389(13), 1211-1219.
- Kaplan, A., & Haenlein, M. (2020). Rulers of the World: An Introduction to Artificial Intelligence. *Business Horizons*, 63(1), 49-60.
- Khan, B., Fatima, H., Qureshi, A., Kumar, S., Hanan, A., Hussain, J., & Abdullah, S. (2023). Drawbacks of artificial intelligence and their potential solutions in the healthcare sector. *Biomedical Materials & Devices*, 1(2), 731-738.
- Mannuru, N. R., Shahriar, S., Teel, Z. A., Wang, T., Lund, B. D., Tijani, S., ... & Vaidya, P. (2023). Artificial intelligence in developing countries: The impact of generative artificial intelligence (AI) technologies for development. *Information Development*, 02666669231200628.

- Moor, M., Banerjee, O., Abad, Z. S. H., Krumholz, H. M., Leskovec, J., Topol, E. J., & Rajpurkar, P. (2023). Foundation models for generalist medical artificial intelligence. *Nature*, 616(7956), 259-265.
- Ng, A. (2017). Artificial Intelligence is the New Electricity. *Harvard Business Review*.
- Nikolinakos, N. T. (2023). A European Approach to Excellence and Trust: The 2020 White Paper on Artificial Intelligence. In *EU Policy and Legal Framework for Artificial Intelligence, Robotics and Related Technologies-The AI Act* (pp. 211-280). Cham: Springer International Publishing.
- O'Neil, C. (2016). *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown Publishing Group.
- Rospigliosi, P. A. (2023). Artificial intelligence in teaching and learning: what questions should we ask of ChatGPT?. *Interactive Learning Environments*, 31(1), 1-3.
- Russell, S., & Norvig, P. (2016). *Artificial Intelligence: A Modern Approach* (3rd ed.). Pearson.
- Salvagno, M., Taccone, F. S., & Gerli, A. G. (2023). Can artificial intelligence help for scientific writing?. *Critical care*, 27(1), 75.
- Scherer, M. U. (2015). Regulating artificial intelligence systems: Risks, challenges, competencies, and strategies. *Harv. JL & Tech.*, 29, 353.
- Sheth, A., Roy, K., & Gaur, M. (2023). Neurosymbolic artificial intelligence (why, what, and how). *IEEE Intelligent Systems*, 38(3), 56-62.
- Tegmark, M. (2018). *Life 3.0: Being human in the age of artificial intelligence*. Vintage.
- Ulnicane, I. (2022). Artificial Intelligence in the European Union: Policy, ethics and regulation. In *The Routledge handbook of European integrations*. Taylor & Francis.
- Umamaheswari, S., & Valarmathi, A. (2023). Role of artificial intelligence in the banking sector. *Journal of Survey in Fisheries Sciences*, 10(4S), 2841-2849.
- Vinge, V. (1993). *The Coming Technological Singularity: How to Survive in the Post-Human Era*. Vision-21: Interdisciplinary Science and Engineering in the Era of Cyberspace. NASA.
- Wang, H., Fu, T., Du, Y., Gao, W., Huang, K., Liu, Z., ... & Zitnik, M. (2023). Scientific discovery in the age of artificial intelligence. *Nature*, 620(7972), 47-60.
- Yüksel, N., Börklü, H. R., Sezer, H. K., & Canyurt, O. E. (2023). Review of artificial intelligence applications in engineering design perspective. *Engineering Applications of Artificial Intelligence*, 118, 105697.
- Zhang, B., Zhu, J., & Su, H. (2023). Toward the third generation artificial intelligence. *Science China Information Sciences*, 66(2), 121101.
-

## REVOLUTION IN SOFTWARE DEVELOPMENT: THE IMPACT OF ARTIFICIAL INTELLIGENCE ON PRODUCTIVITY AND CODE QUALITY

### **Abstract**

Software development is undergoing a fundamental transformation thanks to the integration of artificial intelligence (AI). This thesis explores how AI is redefining software engineering through code automation, development optimization, enhanced testing, and improved software system maintenance. It examines the impact of machine learning, generative models, and intelligent tools on programmer productivity, software quality, and development process efficiency. Special attention is given to ethical challenges, security risks, and shifts in traditional roles within the software industry. Based on an analysis of the latest technologies and case studies, the study concludes that AI not only accelerates software development but also lays the foundation for a new paradigm in this field.

**Keywords:** Artificial intelligence, software development, automation, machine learning, generative models, productivity, testing, ethical challenges, security, software industry.