

Kódolás az MI korában:

új szerepek, új dilemmák, új szakma

Az MI már nemcsak kódot ír – hanem beleszól a fejlesztésbe. Újratervezi a szerepeket, átalakítja a tanulási íveket és belép a döntéshozatalba is. A junior pozíciók meginognak, a Z generáció intuícióból épít, a senioroknak pedig új szemléletre van szükségük. A kódolás nem tűnik el, csak épp teljesen átalakul. Hogyan és mivé? Erről beszélgettünk két szakértővel.

Már nem az a kérdés, hogy az MI írhat-e kódot, hanem az, hogy kivel és hogyan érdemes együtt kódolnia. A mesterséges intelligencia nem csupán gyorsítja a fejlesztést, hanem újradefiniálja a fejlesztői szerepeket, megtöri a hagyományos junior-senior hierarchiát és belép a szoftverképzés legmélyebb döntési szintjeire. Miközben a generációváltás egyre természetesebben használja a párbeszédalapú kódolást, a vállalatoknak és oktatási rendszereknek is alkalmazkodniuk kell egy olyan világhoz, ahol a kreatív tervezés, a rendszergondolkodás és a technológiai intuíció fontosabb, mint a pontos szintaxis ismerete. Ez nem sci-fi: a GitHub Copilot, a Claude vagy a ChatGPT már ma is átírja a kódsorokat, és velük együtt a szoftverfejlesztés jövőjét.

MI és kódolás: új korszak a szoftverfejlesztésben

Hogyan néz ki ez az átalakulás a gyakorlatban? A nagyvállalatoknál, mint a Google vagy Microsoft, már a kód 20-

30 százalékát MI generálja. Egyes előrejelzések szerint az elkövetkezendő hónapokban ez az arány akár 90 százalékra is emelkedhet.

Egy 2025-ös globális felmérés szerint a fejlesztők 84 százaléka használ rendszeresen MI-eszközöket. A Qodo statisztikái mindemellett azt mutatják, hogy a fejlesztők 78 százaléka természetesen növekedésről számol be, míg



Matányi Zoltán

17 százaléka tízszeres hatékonyságnövekedésről. 59 százalékuk szerint az MI javított a kódminőségen, 21 százalékuk szerint viszont rontott. Más kutatások szerint az MI kimeneteinek átlagosan 61 százalékát refaktorálni kell.

A Stack Overflow Survey 2025-ben a 18–24 éves fejlesztők körében kiugró aktivitást mutatott: 91 százalék napi vagy heti szinten használ MI-eszközöket, átlagban a kódjuk 28 százalékából MI-generált, viszont csak 13 százalék az, aki több mint 50 százalékban MI-produktumra támaszkodik.

Az MI mint társfejlesztő

A mesterséges intelligencia megjelenése nemcsak a fejlesztési folyamatokat gyorsítja, hanem teljesen újradefiniálja a fejlesztői szerepeket is. Matányi Zoltán MI fejlesztési vezető, technológiai szakértő szerint a kódolás mint szakma két útra különült.

„Ha egy tapasztalt programozó az MI-t asszisztensként használja, akkor az olyan, mintha építésként irányítaná a folyamatot. Tudja, mit akar építeni,

FOTÓ: UNPLASH

Az MI már nemcsak kódot ír – hanem beszél a fejlesztésbe. Újratervezi a szerepeket, átalakítja a tanulási íveket és belép a döntéshozatalba is. A junior pozíciók megínognak, a Z generáció intuícióból épít, a senioroknak pedig új szemléletre van szükségük. A kódolás nem tűnik el, csak épp teljesen átalakul. Hogyan és mivé? Erről beszélgettünk két szakértővel.

az MI csak segít gyorsabban és precízebben rakni a „téglikat”. Ezzel szemben egy laikus, aki nem ért a fejlesztéshez, de hozzáfér egy MI-eszközhöz, olyan, mint egy háztulajdonos, aki egy robotot irányít. Így is lehet esztétikus és funkcionális dolgokat létrehozni, de a komplexitás, a struktúra és a felelősség szintje teljesen más. Ezért gondolom, hogy külön kell vizsgálni, miként használják a fejlesztők az MI-t, és hogyan azok, akiknek nincs meg a technikai tudása” – mondta.

Achilles Georgiu mesterségesintelligencia-szakértő ehhez kapcsolódva azt hangsúlyozta, hogy a fejlesztői munka nem feltétlenül szűnik meg, de gyökeresen más készségeket kíván.

A Forrester Research szerint a vállalati MI-megoldások 68 százalékát nem integrálják semmilyen központi rendszerbe, így az „MI-szigetmegoldások” növekedése strukturális problémákhoz vezet.

Amit érdemes megjegyezni

- A jövő programozója nem kódol – rendszert tervez és MI-vel kommunikál.
- A kódolás átadja a helyét a „gondolkodtató utasításoknak”, ahol az üzleti célok értelmezése és strukturálása válik elsődlegessé.
- Az MI nem váltja ki a programozót, hanem új szerepet ad neki.
- A hagyományos programozói senioritás helyét a problémamegoldó és rendszertervező gondolkodás veszi át.
- Az MI-asszisztensek lehetnek a jövő operációs rendszerei.
- Nemcsak eszközök, hanem partnerként működnek: javaslatokat tesznek, újraterveznek, optimalizálnak.
- A „digitális szemét” korábban a tiszta gondolkodás a legnagyobb érték.
- A strukturálatlan, felesleges MI-alkalmazások csak káoszt szülnek – de a jól használt MI valódi értéket teremt.

Áthelyeződött a szakmai belépési küszöb

„Amikor elkezdtem programozni, még gépi kódon dolgoztunk, bitenként nyitva-zárva a logikai kapukat. Aztán jöttek a magas szintű nyelvek, most pedig ott tartunk, hogy már természetes nyelven is utasíthatjuk az MI-t, hogy készítsen egy alkalmazást. Ez nem csak kényelmi ugrás – ez szemléletváltás. A jövő programozóinak már nem feltétlenül kell ismerniük a szintaxist, inkább azt kell tudniuk, hogyan kommunikáljanak egy intelligens rendszerrel, szervezzék meg a munkát, és miként ismerjék fel a hibákat, amiket az MI generál. Ez már nem kódolás – ez már koncepcióépí-

tés, digitális programépítés” – hozott saját példát Achilles Georgiu.

Teljesen áthelyeződött a szakmai belépési küszöb is. Az MI megjelenésével a kódolás világába való belépés ma már nem a nyelvtani szabályok ismeretén, hanem az MI hatékony irányításán és a vele való kommunikáción múlik.

Mindkét szakértő egyetért abban, hogy az MI nem elveszi a munkát, hanem átrendezi azt. A hangsúly az algoritmusok írásáról áttevődik a problémaértelmezésre, a rendszertervezésre és a folyamatos tanulásra. Aki ezt felismeri és alkalmazkodik, az nem kiszorul a fejlesztés világából, épp ellenkezőleg: kulcsszereplőjévé válik.

Mestersége címere: MI-közvetítő

Matányi Zoltán szerint azonban van egy határ, amit az MI egyelőre nem képes átlépni: „Bármennyire is intelligens, az egyelőre nem képes egy lépésben komplex, több dimenzióban történő gondolkodásra. A technológiai tudá-

sa szenior szintű lehet, de a kontextus – például üzleti célok és összefüggések – még mindig emberi feladat.”

Az MI szenior szintű technológiai tudása azonban előhoz egy másik vitatott kérdéskört, a juniorfejlesztők és programozók helyzetét. Az idén a Google belső jelentése szerint a kódok 25 százaléka már generatív MI segítségével készül, ráadásul ezek túlnyomó része juniorszintű fejlesztést vált ki.

Ez felveti a kérdést: vajon szükség lesz-e egyáltalán kezdő programozókra?

„Ami most juniorszerepkör, az már nem fog így létezni. Nem azért, mert nem lesz szükség belépő szintű tudásra, hanem mert teljesen más készségek kellenek majd és másfajta belépési küszöbök lesznek. Nem azt várjuk el, hogy tudj kódot írni, hanem hogy tudd, hogyan működik a rendszer és hogyan adj utasításokat egy MI-nek. Egyesek ezt a lépcsőt akár ki is fogják hagyni, és átugorják a jelenlegi tanulási ívet” – mondta Achilles Georgiu.

Ez az „ugrás” viszont csak akkor sikeres, ha a junior helyett újfajta „MI-navigátor” szerepkör jön létre, amely nem a kódolási szintaxisra, hanem a gondolkodásmódra és problémaértelmezésre épül.

„Lehet, hogy nem is programozónak fogjuk hívni a jövő szakembereit, hanem MI-közvetítőnek vagy MI-idomároknak, aki érti, hogyan kell egy problémát megfogalmazni, és mit kell mondani az asszisztensnek, hogy valódi megoldás szükséges” – folytatja.

Achilles Georgiu



Ebben a jövőben nem a kódolási nyelv ismerete lesz a kulcs, hanem a gondolkodásmód: az, hogy valaki strukturáltan képes rendszerekben látni egy problémát. Matányi Zoltán ezt a legősszerű modularitáshoz hasonlítja, ahol a feladat az, hogy különböző modellekből, adatbázisokból, API-ból és MI-eszközökből összerakjuk a működő és biztonságos rendszert.

Juniorból szenior: ugrás vagy evolúció?

A jelenlegi junior-szenior rendszer tehát átalakulóban van. A Z generáció (és még inkább a Gen Z MI-native alcsoport) már nemcsak megtanulja használni az eszközöket, hanem intuitívan együtt él velük. „Olyan ugrásokat látunk, hogy akár tíz szeniorfejlesztő hároméves munkáját egy tehetséges junior már nyolc óra alatt képes lemodellezni MI segítségével” – meséli Achilles Georgiu. Matányi Zoltán ezt digitális váltásként írja le:

„Mint amikor az analóg technológiát leváltotta a digitális. Nem ugyanazt csináljuk gyorsabban, hanem teljesen új lehetőségeket nyitunk ki, amelyek korábban elképzelhetetlenek voltak. Lesznek olyan fiatalok, akik egy MI-ugrással megelőzik a mostani szeniorokat. Nem a tapasztalatok alapján, hanem azért, mert más szemléletük van.”

Jól mutatja ezt a jelenséget az idei Stack Overflow Survey: a Z generációs programozók 31 százaléka nem tanult meg klasszikus programozási nyelveken dolgozni, kizárólag MI-alapú low-code vagy no-code környezetekkel dolgozik.

Oktatás és alkalmazkodás: készen állunk?

De vajon készen áll-e az oktatás erre az ugrásra? Kínában már kötelező MI-t is tanítani – Magyarországon még csak gyerekecipőben járunk. „Nem az a kérdés, mit tanítunk meg, hanem az, hogyan fejlesszük ki például a kritikus gondolkodást. A konkrét technológiák ugyanis folyamatosan változnak” – hangsúlyozza Achilles Georgiu.

A szakértők szerint a személyes MI-asszisztensek elterjedése új alapokra helyezi a tanulást: ezek az eszközök nemcsak támogatják, hanem formálják is a gondolkodást.

Matányi Zoltán szerint az MI szerepe az oktatásban éppen olyan fontos, mint a vállalatoknál: „Aki már MI-vel tanul programozni, az másképp közelít a problémákhoz. A következő generáció nem a mi példánkat fogja másolni, hanem új szabályokat ír.”

Olyannyira igaz ez, hogy a Gartner legfrissebb kutatásában is foglalkozott a jelenséggel, mint állítják, a Z generációs fejlesztők 62 százaléka már MI-alapú eszközökkel sajátítja el az első programozási tapasztalatait.

KISS FRANCISKA

McKinsey idei MI Skills Reportja szerint a jövő MI-mérnökei 43 százalékának nincs hagyományos informatikai végzettsége, hanem új típusú hibrid képzésekből érkeznek.