

Vibe coding en het probleem van de auteursrechtelijke bescherming van software

Fulco Blokhuis*

Inleiding: Vibe coding - een revolutie in softwareontwikkeling

Het werk van veel ontwikkelaars wordt steeds meer door generatieve AI (GenAI) gedaan. Software tools, zoals Lovable, Cursor, Claude Cowork, Codex en AI Studio kunnen werkende software ontwikkelen, testen en uitvoeren op basis van een of meerdere prompts. Deze vorm van programmeren wordt – sinds februari 2025¹ – vibe coding genoemd. Vibe coding is inmiddels mainstream aan het worden.

Met behulp van Claude wordt nu in een paar uur of dagen software gebouwd waar dat normaal maanden kostte. Ontwikkelaars in Silicon Valley sturen meerdere AI agents aan, ook wel agent coding genoemd. De agents schrijven zelf code, of delegeren die taak aan andere AI agents. OpenAI maakte bekend dat haar AI Agents een miljoen regels aan code hadden geschreven, aan de hand van een uitgebreide instructie, maar zonder menselijke interventie. Het aantal apps in de appstore is in december 2025 met 60% toegenomen.

Deze verschuiving van handmatig programmeren naar "prompten" heeft fundamentele gevolgen voor de juridische bescherming van software. Het traditionele auteursrechtelijke model gaat uit van menselijke creativiteit die zich uit in code, maar bij vibe-coding is de menselijke creativiteit verschoven naar de prompt, terwijl de AI de code schrijft. Dat roept de vraag op: is software nog te beschermen door het auteursrecht?

In dit artikel worden de auteursrechtelijke gevolgen van deze ontwikkeling geschetst.

Vibe coding in een notendop

Vibe coding is een ontwikkelmethode waarbij ontwikkelaars natuurlijke taal gebruiken om AI-modellen aan te sturen voor het genereren van functionele code. In plaats van regel voor regel te programmeren, beschrijft de 'vibe coder' de gewenste functionaliteit, (soms) vormgeving en 'sfeer' van de applicatie, waarna de AI de technische implementatie verzorgt. De GenAI-tool schrijft een werkende applicatie, een prototype, kan bugs verbeteren en de applicatie zelfs (laten) uitvoeren. Dit kan daarna met enkele prompts verbeterd of aangepast kan worden. Ook documentatie voor de software kan met GenAI gegenereerd worden.

Vibe coding heeft ook zijn weg gevonden naar de advocatuur. Er zijn wereldwijd advocaten die oplossingen voor hun workflow vibe coden. Op Legalquants.com delen advocaten hun vibecode-projecten. Van word-plugins, tot functies die vergelijkbaar zijn met legal AI tool Harvey.

Vibe coding is geen experimentele trend, maar levert snel productiewaardige software op die traditioneel maanden ontwikkeltijd zou vergen. Sommige developers klagen al dat ze inmiddels niet meer doen dan GenAI code te controleren.

Voor- en nadelen

Vibe coding heeft veel voordelen:

- Mensen die niet kunnen coderen kunnen nu werkende software of apps maken die anders niet gemaakt zouden worden.

* [Fulco](#) is partner bij advocatenkantoor Boekx sinds 2010. Hij is gespecialiseerd in mediarecht en intellectueel eigendom. Zijn hart ligt bij complexe

procedures op het snijvlak van technologie en intellectuele eigendom.

¹ Andrej Karpathy muntte de term op 3 februari 2025 op X.

- Het bouwen van software wordt veel toegankelijker en goedkoper.
- Het gaat bovendien sneller. Iteratiecycli worden korter, testen gaan sneller en in veel grotere omvang.
- Software op basis van weinig gebruikte programmeertalen (zoals de oude taal Cobol die gebruikt wordt in de financiële sector) kan beter onderhouden worden. Het nieuws zorgde in februari 2026 voor een gigantische daling van de aandelen van IBM:²

Er zijn ook nadelen:

- De generatieve AI tools zijn getraind op miljarden regels code, zonder dat de makers daarvoor toestemming hebben gegeven of zijn gecompenseerd. Er lopen in de VS al meer dan 90! rechtszaken over kwesties omtrent auteursrecht en generatieve AI.

Een van de eerste rechtszaken tegen AI ging precies over dit onderwerp.³
- De security en begrijpelijkheid van de code vergt (vooralsnog menselijke) aandacht. Niet-technici kunnen nu ook software maken en dat vergroot kwetsbaarheden.
- AI-code bevat aantoonbaar meer grote fouten (logic bugs, verkeerde afhankelijkheden, misconfiguraties) dan puur handgeschreven code.⁴
- Junior developers krijgen minder werk en leren daarmee ook minder goed het vak.

- De developers begrijpen hun code minder goed.
- Er kleven ook open source licentierisico's aan vibe coding. AI-tools zijn getraind op publieke code-repositories, waaronder code onder de GPL en andere *copyleft*-licenties. Wanneer een AI-tool code genereert die afgeleid is van dergelijke open source software, kan de gebruiker onbedoeld software creëren die inbreuk maakt.⁵
- Een ander nadeel is de beperkte bescherming van vibe code. Daar gaat dit artikel dieper op in.

Het auteursrechtelijk werkbegrip

Welke gevolgen heeft dit voor de auteursrechtelijke bescherming van software?

Het auteursrecht kan al sinds de jaren 80 software beschermen. In 1999 werd dat nader geregeld in de Softwarerichtlijn (91/250 EG).

Bescherming van software

Computerprogramma's worden als "werken van letterkunde" beschermd door auteursrecht, en het begrip omvat ook voorbereidend ontwerp materiaal.⁶

Artikel 1(3) van de Softwarerichtlijn 2009/24/EG⁷ ("Software RI") bepaalt dat computerprogramma's worden beschermd "wanneer het in die zin oorspronkelijk is, dat het een eigen schepping van de maker is". De bescherming ziet op "the expression in any form" van het programma (art 1(2)); in de praktijk gaat het om wat in/als code (broncode/doelcode) is vastgelegd en daarmee de concrete vormgeving van het programma.

² <https://www.reuters.com/business/ibm-posts-steepest-daily-drop-since-2000-after-anthropic-says-ai-can-modernize-2026-02-24/>

³ <https://githubcopilotlitigation.com/>

⁴ https://en.wikipedia.org/wiki/Vibe_coding

⁵ Het is verstandig om AI-gegenereerde code te scannen op dergelijke risico's, net zoals een developer bij reguliere software moet aangeven welke open source

software geïntegreerd is in de software die hij heeft gebouwd voor de opdrachtgever.

⁶ Artikel 1(1) Softwarerichtlijn 2009/24/EG.

⁷ De eerste EU-sofwarerichtlijn is **Richtlijn 91/250/EEG van de Raad van 14 mei 1991** betreffende de rechtsbescherming van computerprogramma's. Deze is later vervangen door Richtlijn 2009/24/EG.

Het auteursrecht beschermt niet functies, ideeën of concepten. Een object (het werk) komt alleen voor bescherming in aanmerking als het de persoonlijkheid van de maker weerspiegelt door vrije en creatieve keuzes. Keuzes die weliswaar vrij zijn, maar niet bijdragen aan een uniek, persoonlijk stempel van de maker, komen niet voor bescherming in aanmerking. Er moet sprake zijn van een eigen intellectuele schepping van de auteur, die diens persoonlijkheid weerspiegelt doordat hij vrije en creatieve keuzes heeft gemaakt, én van een voorwerp dat voldoende nauwkeurig en objectief identificeerbaar is. Dit is in diverse arresten uitgemaakt: Infopaq⁸, Painer⁹, en Brompton¹⁰, Cofemel¹¹, Levola¹², BSA¹³ en recent Mio/Konektra¹⁴ en Călinescu.¹⁵

De Softwarerichtlijn harmoniseert ook de kern-exploitatierchten: reproductie (incl. technisch noodzakelijke tijdelijke reproducties), bewerking/vertaling/adaptatie, en distributie (met uitputting na eerste verkoop in de EU voor die kopie, behalve verhuur). Een uitwerking van het beginsel dat inhoud vrij moet blijven om gebruikt te worden, is artikel 6 Software RI, geïmplementeerd in artikel 45m Aw. Dat artikel staat toe dat de code van software door anderen dan de rechthebbenden gedecompileerd mag worden om een ander softwareprogramma erop te laten aansluiten.

Alleen de vorm, niet de functies

De hoofdgedachte van de zogenaamde *Idea Expression dichotomy* is vastgelegd in diverse verdragen en richtlijnen:

Artikel 9 lid 2 TRIPS Verdrag

“Copyright protection extends to expressions and not to ideas, procedures, methods of operation or mathematical concepts as such.”

⁸ HvJ EU 16 juli 2009, C-5/08, NJ 2011-29/30, nr. 288. mnt. Hugenholtz (Infopaq/Danske Dagblades Forening I).

⁹ HvJ EG 1 december 2011, C-145/10, NJ 2013/66 (Painer).

¹⁰ Hof HvJ EG arrest van 11 juni 2020, Brompton Bicycle, C-833/18, EU:C:2020:461.

¹¹ HvJ EU 12 september 2019, IEF 18680, IEFbe 2938; ECLI:EU:C:2019:721 (Cofemel tegen G-Star Raw).

¹² HvJ EU 13 november 2018; IEF 22236; ECLI:EU:C:2018:899 (Levola/Smilde).

¹³ HvJ EU, 22 december 2010, C-393/09 (BSA).

Artikel 2 WIPO Auteursrechtverdrag

De bescherming van het auteursrecht strekt zich uit tot uitingen en niet tot ideeën, procedures, werkwijzen of wiskundige concepten als zodanig.

Artikel 1-2 Software Richtlijn

De bescherming overeenkomstig deze richtlijn wordt verleend aan de uitdrukkingswijze, in welke vorm dan ook, van een computerprogramma. De ideeën en beginselen die aan enig element van een computerprogramma ten grondslag liggen, met inbegrip van de ideeën en beginselen die aan de interfaces daarvan ten grondslag liggen, worden niet krachtens deze richtlijn auteursrechtelijk beschermd.

Van belang is nog dat de eerste Softwarerichtlijn mede tot stand is gekomen door een intensieve lobby van de softwarebranche. Daarvoor was er nog een debat of het auteursrecht wel een geschikte rechtsgrond was om software te beschermen, of dat een sui generis recht nodig was. Verkade schreef al in 1970 in de BIE dat auteursrecht geen goede grond was voor bescherming van software, omdat het niet de functies van de software kan beschermen.¹⁶

In het verleden is er discussie gevoerd over de vraag of een werk het persoonlijk stempel van de maker heeft. Uit een analyse van recente jurisprudentie blijkt dat dit verweer nauwelijks meer gevoerd wordt.¹⁷ Ik verwacht dat dit debat nu nieuw momentum krijgt door vibe-coding en door het Mio/Konektra-arrest, waar hieronder nader op wordt ingegaan.

¹⁴ HvJ EG 4 december 2025, C-580/23 & C-795/23 (Mio/Konektra).

¹⁵ HvJ EG 19 maart 2026, C-649/23 (Călinescu).

¹⁶ BIE, 1970, nr3, Mr. D.W.F. Verkade, Bescherming van computerprogramma's (blz. 58-69).

¹⁷ Rechtbank Gelderland 10 januari 2018, ECLI:NL:RBGEL:2018:156,

<https://deeplink.rechtspraak.nl/uitspraak?id=ECLI:NL:RB OVE:2016:4111> en

<https://deeplink.rechtspraak.nl/uitspraak?id=ECLI:NL:RB GEL:2015:4405>

GenAI Output: geen auteursrechtelijke bescherming als er geen menselijke creativiteit tot uitdrukking is gebracht.

Het uitgangspunt voor auteursrechtelijke bescherming is dat alleen menselijke creativiteit wordt beschermd.¹⁸

Werken die met GenAI zijn gemaakt zijn niet beschermd, als daar geen menselijke creativiteit aan te pas is gekomen. Waar die grenzen liggen is nu onderwerp van debat in een aantal procedures. In Europa is daar betrekkelijk weinig jurisprudentie over. Er zijn op dit moment slechts twee uitspraken bekend waarin de vraag of GenAI output beschermd is, wordt behandeld.¹⁹ In beide zaken werd bescherming afgewezen, omdat er geen menselijke creativiteit was aan te wijzen bij de generatie van de output.²⁰

Europa

In Europa is het aan de rechter om te beoordelen of software auteursrechtelijk beschermd is. Die zal dus zelf moeten beoordelen of de creatieve keuzes tot uitdrukking zijn gebracht in de software. Bij de beoordeling van de vraag of vibe coded software auteursrechtelijk beschermd kan zijn zal de eiser moeten stellen en – vermoedelijk ook - bewijzen welke creatieve menselijke keuzes zichtbaar zijn in de software.²¹

In het *Mio/Konektra* arrest²² heeft het Europese Hof bepaald dat de oorspronkelijkheid door de rechter moet worden beoordeeld aan de hand van “specifieke aard van de betrokken soort werken en of het werk uitdrukking geeft aan vrije en creatieve

keuzen die de persoonlijkheid van de auteur weerspiegelen.”

De rechter moet de creatieve keuzes “in de vorm” ervan onderzoeken en identificeren om het als auteursrechtelijk beschermd werk te kunnen aanmerken. Dat kan voor functionele werken, zoals software, en zeker software die met vibe coding tot stand is gekomen, best uitdagend zijn. Een rechter zal aan de code van software – eventueel met hulp van een deskundige – vooral kunnen zien wat functioneel is, maar creativiteit laat zich lastiger herkennen. Enkel de gebruikersinterface (de UI) heeft visuele elementen die gemakkelijker als creatief bestempeld kunnen worden.²³ Bewijs van het proces waarmee de software is gemaakt zal dus een belangrijke rol gaan spelen.²⁴ Softwaredevelopers doen er verstandig aan om het scheppingsproces te gaan vastleggen.

Functionele keuzes

De vraag kan gesteld worden of keuzes die vooral functioneel zijn of gebaseerd zijn op ervaring tot een beschermd werk kunnen leiden, omdat de Hoge Raad in het *Technip/Goossens* arrest²⁵ oordeelde dat een complex functioneel product beschermd is door auteursrecht, omdat vooral functionele keuzes zijn gemaakt bij de selectie, coördinatie en rangschikking. Verschillende schrijvers hebben het arrest echter scherp bekritiseerd.

Uit de jurisprudentie van het Europese Hof volgt dat technische keuzes alleen auteursrechtelijk beschermd kunnen zijn als er in het werk creatieve keuzes in gezien kunnen worden.²⁶

¹⁸ Visser, Robotkunst en auteursrecht, NJB 2023/454.

¹⁹ Vgl. *Městský soud v Praze* (Municipal Court of Prague) 11 oktober 2023, S. Š. v Taubel Legal advokátní kancelář s.r.o. (AI-gegenereerde afbeelding van alleen handen die een contract tekenen, geen auteursrechtelijke bescherming). En AG München 13 februari 2026, 142 C 9786/25 (KI-gegenereerde logo's zonder auteursrechtelijke bescherming wegens ontbreken van een persoonlijke creatieve schepping).

²⁰ Zie uitgebreid M.A. Smit, 'Prompteursrecht: prompteur of AI, wie is de baas in de artificiële arena?', *Ars Aequi* 2025, afl. 1 (AA20250035)

²¹ Zie ook J.R. Torenbosch, IER 2026/3, *Het MIO arrest PW=EIS#EOK/PS*.

²² HvJ EU 4 december 2025, C-580/23 en C-795/23, ECLI:EU:C:2025:941 (*Mio/Konektra*).

²³ Overigens valt de UX buiten bescherming van de softwarerichtlijn, maar mogelijk wel onder de auteursrechtelijke bescherming. HvJ EU 4 december 2025, C-580/23 en C-795/23, ECLI:EU:C:2025:941 (*Mio/Konektra*).

²⁴ Zie Hugenholtz, IER 2020/35; Visser, *Tijdschrift voor internetrecht* 2024, 108.

²⁵ HR 24 februari 2006, ECLI:NL:HR:2006:AU7508, NJ 2007/37.

²⁶ Hof HvJ EG arrest van 11 juni 2020, *Brompton Bicycle*, C-833/18, EU:C:2020:461.

In de zaak Snappet-GEU oordeelde de rechtbank²⁷ en Hof Den Haag²⁸ dat valt te betwijfelen of het Technip arrest de huidige stand van het recht nog weergeeft. Dit oordeel is door de Hoge Raad in stand gelaten.²⁹

AG van Peurseem concludeerde:

“Op basis van de huidige jurisprudentie van het Hof van Justitie voldoet een selectie en/of rangschikking van gegevens die uitsluitend gebaseerd is op wetenschappelijke of technische kennis en inzichten niet aan de eis dat er sprake moet zijn van subjectieve, vrije en creatieve keuzes. Zowel een selectie en rangschikking van wetenschappelijke gegevens, als een selectie en rangschikking van gegevens op basis van didactische kennis en inzichten, kan derhalve slechts auteursrechtelijk beschermd zijn als er bij die selectie nog ruimte is voor dergelijke keuzes.”

Ook de selectie, coördinatie en rangschikking van verschillende regels van AI *generated* code zouden tot een werk kunnen leiden.

Door vibe coden worden ervaren coders meer een soort architect van de software. Zij bedenken de architectuur van de software en geven de AI agents instructies. Het uitdenken daarvan kan zeker creatief zijn. De uitdaging die ik zie is om die creatieve keuzes duidelijk aan te kunnen wijzen in een procedure in de software die met AI-tools gemaakt is.

Uiteindelijk draait het om de vraag of de developer zijn persoonlijkheid in de software tot uitdrukking heeft kunnen brengen door vrije en creatieve keuzes tot uiting te brengen.³⁰ Hoe de invulling van die open norm in de praktijk zal gaan uitpakken voor vibe coded software zal een interessante rechtsontwikkeling zijn. Ik verwacht dat het niet gemakkelijk zal zijn om vibe coded software auteursrechtelijk beschermd te krijgen.

Amerika

In Amerika, waar het US Copyright Office (USCO) een eerste oordeel moet geven voor aldaar aangemelde werken, is er meer specifieke duiding te vinden over de grenzen van prompten en menselijke creativiteit.

Mogelijk leest u dit artikel vanaf een iPhone. Dan staat daarop iOS (26). Daarin zit code die met een AI tool is gemaakt, zo heeft Apple keurig aangegeven bij de registratie op 4 november 2025.³¹

Het USCO heeft software die met GenAI tools is gemaakt, geregistreerd, maar daarbij moet duidelijk worden aangegeven welk deel door mensen, en welk deel door de GenAI tool is geschreven. De rechthebbende/maker moet aangeven welk delen van de software van bescherming zijn uitgesloten, omdat ze met GenAI zijn gemaakt.

Het US Copyright Office heeft in zijn Copyrightability Report (2025) bevestigd dat *human authorship* vereist is voor *copyright protection*.³² De uitgangspunten die het USCO hanteert kunnen als volgt worden samengevat.

- Geen bescherming voor pure AI-output: GenAI outputs gemaakt met weinig of geen menselijke creatieve betrokkenheid (zoals reacties op standaard tekst-prompts) zijn niet vatbaar voor auteursrechtelijke bescherming.
- Er is wel bescherming mogelijk in het geval van voldoende creatieve menselijke keuzes: GenAI outputs kunnen beschermd worden indien er voldoende menselijke creatieve keuzes tot uitdrukking zijn gebracht in de productie of het eindwerk - dit moet per geval beoordeeld worden.

²⁷ Rb Den Haag, 9 november 2016, ECLI:NL:RBDHA:2016:13329 (Snappet-GEU).

²⁸ Hof Den Haag 23 juli 2023, ECLI:NL:GHDHA:2019:3060 (Snappet-GEU).

²⁹ HR 17 december 2021, ECLI:NL:HR:2021:1906 (Snappet-GEU).

³⁰ Zie voor meer verdieping ook M.A. Smit Ars Aquí januari 2025, AA20250035 Prompteursrecht.

³¹ https://publicrecords.copyright.gov/detailed-record/siebel_TX009540780. Zie voor meer voorbeelden <https://chatgptiseatingtheworld.com/2025/02/18/u-s-copyright-office-allows-registration-of-computer-programs-w-ai-generated-source-code-excluded/>
³² <https://copyright.gov/ai/Copyright-and-Artificial-Intelligence-Part-2-Copyrightability-Report.pdf>

- Het US Copyright office eist dat rechthebbenden duidelijk maken wat met GenAI is gemaakt en wat door de mens.

In de VS zijn er ook een aantal rechtszaken over dit onderwerp.

In de zaak *Thaler v. Perlmutter*³³ draaide het om de vraag of een AI-systeem dat een afbeelding had genereerd rechthebbende kon zijn. Thaler stelde zijn AI-systeem bewust als enige auteur op en ontkende iedere menselijke creatieve bijdrage — daarmee was dit altijd al een beperkte testcase voor de bredere vraag naar AI-auteurschap. Op 2 maart 2026 weigerde het U.S. Supreme Court *certiorari* in deze zaak, waardoor de uitspraak van het DC Circuit intact bleef dat de *Copyright Act* menselijk auteurschap vereist.

De zaak die voor de praktijk van vibe coding relevanter is, is *Allen v. Perlmutter*³⁴, aanhangig in de District Court of Colorado. Hierin voert Jason Allen aan dat honderden iteratieve prompts — waarmee hij genre, toon, compositie en stijl specificeerde — niet enkel onbeschermd ideeën opleveren, maar creatieve keuzes die in het eindwerk herkenbaar zijn. Deze zaak raakt de kern van het vibe coding debat: wanneer is een gedetailleerd instructiebestand voldoende om van menselijk auteurschap te spreken?

Waar ligt de grens?

Tussen output die via een enkele prompt enerzijds is genereerd en via een uitvoerig instructie bestand met sub-documenten anderzijds, zit een wereld van verschil. Het eerste zal niet tot een auteursrechtelijk beschermd werk leiden, het tweede wellicht wel. Waar het steeds om zal gaan is dat de menselijke creatieve keuzes zichtbaar moeten zijn in de output. Deze zijn doorslaggevend. Mogelijk kan een uitvoerig instructiebestand (zoals het *agent.md* wat OpenAI gebruikt voor haar AI tool Codex)³⁵, *waarmee je* een AI agent kan vertellen hoe het project werkt (tests, lint, structuur, conventies), een soort README voor agents, wel voldoende creatieve keuzes herbergen, die

vervolgens zichtbaar zijn in de output. Op die manier kan betoogd worden dat de menselijke creatieve keuzes herkenbaar zijn in de vibe coded software, zodat die auteursrechtelijk beschermd kan worden. Het USCO stelt echter dat een gedetailleerde prompt niet voldoende is, om tot een beschermbaar werk te komen. Ik vraag mij af of zij toen al voor ogen hadden dat er agents code konden schrijven op basis van één uitgebreid en gedetailleerde instructie.

Het USCO behandelt in het rapport deze mogelijkheid en stelt dat het niet voldoende is om te concluderen dat de mens via prompts invloed kan uitoefenen op de creatieve keuzes die bij de generatie van de output worden gemaakt.

Maar het USCO stelt ook vast:

*There may come a time when prompts can sufficiently control expressive elements in AI-generated outputs to reflect human authorship. If further advances in technology provide users with increased control over those expressive elements, a different conclusion may be called for.*³⁶

Het iteratief prompten leidt volgens het USCO ook niet snel tot een werk. In Azië is rechtspraak gewezen waarbij mensen met uitgebreide meerdere prompts wel een auteursrechtelijk beschermd werk konden creëren.³⁷

Gevolgen voor bescherming, contracten en M&A

Documentatie cruciaal

Partijen die software ontwikkelen, of dit nu door mensen of door GenAI wordt gedaan, moeten goed documenteren of loggen welke menselijke en welke creatieve bijdragen er zijn gemaakt.

Ook voor softwarebedrijven die geen gebruik maken van AI tools, is het noodzakelijk om goede documentatie te hebben. Vanaf 2025 moeten rechthebbenden in het kader van hun stelplicht meer onderbouwen welke menselijke creatieve keuzes er gemaakt zijn, indien zij auteursrechtelijke

³³ *Thaler v. Perlmutter*, 687 F. Supp. 3d 140 (D.D.C. 2023), bekrachtigd in 130 F.4th 1039 (D.C. Cir. 2025).

³⁴ Zie (pendente) *Jason Allen v. Shira Perlmutter* en U.S. Copyright Office, Civil Action No. 1:24-cv-02665-WJM (D. Colo.).

³⁵ <https://developers.openai.com/codex/guides/agents-md/>

³⁶ <https://copyright.gov/ai/Copyright-and-Artificial-Intelligence-Part-2-Copyrightability-Report.pdf> p 21

³⁷ Hugenholtz, *AR* 2024, p. 129.

bescherming wensen. Het is sinds 2025 immers ongeloofwaardig dat software geheel door mensen geschreven wordt. Het enkel tonen van de broncode zal geen inzicht verschaffen in de wijze waarop het tot stand is gekomen.

Voor 2025 moest de eiser ook stellen en bewijzen dat zijn werk EOK heeft en het PS droeg, maar voorheen kon de eiser in een kortgeding vaak volstaan met de stelling dat hij de maker was en 'bloot' stellen dat zijn werk niet ontleend is aan een ander werk en ontstaan is door creatieve keuzes, omdat die stelling niet of niet gemotiveerd werd betwist. Maar nu zal een gedaagde betwisten dat de broncode tot stand is gekomen door de creatieve keuzes van de eisers en moet eiser meer, aan de hand van documentatie etc., onderbouwen.

Deze stelplicht geldt ook voor softwareontwikkelaars die niet of nauwelijks gebruik maken van vibe code tools.

Gevolgen voor bescherming

De onzekerheid wanneer er voldoende creatieve keuzes zijn gemaakt bij het bouwen van software met AI-tools heeft gevolgen.

Ten eerste is er onzekerheid over bescherming: Als vibe coded software grotendeels AI-gegenereerd is zonder gedocumenteerde menselijke creatieve keuzes, is er geen of beperkte auteursrechtelijke bescherming. Dit heeft mogelijk effect op de bedrijfswaarde: software-assets die niet vatbaar voor bescherming zijn, verliezen fors aan waarde bij acquisities of licensing. En omdat functies niet beschermd zijn, is het soms vrij gemakkelijk om een vergelijkbaar product te genereren. Zelf kwam ik tot een redelijk werkend prototype met de volgende stappen:

1. Analyseer de functies van software pakket X.³⁸
2. Schrijf een prompt voor [Vibecode tool] op basis van de analyse.
3. Plak de prompt in [Vibecode tool].
4. Wijzig de output naar wens.

Het heeft ook gevolgen voor het risico op een inbreuk: Als vibe coding resulteert in code die sterk

lijkt op bestaande beschermde werken (omdat AI getraind is op publieke repositories), riskeert de gebruiker aanspraken wegens auteursrechtinbreuk.

Beveiligingsrisico's

Onderzoek uit december 2025 analyseerde vijftien applicaties gebouwd met vijf populaire vibe coding-tools en vond in totaal 69 kwetsbaarheden, waaronder kritieke lekken in API-autorisatielogica. Er zijn inmiddels ook concrete CVE's³⁹ gepubliceerd voor de tools zelf. Dit illustreert dat er ook risico's in de vibe code tools zelf zitten.

Contractuele implicaties

1. Licentievoorwaarden AI-tools: veel AI-codeertools hebben specifieke terms over eigendom van output. Deze moeten zorgvuldig worden geanalyseerd.
2. Werknemerscontracten: het is raadzaam om direct alle contracten, waarin een IP-bepaling staat die betrekking heeft op software, aan te passen aan de huidige ontwikkelingen.
3. Garanties bij softwareleveringen: leveranciers die gebruik maken van vibe-coding kunnen geen waterdichte garanties geven dat AI-code geen inbreuk maakt op rechten van derden.

M&A Due Diligence

Bij overname van softwarebedrijven die vibe coding gebruiken:

1. IP-audit essentieel: welk deel van de codebase is AI-gegenereerd? Als daar geen documentatie over bestaat, heeft de koper een belangrijk argument om de prijs te drukken.
2. Aansprakelijkheidsrisico's: wie is aansprakelijk bij auteursrechtclaims?
3. Waardering: niet-beschermbare software heeft lagere economische waarde.

³⁸ Het helpt om hierbij een overzichtelijk pakket te kiezen, maar de vibe code community van Jamie Tso

heeft ook al delen van Harvey nagemaakt.
³⁹ Common Vulnerabilities and Exposures.

Oplossingen voor alternatieve bescherming?

Gezien de beperkingen van auteursrecht op AI-gegenereerde code, is de vraag of er alternatieven zijn. Databankenrecht, bedrijfsgeheimen, octrooi, modellenrecht voor de UI en voor incidentele aspecten van de software naburige rechten bieden mogelijk een alternatief. Maar geen van deze rechten biedt dezelfde brede bescherming als het auteursrecht deed.

Octrooirecht (patent)

Een software-innovatie kan octrooieerbaar zijn als het om een technische uitvinding gaat die nieuw en inventief is; “software als zodanig” is niet genoeg voor bescherming, maar een technisch effect/technische oplossing kan wél in aanmerking komen. Alleen is dan het technisch effect beschermd en niet de code.

Bedrijfsgeheimen (knowhow)

Alles wat geheim kan worden gehouden (algoritmische details, feature roadmaps, parameterkeuzes, build/deploy scripts, bepaalde datasets) en handelswaarde heeft, kan worden beschermd als bedrijfsgeheim. Dan moeten er wel beschermingsmaatregelen genomen worden, zodat de informatie ook geheim is en blijft.

Databankenrecht (sui generis)

Als het product leunt op een dataset of verzameling gegevens (bijv. platformdata, trainingscorpora, catalogi), kan de databank als databank bescherming krijgen—naast eventueel auteursrecht op de structuur—als er substantieel is geïnvesteerd in het verzamelen/verifiëren/presenteren. In de praktijk blijkt de substantiële investeringseis vaak een hoge drempel voor een succesvol beroep op het databankenrecht. Bovendien is er bij vibe-coding juist sprake van lagere kosten, dus geen investering.

Modellenrecht (design) voor UI-elementen

Het uiterlijk van (delen van) een product kan soms als model worden beschermd (denk aan iconen, UI-elementen of schermlayouts als “appearance”), via registratie bij BOIP (Benelux) met een beperkte looptijd (verlengbaar). Daarvoor is geen menselijke

creativiteit nodig. Het hoeft slechts nieuw te zijn en een eigen karakter te hebben in de zin van artikelen 5 en 6 van de Gemeenschapsmodellenverordening, c.q. artikel 3.3 BVIE. Modellenrecht beschermt echter niet de broncode, maar enkel het uiterlijk.

Contracten en aanvullende grondslagen

Contractrecht is in de praktijk cruciaal: licentievoorwaarden, IP-overdracht bij development, NDA’s, concurrentiebedingen, en gebruiksbeperkingen kunnen bescherming/handhaving versterken. Maar het beschermt niet tegen inbreuken door derden.

Geen van deze rechtsgronden biedt echter dezelfde vergaande bescherming die software bedrijven gewend waren. De regels inzake de bescherming van bedrijfsgeheimen komen het meeste in aanmerking. Ook daarvoor zijn strakke afspraken met de softwareleverancier nodig. Maar ook dan kan je je afvragen of de broncode wel geheim is. In een analoge redenering, de output van een GenAI tool viel ook niet onder de geheimhouding van advocaten, zo oordeelde de Amerikaanse rechter.⁴⁰ Volgens judge Rakoff van het Southern District New York kan het invoeren van vertrouwelijke advocaat-cliëntcommunicatie in een extern AI-systeem worden aangemerkt als informatieverstrekking aan een derde, waardoor de vertrouwelijkheid van die communicatie verloren gaat.

Als software dezelfde bescherming moet hebben als pre 2025, dan is mogelijk een nieuw sui generis recht nodig.

Bredere impact

Ten slotte verdient de bredere impact op het open source-ecosysteem aandacht.

Het is de vraag hoe jonge developers nog aan een baan en ervaring komen als steeds meer software gebouwd wordt door AI Agents, aangestuurd door voice-memo’s van een mens.

⁴⁰ <https://natlawreview.com/article/your-ai-conversations-are-not-privileged-what-new-sdny-ruling->

[means-every-lawyer](https://natlawreview.com/article/your-ai-conversations-are-not-privileged-what-new-sdny-ruling-)

Een aantal developers stellen dat hun productiviteit met 10x is toegenomen. Dat leidt er ook toe dat er minder mensen nodig zijn.⁴¹

Vibe coding dreigt de gemeenschap van open source-ontwikkelaars te destabiliseren: maintainers⁴² worden overspoeld met door AI-gegenereerde pull requests en bugreports van lage kwaliteit. Sommige projecten verbieden AI-gegenereerde bijdragen expliciet. De open source gemeenschap is juist gericht op het breed delen van software en het met elkaar verbeteren, terwijl die open source code gebruikt is voor training van de modellen waarmee de vibe-coding nu werkt.

Het wordt nog gekker. Er is zelfs een – zijdelings relevant - verhaal van een maintainer Scott Shambaugh die te maken kreeg met lasterlijke blogs, omdat hij aan een AI-agent op basis van Open Claw weigerde toestemming te geven. De blog was geschreven door de AI agent. De AI-agent verzamelde informatie over Shambaugh en publiceerde een blogpost getiteld “Gatekeeping in Open Source: The Scott Shambaugh Story”, met persoonlijke aanvallen en speculaties over zijn motieven. De AI-agent verspreidde die tekst vervolgens via GitHub-comments en sociale kanalen, beschuldigde hem van “gatekeeping” en probeerde druk te zetten om de code toch te accepteren.

Conclusie

Vibe coding en agentic coding transformeert softwareontwikkeling exponentieel en fundamenteel. Het creëert echter ook juridische risico's omtrent de bescherming. AI-gegenereerde output blijft problematisch te beschermen door middel van het auteursrecht.

Softwareontwikkelaars en vibe coders worden aangeraden om hier beleid voor te ontwikkelen:

1. documenteer menselijke creatieve input, leg vast wat er met welke AI-tool precies wordt gemaakt;

2. combineer auteursrecht met databankenrecht, bedrijfsgeheimen of andere rechtsgronden;
3. pas contracten aan de vibe code realiteit aan.

De Europese wetgever zal mogelijk moeten ingrijpen met sui generis-bescherming, zoals eerder gebeurde bij databanken en halfgeleider-topografieën. Daarmee wordt het debat van eind jaren 80 bijna vijftig jaar later weer relevant. En krijgt Verkade 56 jaar na zijn BIE-artikel alsnog gelijk, het auteursrecht is niet meer geschikt om software goed te beschermen.

⁴¹ <https://www.nytimes.com/2026/03/12/magazine/ai-coding-programming-jobs-claude-chatgpt.html>

⁴² In security-/opensource-context is een “maintainer” degene die formeel verantwoordelijk is voor het

onderhouden van een project of component: code, releases, security-fixes en het accepteren of afwijzen van bijdragen.