

Future Energy
Lab

REPORT

Green Coding

Mit stromsparender Software zu
einer nachhaltigeren Digitalisierung

Impressum

Herausgeber:

Deutsche Energie-Agentur GmbH (dena)
Chausseestraße 128 a
10115 Berlin
Tel: +49 30 66 777-0
Fax: +49 30 66 777-699
E-Mail: info@dena.de
Internet: www.dena.de, www.future-energy-lab.de

Autorinnen und Autoren:

Geerd-Dietger Hoffmann, Green Coding Solutions GmbH
Prof. Dr. Verena Majuntke, HTW Berlin
Arne Tarara, Green Coding Solutions GmbH,
Jana Hammerer, dena
Eva Steiger, dena
Hendrik Zimmermann, dena

Konzeption & Gestaltung:

Heimrich & Hannot GmbH

Stand:

12/2024

Alle Rechte sind vorbehalten. Die Nutzung steht unter dem Zustimmungsvorbehalt der dena.

Bitte zitieren als:

Deutsche Energie-Agentur (Hrsg.) (dena, 2024) „Green Coding – Mit stromsparender Software zu einer nachhaltigeren Digitalisierung“



Bundesministerium
für Wirtschaft
und Klimaschutz

Die Veröffentlichung dieser Publikation erfolgt im Auftrag des Bundesministeriums für Wirtschaft und Klimaschutz. Die Deutsche Energie-Agentur GmbH (dena) unterstützt die Bundesregierung in verschiedenen Projekten zur Umsetzung der energie- und klimapolitischen Ziele im Rahmen der Energiewende.

Inhalt

Vorwort	4
Executive Summary	5
1. Einleitung	7
2. Stand der Wissenschaft	9
2.1 Begriffsdefinition Green Coding	11
2.2 Green Coding in den Phasen des Softwarelebenszyklus	13
2.3 Herausforderungen bei der Anwendung von Green Coding.....	17
3. Umfragen	19
3.1 Entwicklung der Umfragen.....	20
3.2 Befragung von Expertinnen und Experten und offene Umfrage	20
3.3 Auswertung der Umfrageergebnisse	22
4. Handlungsempfehlungen	34
4.1 Maßnahmenkatalog.....	35
4.2 Ansatz und Methodik.....	39
4.3 Erarbeitung der Handlungsempfehlungen	39
4.3.1 Politik	39
4.3.2 Unternehmen	43
4.3.3 Bildungseinrichtungen	45
4.3.4 Kommunikation und Zusammenarbeit	47
5. Fazit	49
Glossar	51
Literaturverzeichnis	52

Vorwort

Sehr geehrte Leserinnen und Leser,

die fortschreitende Digitalisierung unserer Gesellschaft und Wirtschaft bringt erhebliche Herausforderungen in Bezug auf den Energie- und Ressourcenverbrauch im Bereich der Informations- und Kommunikationstechnologie (IKT) mit sich. Softwareanwendungen spielen dabei eine zentrale Rolle: Sie treiben nicht nur innovative und effiziente Prozesse voran, sondern tragen auch maßgeblich zu einem wachsenden Energieverbrauch bei – sowohl durch ihre direkte Nutzung als auch durch den Betrieb der zugrundeliegenden Infrastruktur. Angesichts der globalen Klimaziele und der Dringlichkeit, diese zu erreichen, rücken Themen wie die energieeffiziente Gestaltung von Software, das sogenannte Green Coding, zunehmend in den Fokus. Mit diesem Report widmen wir uns bei der dena erstmals eingehend diesem wichtigen Thema.

Dieser Report knüpft an eine Reihe von Arbeiten des Future Energy Lab der Deutschen Energie-Agentur (dena) an, die sich mit dem Stromverbrauch digitaler Technologien und Anwendungen befassen. Hierzu gehören unter anderem die im Oktober 2023 veröffentlichte Studie „Neue Energiebedarfe digitaler Technologien – Untersuchung von Schlüsseltechnologien für die zukünftige Entwicklung des IKT-bedingten Energiebedarfs“ (dena, 2023a), die Studie „Energieeffiziente Künstliche Intelligenz für eine klimafreundliche Zukunft“ (dena, 2024) sowie der Leitfaden „Blockchains und ihren Stromverbrauch neu denken – Ein Leitfaden für das stromsparende Design dezentraler Dateninfrastrukturen“ (dena, 2023b). Mit Green Coding betreten wir nun Neuland, indem wir die Rolle von Software entlang ihres gesamten Lebenszyklus analysieren und systematische Potenziale zur Strom-einsparung aufzeigen. Dieser Report zeigt, dass zwar einige Coderinnen und Coder bereits darauf achten, stromsparend zu programmieren, die Bedeutung des Themas aber in der Breite der Branche noch nicht hinreichend angekommen ist. Dies ist besonders relevant, da Green Coding auch aus betriebswirtschaftlicher Sicht in den allermeisten Fällen anzuraten ist. Umso wichtiger erscheint es, Green Coding im Sinne einer ganzheitlichen – ökonomischen wie ökologischen – Nachhaltigkeit zu fördern.

Dieser Report versteht sich als Einführung in das Thema Green Coding und bietet eine detaillierte Analyse der Potenziale, die in der energieeffizienten Softwaregestaltung liegen. Dabei beleuchten wir den aktuellen Stand der Praxis und analysieren, wie Green Coding in den verschiedenen Phasen des Softwarelebenszyklus von der Entwicklung bis hin zum Betrieb umgesetzt werden kann. Ziel ist es, nicht nur das Bewusstsein für die Relevanz dieses Ansatzes zu schärfen, sondern auch konkrete Handlungsempfehlungen zu formulieren, die Unternehmen, Politik und Bildungseinrichtungen dabei unterstützen, stromsparende Softwarelösungen erfolgreich in der Praxis zu implementieren.

Green Coding wird in dieser Studie nicht nur als technisches Konzept betrachtet, sondern vor allem als gesellschaftlicher Ansatz. Es geht darum, wie eine ressourceneffiziente Digitalisierung gestaltet werden kann, bei der alle gesellschaftlichen Akteure – von der Wirtschaft über die Politik bis hin zur Zivilgesellschaft – ihren Beitrag leisten müssen. Sensibilisierungsmaßnahmen und eine verstärkte Bildungsarbeit spielen dabei eine zentrale Rolle, um das nötige Bewusstsein für die Herausforderungen und Potenziale zu schaffen. Mit dieser ersten umfassenden Studie möchten wir dazu beitragen, das Verständnis für Green Coding zu vertiefen und verschiedene Akteure zu einer aktiven Mitgestaltung einer nachhaltigen digitalen Zukunft zu bewegen.

Wir freuen uns, Sie mit dieser Studie auf eine spannende und zukunftsweisende Reise mitzunehmen und laden Sie herzlich ein, Teil der Diskussion zu werden.



Hendrik Zimmermann
Teamleiter
Digitale Technologien
Deutsche Energie-Agentur
(dena)



Jana Hammerer
Expertin
Digitale Technologien
Deutsche Energie-Agentur
(dena)

Executive Summary

Die digitale Transformation und der zunehmende Einsatz moderner Informations- und Kommunikationstechnologie (IKT) führen zu einem erheblichen Anstieg des Energieverbrauchs und der damit verbundenen Emissionen. Studien schätzen, dass die IKT derzeit zwischen 1,8 % und 3,9 % der globalen CO₂-Emissionen ausmacht, wobei ein weiterer Anstieg ohne gezielte Gegenmaßnahmen erwartet wird (Freitag et al., 2022). Diese Entwicklung stellt eine wachsende Herausforderung für die globalen Klimaschutzbemühungen dar.

Software ist dabei ein maßgeblicher Treiber des Stromverbrauchs und der damit einhergehenden Emissionen der IKT. Software verursacht ihren Stromverbrauch indirekt durch die Nutzung von Hardware. Je ressourcenintensiver Software ist, desto stärker wird die zugrunde liegende Hardware beansprucht, was zu einem höheren Stromverbrauch führt. Green Coding bietet hierbei eine innovative und wirkungsvolle Möglichkeit, den Stromverbrauch und die damit verbundenen Emissionen von Software und damit der IKT zu senken – und dies ohne Einbußen bei Leistung oder Funktionalität (Hilty et al., 2015). Durch gezielte Maßnahmen in der

Softwareentwicklung lassen sich nicht nur ökologische Vorteile erzielen, sondern auch wirtschaftliche Effizienzgewinne realisieren. Daher ist Green Coding nicht nur ein technisches Werkzeug, sondern auch ein strategischer Ansatz, der eine Schlüsselrolle bei der nachhaltigen Digitalisierung und der Bekämpfung der Klimakrise einnehmen sollte.

Das Ziel dieses Reports ist es, das Potenzial von Green Coding zu untersuchen und daraus konkrete Handlungsempfehlungen für verschiedene Akteure zu entwickeln. Die Methodik des Reports umfasst eine umfassende Literaturrecherche, Umfragen bei Expertinnen und Experten sowie der Öffentlichkeit und die Durchführung von Experten-Workshops. Auf Basis dieser Ansätze wurden in Kapitel 4 die folgenden Handlungsempfehlungen im Detail vorgestellt. Die Empfehlungen adressieren unterschiedliche Herausforderungen und Zielgruppen und sollen eine breite Anwendung von Green Coding erleichtern. Ziel ist es, die ökologische Belastung durch IKT nachhaltig zu reduzieren und Green Coding zu einem festen Bestandteil nachhaltiger Softwareentwicklung zu machen.

Handlungsempfehlung für die Politik	
Einheitliche Definition von Green Coding	Die Bundesregierung sollte einen Stakeholder-Prozess initiieren, um eine einheitliche Definition von Green Coding zu entwickeln, die bundesweit Anwendung finden und weiteren Politiken zugrunde liegen kann.
Transparenzvorschriften	Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von eingebetteten Emissionen bei Hardware entlang ihres gesamten Lebenszyklus einführen.
	Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von operativen und eingebetteten Emissionen durch IT-Dienstleistungen und Cloud-Services entlang ihres Lebenszyklus einführen.
	Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von operativen und eingebetteten Emissionen bei Software entlang ihres Lebenszyklus einführen.
Schaffung von Anreizen für Unternehmen	Die Bundesregierung sollte eine Förderung für Green-Coding-Pilotprojekte einsetzen.
	Die Bundesregierung sollte Maßnahmen zur Optimierung von Software für die Zertifizierung mit dem Blauen Engel finanziell fördern.
	Die Bundesregierung sollte den Erwerb von Software (bzw. Softwarelizenzen), welche mit dem Blauen Engel zertifiziert ist, finanziell fördern, da dies nachhaltige Entwicklungspraktiken stärkt, langfristig Energie- und Ressourceneinsparungen ermöglicht und die Wettbewerbsfähigkeit der deutschen IT-Branche fördert
	Die Bundesregierung sollte gesetzlich vorschreiben, dass in der öffentlichen Beschaffung und im mittelfristigen Einsatz ausschließlich energie- und ressourceneffiziente Software zugelassen wird, um ökologische Nachhaltigkeitsziele wirksam zu unterstützen.

Handlungsempfehlungen für Unternehmen

Stärkung des Bewusstseins	Unternehmen, die Software entwickeln, sollten Kennzahlen auf Softwareprodukt- und Softwareprojektebene einführen, die den Stromverbrauch und die operativen und eingebetteten Emissionen von Software entlang ihres Lebenszyklus erfassen.
	Unternehmen, die Software einkaufen, sollten Transparenz in Bezug auf den Stromverbrauch und die operativen und eingebetteten Emissionen für Hardware, IT-Dienstleistungen, Cloud-Services und Software bei den Herstellern einfordern.
Unternehmensinterne Ausbildungen und Schulungen	Unternehmen sollten interne und zielgruppengerechte Bildungsangebote zum Thema Green Coding bereitstellen, um unternehmensinternes Wissen im Bereich Green Coding aufzubauen.
	Unternehmen sollten Angebote erarbeiten, um ihre Führungskräfte für Green Coding zu sensibilisieren.

Handlungsempfehlungen für Bildungseinrichtungen

Integration von Green Coding in Curricula von Hochschulen	Hochschulen/Universitäten sollten Green Coding in die akademischen Curricula von informatikorientierten Studiengängen integrieren.
	Hochschulen/Universitäten sollten Kurse und Seminare anbieten, um auch Studentinnen und Studenten in nicht informatikorientierten Studiengängen für nachhaltige IT und Software zu sensibilisieren.
Mehr Bildungsangebote	Volkshochschulen, Verbände und weitere Bildungseinrichtungen sollten Schulungsangebote zu Green Coding entwickeln und organisieren.

Handlungsempfehlungen im Bereich Kommunikation und Zusammenarbeit

Erfolgsberichte	Verbände, NGOs und Behörden sollten konkrete Beispiele für die Anwendung von Green Coding sammeln, aufbereiten und öffentlich verfügbar machen.
	Unternehmen sollten eigene Erfolgsgeschichten öffentlich teilen und über soziale Medien und Informationskanäle eine breite Öffentlichkeit über die Erfolgsgeschichten informieren.
	Ministerien und Verbände sollten Preise für herausragende Green-Coding-Praktiken verleihen und sowohl den Preis als auch die Preisverleihung öffentlichkeitswirksam bewerben.

Fazit

Dieser Report zeigt, dass Green Coding eine bedeutende Rolle bei der nachhaltigen Transformation der IKT spielen kann. Durch gezielte Maßnahmen kann Green Coding den Stromverbrauch und die Emissionen des Sektors erheblich senken, ohne Leistungseinbußen zu verursachen. Green Coding kann so einen entscheidenden Beitrag zum Klimaschutz leisten. Die erfolgreiche Einführung von Green Coding erfordert jedoch ein koordiniertes Zusammenspiel zwischen Bildung und Forschung, politischen Anreizen und Unternehmensinitiativen, um bestehende Wissenslücken, technische und organisatorische Hürden sowie unterschiedliche Verständnisse von Green

Coding zu überwinden. Die Umsetzung der im Report erarbeiteten Handlungsempfehlungen bietet eine fundierte Grundlage, um Green Coding in der Softwareentwicklung zu fördern und damit den Stromverbrauch der IKT-Branche zu reduzieren.

1.

Einleitung

Deutschland hat sich im Rahmen des Pariser Klimaabkommens dazu verpflichtet, bis 2045 klimaneutral zu werden, was eine drastische Reduktion der Treibhausgasemissionen in allen Sektoren erfordert. Dies bedeutet, fossile Großkraftwerke durch dezentrale Anlagen zur Nutzung erneuerbarer Energien zu ersetzen und die Deckung des Strombedarfs in Sektoren wie Verkehr, Industrie und Gebäude durch Elektrifizierung klimaneutral zu gestalten. Diese Transformation erfordert eine enge Integration der Sektoren und bringt neue Herausforderungen mit sich, insbesondere durch die fluktuierende Energieerzeugung aus erneuerbaren Quellen. Die Koordination zwischen Energieerzeugung und -verbrauch ist essenziell und genau hier spielt die Digitalisierung eine entscheidende Rolle. Sie ermöglicht die Flexibilisierung und Dezentralisierung des Energiesystems und erhöht durch Technologien wie Echtzeitüberwachung und KI-basierte Prognosen die Energieeffizienz und Steuerbarkeit des Gesamtsystems.

Gleichzeitig steigt durch die Digitalisierung der Strombedarf der Informations- und Kommunikationstechnologie (IKT) erheblich an. Peer-reviewed Studien schätzen den derzeitigen Anteil der IKT an den globalen CO₂-Emissionen auf 1,8 % bis 2,8 %. Unter Berücksichtigung indirekter Emissionen könnte dieser Wert jedoch bis zu 3,9 % betragen, wie eine Analyse von Freitag et al. (2022) zeigt. Ohne signifikante politische und industrielle Anstrengungen wird erwartet, dass die Emissionen weiter steigen. Aktuelle politische Rahmenbedingungen und Klimaversprechen reichen häufig nicht aus, um die Ziele des Pariser Abkommens zu erreichen. Ein sektorweiter Regulierungsansatz wird als notwendig erachtet, um die Emissionen der IKT mit den globalen Klimazielen in Einklang zu bringen (Freitag et al., 2022).

Green Coding bietet in diesem Kontext innovative Ansätze, um Software energie- und ressourceneffizienter zu gestalten, und kann dadurch den Stromverbrauch und die CO₂-Emissionen in der IKT-Branche nachhaltig reduzieren. Im Fokus steht die Entwicklung von Software, die effizienter arbeitet, ohne dabei Funktionalität oder Leistung einzubüßen. Angesichts des wachsenden Drucks, Klimaneutralität zu erreichen, werden die Optimierung von Software und der Einsatz energieeffizienter Hardware immer relevanter. Dieser Report untersucht, wie Green Coding die IKT nachhaltiger gestalten kann, und beleuchtet praxisnahe Lösungen, die von verschiedenen Stakeholder-Gruppen implementiert werden können. Ziel ist es, konkrete Ansätze und Handlungsempfehlungen zu identifizieren, die den Stromverbrauch in der IT-Branche senken und die Digitalisierung langfristig klimafreundlicher gestalten.

Der Report basiert auf einer detaillierten Analyse des derzeitigen Stands der Wissenschaft und direkten Befragungen von Vertreterinnen und Vertretern aus Wirtschaft, Politik, Forschung und Zivilgesellschaft, um möglichst praxisnahe Handlungsempfehlungen abzuleiten.

Der Report ist in mehrere Kapitel unterteilt, die jeweils spezifische Aspekte des Green Coding beleuchten. Jedes Kapitel kann einzeln gelesen werden, um gezielte Informationen zu erhalten. Im Zusammenhang bietet der Report einen umfassenden Überblick über die Thematik. Die Kapitel und ihre Motivation stellen sich wie folgt dar: Nach der thematischen Einführung zeigt das zweite Kapitel die Ergebnisse einer umfassenden Literaturrecherche, die sich mit der Definition und dem aktuellen Forschungsstand zu Green Coding auseinandersetzt. Ziel ist es, ein theoretisches Fundament zu legen und bestehende Wissenslücken zu identifizieren. Zudem werden in diesem Kapitel die praxisrelevanten Forschungsfragen vorgestellt, die sich auf Basis der Literaturrecherche ergeben haben. Diese Forschungsfragen dienen als Grundlage für die anschließende Entwicklung der Umfragen und Interviews, deren Methodik im dritten Kapitel beschrieben wird. Ziel der Befragungen war es, Einblicke in die aktuellen Praktiken und Herausforderungen im Bereich Green Coding zu gewinnen sowie Handlungsempfehlungen zu identifizieren. Die gesammelten Daten wurden analysiert, um fundierte Erkenntnisse daraus abzuleiten. Das vierte Kapitel stellt die wichtigsten Ergebnisse und ihre Implikationen dar. Auf Basis der gewonnenen Erkenntnisse wurden konkrete, umsetzbare Handlungsempfehlungen für relevante Stakeholder im Bereich Green Coding erarbeitet, die in Kapitel 4 vorgestellt werden. Der Report richtet sich dabei gezielt an Akteure aus Politik, Bildungsinstitutionen und Unternehmen und schlägt sowohl technische als auch organisatorische Maßnahmen vor. Diese Handlungsempfehlungen bieten praxisnahe Leitlinien, um Green Coding als Standard in der Softwareentwicklung zu etablieren und damit einen nachhaltigen Beitrag zur Reduktion des Stromverbrauchs im IT-Sektor zu leisten. Dieser Report dient damit als Ressource für alle Stakeholder, die an der Etablierung von Green Coding als Standard in der Softwareentwicklung interessiert sind oder die nach Lösungen suchen, den Sektor IKT nachhaltiger zu gestalten.

Der Report wurde von der Green Coding Solutions GmbH in Zusammenarbeit mit Prof. Dr. Verena Majuntke, Professorin für Software Engineering an der Hochschule für Technik und Wirtschaft Berlin, und der Deutschen Energie-Agentur (dena) erarbeitet. Unterstützt wurde das Projekt durch die Expertise weiterer renommierter Partner: der SYNGENIO AG, der bluehands GmbH & Communication KG, der envite consulting GmbH, der DVC – Digital Venture Consultants (ZiTOS GmbH), oktobit green coaching sowie der metafinanz Informationssysteme GmbH. Gemeinsam setzen sie sich dafür ein, Green-Coding-Praktiken als Standard in der Softwareentwicklung zu etablieren und so einen Beitrag zu stromeffizienten Digitalisierungsprozessen zu leisten. Diese Gruppe an Teilnehmern wird im Folgenden als Konsortium bezeichnet.



2.

**Stand der
Wissenschaft**

Der Energie- und Ressourcenverbrauch der Informations- und Kommunikationstechnologie (IKT) nimmt stetig zu und stellt eine wachsende Herausforderung für die Erreichung globaler Klimaziele dar. Die dena-Studie „Neue Energiebedarfe digitaler Technologien“ zeigt, dass Expertinnen und Experten unterschiedliche Prognosen zur künftigen Entwicklung des Energiebedarfs in diesem Bereich abgeben (dena, 2023). Der derzeitige Anteil der IKT an den globalen CO₂-Emissionen wird in peer-reviewed Studien auf 1,8 % bis 2,8 % geschätzt, während er unter Einbeziehung indirekter Emissionen laut Freitag et al. (2022) bis zu 3,9 % betragen könnte. Ohne wirksame politische und industrielle Maßnahmen ist von einem weiteren Anstieg der Emissionen auszugehen. Gelenbe (2023) schätzt, dass die IKT jährlich etwa 10 % des weltweiten Gesamtstrombedarfs verbraucht, abhängig von den definierten Systemgrenzen.

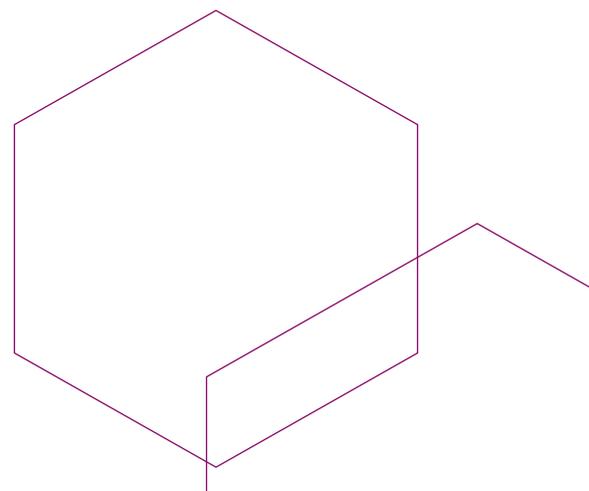
Rechenzentren machten im Jahr 2022 laut International Energy Agency (IEA) allein rund 2 % (460 TWh) des globalen Stromverbrauchs aus – mit einer weiter steigenden Tendenz (Rozite et al., 2023). Eine Analyse von Goldman Sachs zeigt, dass der Strombedarf von Rechenzentren zwischen 2015 und 2019 trotz wachsender Nachfrage relativ konstant bei etwa 200 TWh pro Jahr lag, was auf kontinuierliche Effizienzsteigerungen zurückzuführen ist. Seit 2020 jedoch steigt der Energieverbrauch wieder, insbesondere durch den wachsenden Einsatz Generativer Künstlicher Intelligenz (KI), die nach Schätzungen einen zusätzlichen jährlichen Bedarf von etwa 200 TWh verursachen könnte (Goldman Sachs, 2024).

Im Vergleich zu anderen Sektoren wie Transport oder Produktion galt die IKT lange Zeit als verhältnismäßig umweltfreundlich. Der stark wachsende Stromverbrauch, die steigenden CO₂-Emissionen und die ressourcenintensive Produktion von Hardwarekomponenten zeigen jedoch die ökologische Belastung auf, die die IKT mit sich bringt. Hier gilt es zu beachten, dass zwei Arten von Emissionen durch IKT verursacht werden: die operativen Emissionen, die bei der Nutzung der Hardware, beispielsweise durch den Stromverbrauch von Software, entstehen, und die eingebetteten Emissionen, die bei der Herstellung der Hardware anfallen, wie etwa durch die Nutzung von Treibstoff beim Transport. Hinzu kommen der Einsatz seltener Metalle und die Nutzung natürlicher Ressourcen, wie zum Beispiel die Verwendung von Wasser zum Kühlen von Rechenzentren, was erhebliche soziale und ökologische Herausforderungen mit sich bringt. Als problematisch erweist sich insbesondere das steigende Volumen an Elektroschrott, das weltweit eine erhebliche Belastung für Umwelt und Gesundheit darstellt (Rajesh, 2022).

In Bezug auf den kontinuierlichen Ausbau der Informations- und Kommunikationstechnologie nimmt Software eine Schlüsselrolle ein. IKT lässt sich in drei wesentliche Bereiche einteilen: (1) Rechenzentren, (2) Netzwerke und (3) Endgeräte (Hilty et al., 2009; Malmodin et al., 2024). Steigen die Anforderungen von Software, kann sich dies auf einen oder mehrere Bereiche der IKT auswirken. So führen höhere Anforderungen von Software an die Hardwareressourcen zu einer gesteigerten Nachfrage nach Rechenzentrumskapazitäten. Darüber hinaus werden Endgeräte obsolet, wenn Software aufgrund von Hardwareanforderungen auf den Geräten nicht mehr oder nicht mehr in der für die Endnutzerschaft annehmbaren Qualität ausgeführt werden kann. Nutzt Software zunehmend umfangreichere Datenmengen, so muss das Netzwerk entsprechend ausgebaut werden, um dies zu ermöglichen. Die wachsenden Anforderungen an die Hardware aufgrund der zunehmenden Softwarekomplexität erfordern eine Erweiterung der gesamten IKT-Infrastruktur, um weiterhin eine hohe Qualität der Softwarenutzung sicherzustellen. Zudem führt die fortschreitende Digitalisierung zu einer zunehmenden Durchdringung des täglichen Lebens mit Software, wodurch die Anforderungen an die Leistungsfähigkeit der IKT steigen (D21, 2022).

Ein Ansatz, um die mit IKT assoziierten ökologischen Auswirkungen zu adressieren, ist die Optimierung von Software in Bezug auf ihren Stromverbrauch und die Inanspruchnahme von Hardware. Hier setzt Green Coding an. Green Coding fokussiert auf die Entwicklung von ressourceneffizient arbeitender Software und die Minimierung des Stromverbrauchs. Dies reduziert den Stromverbrauch über alle IKT-Bereiche hinweg und trägt dazu bei, den CO₂-Fußabdruck der Branche zu verkleinern. Die durch Green Coding entstehenden nachhaltigen IT-Lösungen können einen maßgeblichen Beitrag dazu leisten, den steigenden Strombedarf und die damit verbundenen Emissionen der IKT langfristig in den Griff zu bekommen.

Im folgenden Abschnitt 2.1 werden zunächst aktuell bestehende Begriffsdefinitionen von Green Coding analysiert, gefolgt von einer Zusammenfassung von Hebelpunkten zur Einsparung von CO₂ im Softwareentwicklungszyklus in Abschnitt 2.2. Abschließend wird in Abschnitt 2.3 aufgezeigt, welche Hindernisse sich in der Praxis bei der Umsetzung dieser Maßnahmen ergeben und welche Herausforderungen sich im Umgang mit den identifizierten Hebelpunkten stellen.



2.1 Begriffsdefinition Green Coding

Zum Zeitpunkt der Erstellung des vorliegenden Reports existieren nur sehr wenige wissenschaftliche Publikationen, die eine explizite Definition des Begriffs Green Coding geben. Gemäß Junger et al. (2023) ist Green Coding der „Akt des Entwerfens, Entwickelns, Wartens und (Wieder-) Verwendens von Softwaresystemen auf eine Weise, die so wenig Strom und natürliche Ressourcen wie möglich benötigt. Green Coding-Methoden oder -Praktiken bedeuten somit jede Handlung oder Nutzung von Technologie, die darauf abzielt und geeignet ist, dies weiter zu fördern“ (Junger et al., 2023). Radersma beschreibt Green Coding als „die Praxis der Code-Optimierung für minimalen Energieverbrauch“ (Radersma, 2022). Weitere wissenschaftliche Beiträge geben keine explizite Definition des Begriffs an und verstehen Green Coding als Maßnahmen zur Reduktion des Stromeinsatzes für Softwaresysteme (Verdecchia et al., 2021; Poth und Rjollj, 2023).

In nicht wissenschaftlichen Quellen finden sich hingegen eine Vielzahl von Definitionen für den Begriff Green Coding. Auf den Seiten von IONOS wird Green Coding als eine Programmierpraxis definiert, „die darauf ausgelegt ist, möglichst umweltverträgliche Softwaresysteme zu erstellen“. Dabei setzt Green Coding laut IONOS „auf allen Ebenen des Softwareentwicklungsprozesses an“ (IONOS Redaktion, 2023). Eine ähnliche Definition gibt die Website exxeta.com an, auf der Green Coding definiert ist als „eine Reihe von Prinzipien und Techniken, die darauf abzielen, den Stromverbrauch von Software zu reduzieren und gleichzeitig die Effizienz, Leistung und Innovationsfähigkeit von Anwendungen zu steigern“ (Russ, 2023).

Im Booklet zur Workshop-Reihe Green Coding, die im Auftrag des Bundesministeriums für Umwelt, Naturschutz, nukleare Sicherheit und Verbraucherschutz (BMUV) durchgeführt wurde, wird Green Coding wie folgt definiert: „Green Coding bezieht sich [...] auf den möglichst geringen Stromverbrauch der Software. Bezogen auf die Software geht es insbesondere um die Bereiche Softwarearchitektur, die konkreten Softwarekomponenten und die Betriebsplattform. Bezogen auf die Hardwareressourcen ist das Ziel von Green Coding die Ermöglichung einer langen Nutzung der technischen Infrastruktur durch effizientere Software“ (Community Nachhaltige Digitalisierung, 2024).

Auf internationalen Seiten lassen sich Definitionen finden, die das Thema Green Coding eingeschränkter auffassen. So sieht IBM Green Coding als ökologisch nachhaltige Praxis, die auf die Minimierung von Energie abzielt, die für die Abarbeitung von Codezeilen benötigt wird. Ein weiter gefasster Begriff ist laut IBM das sogenannte Green Computing, das es zum Ziel hat, die Umweltauswirkungen von Technologie zu begrenzen. Dazu gehört die Reduzierung des CO₂-Fußabdrucks in intensiven Betriebsabläufen wie Fertigungsstraßen und Rechenzentren

oder sogar im täglichen Betrieb von Geschäftsteams. Unter dieses umfassendere Green-Computing-Konzept fallen auch Green-Software-Anwendungen, die unter Verwendung von Green-Coding-Praktiken entwickelt wurden (IBM Cloud Education, 2023). Die Green Software Foundation definiert Green Coding als „eine Frage der Codequalität“. Jegliche Design- und Implementierungsentscheidungen, die den ökologischen Fußabdruck positiv beeinflussen, fallen demnach unter Green Coding (Green Software Foundation, 2023a).

Bei näherer Analyse der Definitionen lässt sich feststellen, dass sich unter dem Begriff Green Coding die Menge von Methoden, Aktivitäten und Praktiken zusammenfassen lässt, die zu einer Reduzierung des Stromverbrauchs von Software führen. IONOS führt das Adjektiv „umweltverträglich“ an, wobei die Bedeutung von „umweltverträglich“ nicht klar definiert ist. Dabei umfassen diskutierte Definitionen zum Beispiel nicht, welche Phasen des Softwarelebenszyklus Methoden und Praktiken adressieren sollten, um unter den Begriff Green Coding zu fallen. Der Begriff Coding allein wird häufig als ein Synonym für das Wort Programmieren verwendet. Green Coding umfasst jedoch deutlich mehr als den Akt des Programmierens an sich. Green Coding erstreckt sich über alle Lebensphasen der Softwareentwicklung, wie beispielsweise Design und Betrieb, und nicht nur auf die Implementierung. Diese mangelnde Klarheit kann dazu führen, dass das Konzept nicht eindeutig erfasst und umgesetzt werden kann. Um Green Coding effektiv fördern zu können, ist es notwendig, eine präzisere Definition zu entwickeln, die klare Kriterien beschreibt.

Unklarheit besteht auch hinsichtlich des Ergebnisses, das durch die Anwendung von Green Coding entsteht. Ein mögliches Resultat könnte Green Software sein. Gemäß der Green Software Foundation (GSF) ist Green Software eine Software, die effizient in Bezug auf den Ausstoß von CO₂ ist. Dabei können laut GSF nur drei Aktivitäten die CO₂-Emissionen von Software reduzieren. Dies sind Stromeffizienz, Hardwareeffizienz und Carbon Awareness, wobei Letzteres die Fähigkeit der Software beschreibt, die aktuelle CO₂-Intensität des Stromnetzes zu erfassen und die Ausführung der Software zeitlich und/oder räumlich an sie anzupassen. Das bedeutet beispielsweise, ressourcenintensive Aufgaben in Zeiträume zu verlagern, in denen die Stromproduktion weniger CO₂-intensiv ist (Green Software Foundation, kAc).

Ein weiterer Begriff, der in der Wissenschaft häufiger vorkommt, ist der Begriff der nachhaltigen Software (Sustainable Software). Im Brundtland Report der Vereinten Nationen (UN) (Brundtland, 1987) wird eine nachhaltige Entwicklung als eine solche definiert, „die die Bedürfnisse der gegenwärtigen Generation befriedigt, ohne die Möglichkeiten zukünftiger Generationen zu gefährden, ihre eigenen Bedürfnisse zu befriedigen“.

Bezogen auf Software kann der Begriff nachhaltige Software in zweierlei Hinsicht verstanden werden: (1) als eine Software, die nachhaltige Ziele unterstützt, und (2) als eine Software, die in sich selbst nachhaltig ist, das heißt, die Software, ihre Entwicklung und ihre Nutzung sind nachhaltig (Penzstadler et al., 2014a). Im Rahmen dieses Reports wird der Begriff nachhaltige Software im Sinne der zweiten Definition verstanden.

Nachhaltigkeit adressiert laut Brundtland Report immer drei Dimensionen: (1) die ökologische Dimension, (2) die ökonomische Dimension und (3) die soziale Dimension. Definitionen von nachhaltiger Software greifen diese Dimensionen auf. Dick et al. (2013) definieren nachhaltige Software als „Software, deren Auswirkungen auf Wirtschaft, Gesellschaft, Menschen und Umwelt, die sich aus der Entwicklung, Bereitstellung und Nutzung der Software ergeben, minimal sind und/oder die einen positiven Effekt auf die nachhaltige Entwicklung haben“ (Dick et al., 2013). Gemäß Calero et al. (2021) ist nachhaltige Software „stromeffizient, minimiert die Umweltbelastung der unterstützten Prozesse und hat einen positiven Einfluss auf die soziale und/oder wirtschaftliche Nachhaltigkeit“ (Calero et al., 2021). Im Vergleich dazu adressiert Green Coding maßgeblich die ökologische Dimension. Dies umfasst den Stromverbrauch der Software im Betrieb aufgrund von Hardwarenutzung und die damit einhergehenden Emissionen (operative Emissionen) sowie den Stromverbrauch und dessen Emissionen entlang des gesamten Lebenszyklus von Hardware (eingebettete Emissionen).

Im Kontext des Begriffs Sustainable Software präsentiert sich insbesondere die Arbeit „The GREENSOFT Model: A reference model for green and sustainable software and its engineering“ der deutschen Forschergruppe von Naumann et al. (2011) mit 430 Zitationen laut Google Scholar im Oktober 2024 als eine sehr häufig zitierte Quelle im Bereich der nachhaltigen Software. Eine Analyse ähnlicher wissenschaftlicher Beiträge zeigt, dass diese inhaltlich verwandten Arbeiten maßgeblich zwischen 2009 und 2016 veröffentlicht wurden und sich auf verschiedene Aspekte nachhaltiger Softwareentwicklung konzentrieren.

Ähnlich zeigt sich im Kontext des Begriffs Software Sustainability die Arbeit von Venters et al. „Software sustainability: Research and practice from a software architecture viewpoint“ (Venters et al., 2018) mit 219 Zitationen im Oktober 2024 als ein häufig zitierter Beitrag. Auch hier verdeutlicht eine Betrachtung inhaltlich ähnlicher Arbeiten, dass die Anzahl von Publikationen zum Thema Software Sustainability ab 2009 wesentlich zunahm, wobei die Veröffentlichungen bis 2021 reichen. Dies zeigt eine thematische Entwicklung über mehr als ein Jahrzehnt hinweg, bei der zunehmend Aspekte wie die Langlebigkeit und Wartbarkeit sowie die ökologischen

Auswirkungen von Software adressiert wurden. Doch auch die Einführung der Begriffe Green Software und nachhaltige Software geben keine klare Definition davon, welche Kriterien Software konkret erfüllen muss, um als solche zu gelten. Ein wesentlicher Aspekt ist die Reduktion des Stromverbrauchs durch die Nutzung von Hardwareressourcen wie CPU und Arbeitsspeicher.

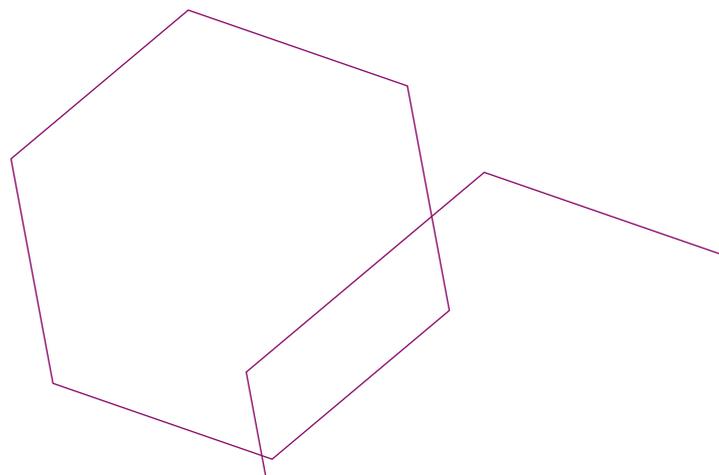
Das Umweltbundesamt hingegen, das die begriffliche Fassung „ressourcen- und energieeffizientes Softwareprodukt“ nutzt, hat über das Umweltzeichen Blauer Engel Kriterien definiert, die ein Softwareprodukt erfüllen muss, um mit dem Umweltzeichen ausgezeichnet zu werden.



Der Blaue Engel für Software (DE-UZ 215)

Der Blaue Engel zeichnet Softwareprodukte aus, die besonders ressourcenschonend und energieeffizient gestaltet sind. Software mit dem Blauen Engel erfüllt hohe Anforderungen an die Umweltverträglichkeit und hilft, den Stromverbrauch im IT-Bereich zu reduzieren. Dazu zählen unter anderem der geringe Ressourcenbedarf sowohl bei der Nutzung als auch bei der Softwareentwicklung, Werbefreiheit und frei verfügbare Updates. Zertifizierte Software reduziert Hardwareanforderungen und verlängert so die Nutzungsdauer von Geräten. Der Blaue Engel ist 06/2024 in der vierten Version erschienen und umfasst seit der aktuellen Version auch Server-Client-Anwendungen.

Durch klare Kriterien und Transparenz fördert der Blaue Engel eine nachhaltige IT-Nutzung und unterstützt Unternehmen sowie die Endnutzerschaft bei der Auswahl umweltfreundlicher Softwarelösungen (Bundesministerium für Umwelt, 2024).



Aber wie weit können der Stromverbrauch und die Nutzung der Hardwareressourcen reduziert werden? Wann ist ein minimaler Ressourceneinsatz für eine maximale Leistungsfähigkeit gegeben? Diese Fragen werden – zumindest aus der akademischen Welt – bisher nicht beantwortet. Auch gibt es keine klare Definition davon, welche Methoden und Praktiken spezifisch unter den Begriff Green Coding fallen.

Für den Kontext dieses Reports definieren wir Green Coding wie folgt:

Green Coding ist definiert als eine Menge von Methoden und Praktiken in der Softwareentwicklung mit dem Ziel, den Stromverbrauch von Software aufgrund der Nutzung von Hardware und die damit einhergehenden operativen und eingebetteten CO₂-Emissionen entlang des gesamten Softwarelebenszyklus zu minimieren. Bei Green Coding geht es demnach um die Reduktion von CO₂-Emissionen durch effiziente Software, das heißt Software, die Hardware und Strom effizient so nutzt, dass möglichst wenig Emissionen erzeugt werden.

2.2 Green Coding in den Phasen des Softwarelebenszyklus

Für die Identifikation von Stromeinsparpotenzialen im Softwarelebenszyklus (Software Development Life Cycle) ist es wesentlich, die einzelnen Phasen zu betrachten.

Abbildung 1 zeigt ein Softwarelebenszyklusmodell mit 6 Phasen: (1) Planung und Analyse, (2) Design, (3) Implementierung, (4) Testen, (5) Deployment und (6) Betrieb und Wartung (in Anlehnung an Balzert, 2011). Wird ein neues Softwareprojekt initiiert, startet der Lebenszyklus mit der Planungs- und Analysephase und durchläuft danach die Phasen (2) bis (6). Wird die Software weiterentwickelt, durchläuft sie wiederholt die Phasen (1) bis (6). Der Zyklus wird dann verlassen, wenn die Software abgeschaltet oder abgelöst wird.

Die Softwareentwicklung ist ein komplexer Prozess. Unabhängig vom Lebenszyklusmodell existieren diverse Softwareentwicklungsmodelle wie Scrum und Kanban als Vertreter agiler Methoden, traditionelle Modelle wie das V-Modell oder hybride Modelle, die unterschiedliche Abläufe definieren. Eine detaillierte Analyse der allgemeinen Softwareentwicklung würde den Rahmen eines Reports sprengen. Im Folgenden werden daher Methoden und Best Practices aufgezeigt, die in der jeweiligen Phase des Softwarelebenszyklus angewendet werden können, um den Stromverbrauch und die Nutzung von Hardwareressourcen durch Software zu reduzieren. Dabei können Methoden und Best Practices häufig auch mehreren Phasen zugeordnet werden. Für die Einschätzung der Einsparpotenziale werden Beispiele mit belegbaren Ergebnissen angeführt.

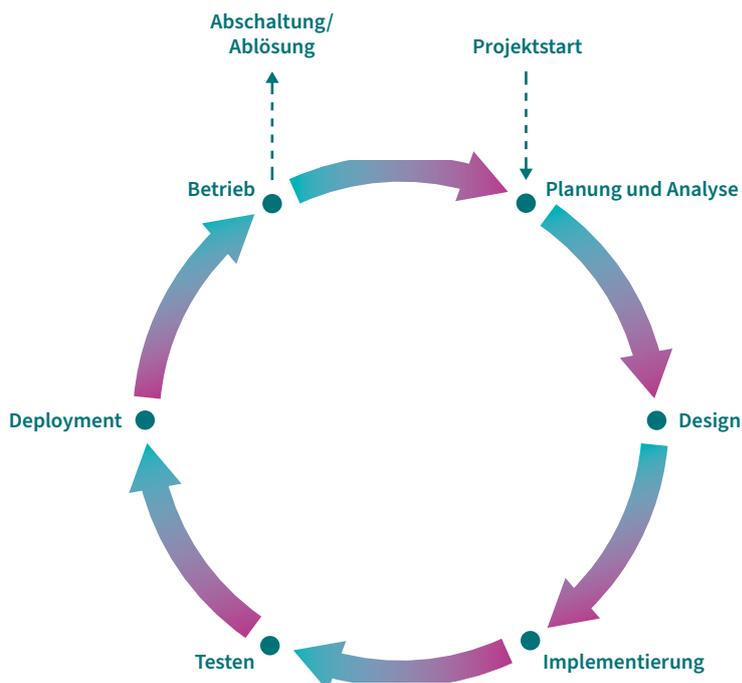


Abbildung 1: Softwarelebenszyklus in Anlehnung an Balzert (2011). Quelle: Balzert (2011)

Planung und Analyse

Die Planungs- und Analysephase umfasst die initiale Planung eines Projekts und die Erhebung der Anforderungen an die zu entwickelnde Software. Bereits in dieser Phase können sich getroffene Entscheidungen auf den Stromverbrauch und die Nutzung von Hardwareressourcen auswirken. Ein Ansatz, Stromeffizienz und Effizienz in Bezug auf die Nutzung von Hardwareressourcen bereits in der Analysephase zu bedenken, ist deren Integration in den Entwicklungsprozess als nichtfunktionale Anforderung. Penzenstadler et al. (Penzenstadler et al., 2014b) vergleichen die Anforderungen an Nachhaltigkeit mit den Anforderungen an Safety und Security. Essenziell dabei ist jedoch, dass die Erfüllung der Anforderungen messbar gemacht und im Entwicklungsprozess kontinuierlich nachverfolgt werden muss. Darüber hinaus können sich die nichtfunktionalen Anforderungen auf eine Vielzahl von Designentscheidungen auswirken, wobei zu diesem Zeitpunkt noch nicht klar ist, wie Entscheidungen getroffen werden müssen, sodass die resultierende Software nachhaltig ist.

Auch die Auswahl der genutzten Programmiersprache ist Teil der Anforderungen an ein Softwareprojekt. Die Energieeffizienz von 27 Programmiersprachen wurde von Pereira et al. (Pereira et al., 2017b) analysiert und verglichen. Bei der Wahl der richtigen Programmiersprache sollten jedoch diverse Faktoren beachtet werden, wie beispielsweise die Expertise der Entwicklerinnen und Entwickler wie auch die Wartbarkeit. Auch kann die Wahl der genutzten Programmiersprache entfallen oder eingeschränkt sein, wenn neue Software in existierende Softwaresysteme integriert wird. Low-Level-Programmiersprachen sind an sich energieeffizienter, jedoch sollte auch der Ressourcenverbrauch von Compiler, Linker und des Bytecode Generator mitberücksichtigt werden. Darüber hinaus ist festzuhalten, dass allein die Wahl der Programmiersprache nicht ausreicht, um Software nachhaltig zu gestalten.

Design

In der Designphase wird die Softwarearchitektur erstellt, indem die Komponenten und ihre Interaktionen basierend auf den Anforderungen definiert werden. Dazu zählen die Struktur der Komponenten sowie ihre Schnittstellen und Datenflüsse. Beim Design einer Webanwendung wird zum Beispiel das System in Frontend, Backend und Datenbank aufgeteilt. Das Design legt fest, wie Benutzeranfragen vom Frontend zum Backend gelangen und wie das Backend auf die Datenbank zugreift, um Produktinformationen bereitzustellen. Die Architektur eines Softwaresystems hat einen direkten Einfluss auf die Langlebigkeit, die effiziente Wartung und die Evolution des Softwaresystems in einer Ausführungsumgebung, die sich kontinuierlich ändert (Kruchten et al., 2006).

Betrachtet man etwa eine Migration in die Cloud, so sollte die Architektur des Softwaresystems so entworfen sein, dass die Software das volle Potenzial der Cloud ausnutzen kann. So kann beispielsweise das Design einer Softwarearchitektur als Microservices-Architektur, in der die Komponenten einzelne Services sind, dieses Ziel unterstützen. Die lose gekoppelten und unabhängig deploybaren und skalierbaren Microservices ermöglichen die dynamische Skalierbarkeit entlang der Nachfrage in einer Cloud-Umgebung (Surianarayanan et al., 2019). Die Cloud-Infrastruktur ermöglicht die automatische Skalierung der Dienste, die eine hohe Last erfahren, und die Abschaltung von ungenutzten Diensten. Bei der Entscheidung sollte jedoch beachtet werden, dass mit dem Einsatz von Microservices eine höhere Netzwerklast einhergehen kann.

Die Entscheidung, eine Software so zu gestalten, dass sie die CO₂-Emissionen in ihrer Ausführung berücksichtigen kann, wird ebenfalls in der Designphase getroffen. Die grundlegende Idee der sogenannten Carbon Awareness ist es, Dienste und Workloads so zu gestalten, dass sie die CO₂-Intensität des aktuellen Strommix in ihrer Ausführung beachten können (Green Software Foundation, kAb). Ist eine Anwendung oder sind ihre Teile carbon-aware, so kann das System selbst oder eine übergeordnete Einheit entscheiden, welche Komponenten wann und in welcher Region ausgeführt werden.



Fallstudie – Carbon Aware Computing

Eine Fallstudie zu Carbon Aware Computing wurde von Microsoft, UBS und WattTime durch die GSF durchgeführt und die Ergebnisse wurden veröffentlicht. UBS entwickelte in Zusammenarbeit mit Microsoft das carbon-aware-sdk. Das carbon-aware-sdk wurde eingesetzt, um Carbon Awareness in die UBS-Risiko-Plattform (ACQA) zu implementieren. Mithilfe des sdk konnte durch die Anbindung von WattTime die CO₂-Intensität des aktuellen Strommix bestimmt und es konnten Workloads innerhalb eines 24-Stunden-Fensters zeitlich versetzt werden. Dieser Ansatz zeigte ein Reduktionspotenzial der SCI (Software Carbon Intensity) von ACQA von ca. 15 % (Buchanan et al., 2023).

Design Patterns sind wiederverwendbare Lösungen für häufig auftretende Probleme im Softwaredesign. Sahin et al. (2012) haben den Energieverbrauch von Anwendungen auf einem eingebetteten System in zwei Varianten verglichen: einmal mit und einmal ohne die Verwendung von Design Patterns. Insgesamt untersuchten sie 15 der 23 von Gamma et al. (Gamma et al., 1994) beschriebenen Design Patterns. Dabei zeigte sich, dass insbesondere die Muster *Flyweight*, *Mediator* und *Proxy* den Stromverbrauch reduzieren können, während das *Decorator-Muster* den Stromverbrauch deutlich erhöht. Das *Flyweight-Muster* reduziert den Speicherbedarf, indem es die Wiederverwendung von Objekten ermöglicht, die sich nur in wenigen Eigenschaften unterscheiden. Das *Mediator-Muster* zentralisiert die Kommunikation zwischen verschiedenen Objekten, was die Anzahl der direkten Verbindungen und damit die Komplexität reduziert. Das *Proxy-Muster* steuert den Zugriff auf ein Objekt und kann dabei auch die Ressourcennutzung optimieren, indem es den Zugriff verzögert oder einschränkt. Im Gegensatz dazu fügt das *Decorator-Muster* dynamisch Funktionalitäten zu einem Objekt hinzu, was die Anzahl der Objekte und damit den Stromverbrauch erhöhen kann.

Webbasierte Softwareanwendungen führen häufig Rechenprozesse auf Serverinfrastrukturen oder in speziellen Rechenzentren durch, wobei die Daten über Schnittstellen wie HTTP (Hypertext Transfer Protocol) übertragen werden. Hier sollte auf eine minimierte Netzwerklast geachtet werden. Um den Ressourcenverbrauch zu reduzieren, kann Edge Computing eingesetzt werden, bei dem mehr Rechenoperationen direkt am Rand des Netzwerks nahe der genutzten Ressource ausgeführt werden (Beck et al., 2016). Eine regelmäßige Überprüfung der verwendeten Bibliotheken und Plugins kann ebenfalls die Datenübertragung über das Internet verringern. Das Caching reduziert die Anzahl der übertragenen Datenpakete und den erforderlichen Stromverbrauch sowohl auf der Geräte- als auch auf der Serverseite. Zusätzliche Vorteile sind kürzere Ladezeiten und ein stabileres Verhalten bei hohen Lastspitzen (Erman et al., 2011). Eine Case Study der Green Coding Solutions, die den Stromverbrauch von Wordpress und HUGO verglichen hat, konnte zeigen, dass für Webseiten mit viel Idle-Time eine verhältnismäßig große Menge an Strom im Idle-Modus verbraucht wird. Folglich sollte die Webanwendung in diesem Fall nicht für den aktiven Modus, sondern für den Idle-Zustand optimiert und beispielsweise ein Hosting mit einer dynamischen Skalierung auf Null gewählt werden (Tarara, 2022).

Implementierung

Der Bitkom-Leitfaden für ressourceneffiziente Programmierung fasst eine Menge von Aspekten zusammen, die für die Entwicklung von nachhaltiger Software zu beachten sind. Dazu zählen beispielsweise Abwärtskompatibilität, Wartbarkeit, Auswahl energieeffizienter Modi, Standard-Nachhaltigkeitssettings und Transparenz (Bitkom, 2021). Lieder identifiziert auf einem Blog des Unternehmens adesso (Lieder, 2024) weitere Sammlungen von Methoden und Best Practices wie das Handbook for Sustainable Design of Digital Services (Institute for Sustainable

IT, 2021), die W3C Web Sustainability Guidelines (Sustainable Web Design, 2024), den Kriterienkatalog des Umweltcampus Birkenfeld (Hilty et al., 2017) und das Tactics-Archiv der VU Amsterdam (Lago, kA). Ferner zeigt sich beispielsweise, dass die Auswahl von effizienten Datenstrukturen und Algorithmen einen maßgeblichen Faktor bei der Energieeffizienz und Hardwarenutzung von Software spielt (MoldStud, 2024). So können Tools wie GreenC5 (Michanan et al., 2017) die optimale, energieeffiziente, dynamische Datenstruktur in der C5 Collection für C# auswählen. Die Experimente zeigen, dass die Mediane der Strom einsparung auf zwei Computern 61 % und 60 % betragen. Das Rahmenwerk SEEDS (Manotas et al., 2014) optimiert automatisch Java-Anwendungen durch die Auswahl der energieeffizientesten Implementierungen von Java Collection APIs und konnte in ersten Versuchen Stromeinsparungen von bis zu 96 % zeigen. Inwiefern sich die Ergebnisse auf komplexe Anwendungen übertragen lassen, muss jedoch noch evaluiert werden.

Source Code Analysis beschreibt den Prozess der Analyse von Code, um mögliche Schwachstellen im Code vor dem Deployment zu identifizieren. Im Kontext der Ressourceneffizienz sollte der Code beispielsweise auf unerreichbaren Code und ungenutzte Variablen und Funktionen hin untersucht werden, da sie unnötigerweise Ressourcen in Anspruch nehmen. Pereira et al. (Pereira et al., 2017a) zeigten in ihrer Studie, dass ihr Ansatz, der Ineffizienzen im Code identifiziert, den Stromverbrauch im Durchschnitt um 18 % senken kann. Das ecoCode-Projekt (Le Goer, 2023) stellt eine Liste von statischen Code Analyzern zur Verfügung, um Bereiche im Code zu identifizieren, die negative ökologische Auswirkungen haben, zum Beispiel hinsichtlich des Stromverbrauchs und der Hardwarenutzung. Der Einsatz von CAST Highlight, einer Software, um Code zu identifizieren, der Verbesserungspotenzial in Bezug auf den Stromverbrauch und die Hardwareressourceneffizienz aufweist, und diesen Code korrigiert, konnte in einem Szenario eine 5 %ige Einsparung der Laufzeit ermöglichen. Darüber hinaus zeigte die Software ein Einsparpotenzial von 400 kg CO₂-Emissionen und eine mögliche Reduktion des Stromverbrauchs von über 1.000 kWh pro Jahr auf (Green Software Foundation, 2024).

Testen und Deployment

Software wird häufig in mehreren Umgebungen bereitgestellt, zum Beispiel in einer Entwicklungsumgebung, einer Testumgebung und einer Produktivumgebung. Die Entwicklungsumgebung dient dazu, das Produktivsystem zu imitieren und eine Sandbox für die Entwicklung und Experimente zur Verfügung zu stellen. Die Testumgebung sollte das Zielsystem so gut wie möglich nachbilden, um Software ausführlich zu testen. Erst wenn die Software den geforderten Qualitätsansprüchen genügt, wird sie in der Produktivumgebung angewendet. Die Anzahl und das Ausmaß der Umgebungen haben einen nicht unwesentlichen Einfluss auf den Stromverbrauch und die Nutzung der Hardwareressourcen in der Softwareentwicklung. Sinnvoll ist hier beispielsweise die Automatisierung von Monitoring Dashboards, um den Stromverbrauch und die Nutzung von Hardwareressourcen zu überwachen und Schwachstellen zu identifizieren.

Mair (2023) führt ebenfalls die Optimierung von Testumgebungen an. Demnach sollten nur Tests ausgeführt werden, die von einer Code-Änderung betroffen sind. Auch sollte nur bei Bedarf und nicht automatisiert zu festen Zeiten getestet und darauf geachtet werden, dass die Ausführung von Tests bestmöglich parallelisiert werden kann, ohne Strom und Hardwareressourcen durch Wartezeiten zu nutzen (Mair, 2023). ECO CI ist zum Beispiel eine Sammlung von Tools, um den Stromverbrauch und die CO₂-Emissionen von CI-Pipelines zu messen. Der ECO CI Activity Checker überspringt Testläufe, wenn für automatisierte Workflows kein Commit stattgefunden hat oder ein manueller Lauf gestartet wurde.

Betrieb

Insbesondere der Betrieb von Software kann einen hohen Strom- und Ressourcenbedarf haben. Eine Wahl, die Unternehmen treffen können, ist die Migration in die Cloud. Cloud-Lösungen bieten skalierbare und flexible Möglichkeiten, um Anwendungen in der Cloud zu betreiben. Insbesondere sogenannte *Hyperscaler* ermöglichen einen energieeffizienten Betrieb im Vergleich zu On-Premises-Lösungen. Die Energieeffizienz eines Anbieters wird mithilfe der Power Usage Effectiveness (PUE) angegeben, wobei ein idealer Wert bei 1,0 liegt. Dies bedeutet, dass die gesamte Energie für die eigentliche IT-Infrastruktur genutzt wird, ohne Verluste durch Kühlung oder andere Systeme. Im Jahr 2022 erreichte die Google Cloud eine PUE von 1,12, was nahe am idealen Wert liegt und folglich eine hohe Energieeffizienz darstellt. Im Vergleich dazu weisen traditionelle Hosting-Lösungen häufig einen deutlich höheren PUE von ca. 2,0 auf (Jones, 2018). Darüber hinaus können Informationen über die Verwendung erneuerbarer Energien und die Abwärmenutzung sowie Transparenz hinsichtlich der Emissionen als Faktoren für die Auswahl betrachtet werden.

In der Cloud können unterschiedliche Praktiken angewendet werden, um den Stromverbrauch zu reduzieren und die Hardwarenutzung zu optimieren. Beispiele dafür sind die dynamische Skalierung, bei der Ressourcen automatisch an den aktuellen Bedarf angepasst werden, und das Abschalten ungenutzter Instanzen, um Strom zu sparen. Auch die Nutzung von stromeffizienten Hardwarekomponenten und das gezielte Platzieren von Workloads in Rechenzentren mit erneuerbaren Energien zählen dazu. Die Green Software Foundation hat dazu einen Katalog mit Green Software Patterns für die Cloud veröffentlicht (Green Software Foundation, kA), der kontinuierlich erweitert wird.



Use Case – Etsy Cloud Migration

Die ersten elf Jahre betrieb Etsy (etsy.com) seine Plattform in einem Co-Location-Rechenzentrum. Mit zunehmendem Wachstum von Etsy lernte das Unternehmen, dass die Bereitstellung der Software in der Cloud schneller vorstättenging als der Ausbau des Co-Location-Rechenzentrums. Etsy schätzte, dass die Erweiterung um 150 Server Monate dauern würde, während die Nutzung von Cloud-Ressourcen 4 Minuten erforderte. Etsy migrierte in die Cloud und konnte eine geschätzte Stromersparung von 13 % (von 7.330 MWh in 2018 auf 6.376 MWh in 2019) erreichen (Hollier et al., 2020).

Ein Ansatz, der nicht auf technische Emissionseffizienz abzielt, sondern auf das Emissionsbewusstsein, ist das sogenannte Demand Shaping. Die grundlegende Idee dieses Ansatzes ist die Berücksichtigung des aktuell verfügbaren grünen Stroms im Stromnetz. Wenn der Anteil erneuerbarer Energien hoch ist, werden eine größere Anzahl an Rechenoperationen oder Rechenoperationen schneller ausgeführt, da sowohl der Stromverbrauch der Geräte als auch der Stromverbrauch der für die Datenübertragung erforderlichen Netzwerke zu diesem Zeitpunkt geringere Emissionen verursachen. Ein bekanntes Beispiel für Demand Shaping ist die automatische Anpassung von Videoplayern im Streaming-Bereich, bei der die Anpassung der Qualität aufgrund einer schlechten Netzwerkverbindung der Endnutzerin bzw. des Endnutzers vorgenommen wird. Ähnlich kann die Datenübertragung angepasst werden, wenn die CO₂-Intensität des Strommix hoch ist. Somit wird der Dienst erbracht, jedoch in potenziell eingeschränkter Qualität. Die verfügbaren Ressourcen werden entsprechend nur zum Teil ausgeschöpft.

Im Kontext von Demand Shaping spielt das Prinzip der Suffizienz eine wichtige Rolle. Es beschreibt die bewusste Entscheidung, den Ressourcenverbrauch durch Anpassung und Reduzierung des Bedarfs zu verringern, anstatt sich ausschließlich auf Effizienzsteigerungen zu verlassen. Dadurch wird der Verbrauch auf das Notwendige beschränkt, was zur langfristigen Schonung der Ressourcen beiträgt.

Im Kontext von *Demand Shifting* kann eine zeitliche oder eine räumliche Verschiebung vorgenommen werden, wenn Software carbon-aware ist, wie bereits im Abschnitt „Design“ beschrieben. Bei einer zeitlichen Verschiebung werden die Rechenoperationen in ein Zeitfenster verlegt, in dem die durch Stromnutzung verursachten Emissionen geringer sind. Bei einer räumlichen Verschiebung wird die Rechenoperation in ein Rechenzentrum in einer anderen Region verschoben, in der emissionsärmerer Strom im Strommix verfügbar ist.



Use Cases – Carbon-Awareness bei Google und Microsoft

Ein Use Cases im Kontext von Carbon Awareness und Demand Shifting sind die Google Carbon Aware Data Centers (Koningstein, 2021). Der Ansatz basiert auf der Idee, die CO₂-Intensität für den nächsten Tag vorherzusagen und die anfallenden Workloads entsprechend zu modellieren und auszuführen. Radovanovic et al. (2022) konnten zeigen, dass der durchschnittliche Strombedarf eines Clusters um 1 % bis 2 % reduziert werden kann, wenn aufgrund der Vorhersage die Workloads in einen anderen Zeitraum verschoben werden.

Ein weiterer Use Case ist das Carbon Aware Windows von Microsoft. Die grundlegende Idee dieses Ansatzes ist die Durchführung von Windows-Updates in diversen Ländern zu Zeiten, wenn die CO₂-Intensität in der jeweiligen Region gering ist (Weatherbed, 2022).

Die angeführten Beispiele von Green Coding in den unterschiedlichen Phasen des Softwarelebenszyklus zeigen, dass es in jeder Phase Maßnahmen gibt, die zur Reduktion des Stromverbrauchs und der Nutzung von Hardwareressourcen durch Software beitragen können. Die Entscheidungen, die in der Planungs- und Analysephase sowie in der Designphase getroffen werden, haben langfristig eine Auswirkung auf den Stromverbrauch von und die Hardwarenutzung durch Software. Bei der Planung und der Erhebung der Anforderungen an eine Software sollten Energieeffizienz und Hardwarenutzung direkt mitbedacht werden, da die Entscheidungen in der

Designphase auf die Erfüllung der identifizierten Anforderungen abzielen und sich auf alle späteren Phasen auswirken. Ist zum Beispiel die Architektur einer Software nur unzureichend für den Einsatz in einer Cloud oder generell auf ein dynamisch skalierbares System ausgelegt, können die Potenziale in einem solchen System in Bezug auf Stromeffizienz und Hardwarenutzung in der später gelagerten Betriebsphase nicht voll ausgeschöpft werden.

Die analysierten Quellen zeigen auch, dass es noch sehr wenige Daten in Bezug auf erfolgreiche Anwendungsfälle von Green Coding gibt. Ein Grund dafür könnte sein, dass das Thema noch relativ neu und/oder technisch anspruchsvoll ist und demnach noch nicht in der Breite angewendet wird. Die analysierten Einträge auf Unternehmens-Websites, die das Thema adressieren, sind ebenfalls relativ jung, was die These der Neuartigkeit untermauert. Es gibt jedoch Vorreiter in Bezug auf Green Coding, wie die Case Study von UBS sowie die Anwendungsbeispiele von Google und Microsoft zeigen. Auch die Literaturrecherche zeigte keine Ergebnisse, die das Potenzial von Green Coding in Frage stellen. Eine mögliche Schlussfolgerung wäre, dass die Industrie das Thema Green Coding zwar aufgegriffen, aber in der Praxis noch nicht ausreichend umgesetzt hat, um messbare Erfolge zu erzielen und zu publizieren. Es könnte auch sein, dass die theoretischen Einsparpotenziale in der praktischen Anwendung nicht immer realisiert werden, da gewisse Komplexitäten nicht berücksichtigt wurden oder Kompromisse eingegangen werden mussten. Diese Beobachtungen unterstreichen die Notwendigkeit, Green Coding bereits in die Anforderungsanalyse zu integrieren, um sicherzustellen, dass die Einsparpotenziale tatsächlich erreicht werden. Trotz der Komplexität kann die frühzeitige Implementierung von Green Coding von entscheidender Bedeutung sein, da sie langfristig erhebliche ökologische Vorteile bieten kann.

2.3 Herausforderungen bei der Anwendung von Green Coding

Die Diskussion der Methoden und Praktiken in den verschiedenen Phasen des Softwarelebenszyklus geben einen Ausblick auf die möglichen Potenziale zur Einsparung von Strom und damit einhergehenden operativen und eingebetteten Emissionen. Dennoch lässt die relativ geringe Anzahl an erfolgreichen Anwendungsfällen und Fallstudien vermuten, dass die praktische Anwendung von Green Coding im Softwarelebenszyklus mit Herausforderungen verbunden ist. Im Folgenden werden die zentralen Herausforderungen beleuchtet, die die Anwendung von Green Coding entlang des Softwarelebenszyklus erschweren.

Im Jahr 2023 initiierte die Green Software Foundation eine globale Umfrage zum Stand von Green Software, bei der 2.191 internationale Expertinnen und Experten aus der Software-

entwicklung befragt wurden. Darüber hinaus führte sie eine Sekundärforschung auf Basis von Veröffentlichungen von namhaften Beratungsunternehmen wie McKinsey und Gartner, kombiniert mit dem Wissen aus wissenschaftlichen Beiträgen und Experten-Interviews, durch und verband die Erkenntnisse mit den erhobenen Daten (Green Software Foundation, 2023b). Die Ergebnisse lassen schlussfolgern, dass umweltfreundliches Entwickeln häufig aus intrinsischer Eigenmotivation geschieht und ökologische Nachhaltigkeit oft gegenüber kurzfristiger Leistung und kurzfristigen Kosten zurückstehen muss. Auch konnte mit Daten belegt werden, dass es bereits Unternehmen gibt, die die Umwelteinflüsse ihrer Software erfassen. 15,6 % der Befragten gaben an, dass das Unternehmen, für das sie arbeiten, die CO₂-Emissionen seiner Software kontinuierlich erfasst, während 43,9 % nur sporadische Messungen

durchführen. Ein weiterer Aspekt des Reports ist die Wichtigkeit von Training und Bewusstseinsbildung für die Umsetzung von Green Coding. Die aktuelle Ausbildung von Expertinnen und Experten, die in der Softwareentwicklung beschäftigt sind, adressiert das Thema Green Coding nur selten (Junger et al., 2023).

Ähnliche Schlussfolgerungen lassen sich auch aus den Ergebnissen des Workshops „Bringing Green Coding to Education – How can we educate developers?“ ableiten, den die Autorinnen und Autoren dieses Reports auf der EcoCompute-Konferenz (www.eco-compute.io/) im April 2024 durchgeführt haben.

Unter den 28 teilnehmenden Vertreterinnen und Vertretern namhafter Unternehmen bestand Einigkeit, dass Bewusstseinsbildung und Training wesentliche Aspekte sind, um das Thema Green Coding in die Breite zu tragen. Darüber hinaus sei es aktuell sehr schwierig, das Thema Green Software bzw. Green Coding im Unternehmen zu verankern. Als ein elementarer Grund hierfür wurde genannt, dass Unternehmen das Thema nicht adressieren. Die Nachhaltigkeit von Software spiele im Unternehmen keine Rolle. Demnach existierten auch keine KPIs (Key Performance Indicators), um Softwarenachhaltigkeit zu erfassen und zu verfolgen. Ähnlich wie bei den Ergebnissen der Studie der Green Software Foundation sprachen sich die Teilnehmerinnen und Teilnehmer für mehr Regulierung bezüglich ökologisch nachhaltiger Softwareentwicklung aus. Sie schlugen beispielsweise vor, existierende Regularien spezifisch auf IT herunterzubrechen und Verantwortlichkeiten für CO₂-Emissionen im Lebenszyklus zu definieren.

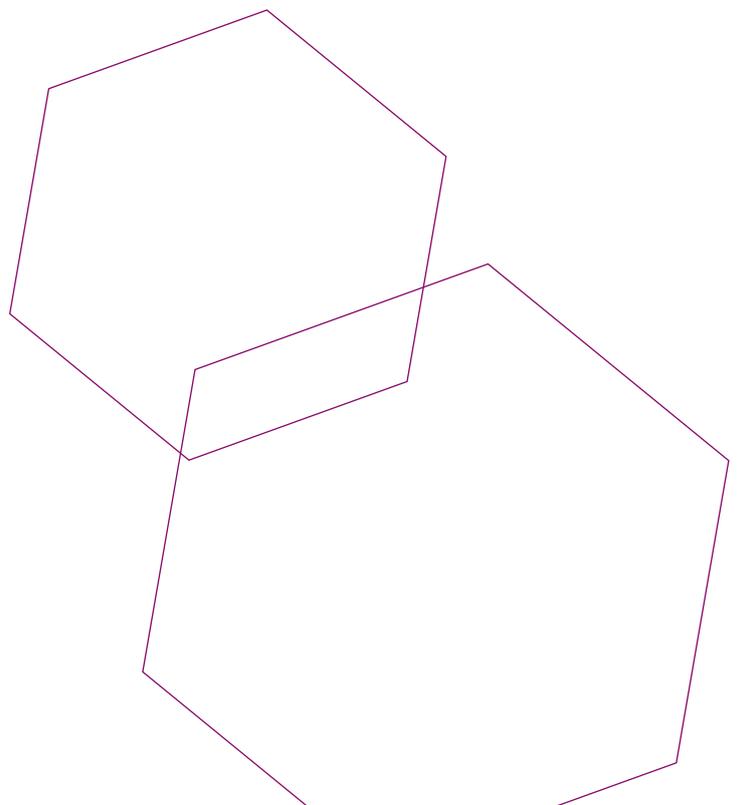
Zusammenfassend lässt sich feststellen, dass der Begriff Green Coding sowohl in wissenschaftlichen Veröffentlichungen als auch in der Unternehmenspraxis präsent ist, jedoch keine einheitliche Definition dafür existiert. Die Literaturrecherche zeigt, dass bestehende Definitionen nicht eindeutig sind, sodass unklar bleibt, welche Maßnahmen tatsächlich unter den Begriff Green Coding fallen. Ebenso ist das Ergebnis von Green Coding unklar. Die Green Software Foundation nennt beispielsweise Green Software als Ergebnis von Green Coding. Welche Kriterien Software erfüllen muss, um als Green Software bezeichnet werden zu können, ist jedoch nicht ausreichend definiert.

Die geringe Anzahl publizierter Fallstudien deutet darauf hin, dass Green Coding bislang wenig Anwendung und Verbreitung gefunden hat. Trotz der zunehmenden Relevanz der Thematik im Kontext der Klimakrise konnten nur wenige fundierte empirische Daten und konkrete Handlungsempfehlungen identifiziert werden. Eine Analyse der Herausforderungen legt nahe, dass die fehlende Anwendung darauf zurückzuführen sein könnte, dass IT und somit auch Software in Unternehmen meist nicht im Fokus von Nachhaltigkeitsanstrengungen stehen und keine Metriken zur Erfassung ihrer Nachhaltigkeit existieren. Insbesondere fehlt es aber auch an Bewusstsein für die Bedeutung und die Auswirkungen von Green Coding, was die Einführung entsprechender Maßnahmen in Unternehmen erschwert.

Im Spezifischen wurden auf Basis der Literaturrecherche die folgenden Forschungsfragen identifiziert, die im Rahmen des vorliegenden Reports beantwortet werden sollen.

- 1. Welche Herausforderungen erschweren die umfassende Einführung von Green Coding in der Softwareentwicklungsindustrie?**
- 2. Welche Stakeholder sind relevant bei der Implementierung von Green Coding?**
- 3. Welche konkreten Maßnahmen sollten relevante Stakeholder treffen, um Green Coding in die breite Anwendung zu bringen?**

Für die Beantwortung der Forschungsfragen wurden im Rahmen des Reports zwei Umfragen konzipiert und durchgeführt, die im Folgenden beschrieben werden.





3.

Umfragen

Das Ziel der Umfragen war es, den aktuellen Wissensstand in Bezug auf Green Coding zu erfassen. Dazu zählen unter anderem das Verständnis des Begriffs, der Stand der Umsetzung in der Praxis und eventuelle Hindernisse oder Hürden, die die Umsetzung verhindern bzw. erschweren. Die Ergebnisse der Umfragen dienten als Basis für die gezielte Ableitung von Handlungsempfehlungen für relevante Stakeholder.

3.1 Entwicklung der Umfragen

Zur Beantwortung der Forschungsfragen wurde zum einen eine Umfrage zur Befragung von Expertinnen und Experten entwickelt. Ziel der Expertenbefragung war es, das spezialisierte Fachwissen und tiefgehende Verständnis des Themengebiets der Teilnehmerinnen und Teilnehmer zu nutzen. Ergänzend wurde eine offene Umfrage aufgesetzt, um den allgemeinen Wissensstand im Themengebiet zu erfassen. Dadurch wurde das Gesamtbild der Untersuchung erweitert und vertieft.

Für die Expertenbefragung wurde eine Umfrage mit 19 Fragen erstellt. Für die öffentlich zugängliche Umfrage wurden 17 Fragen ausgewählt. Die Fragen wurden jeweils auf die Zielgruppe – Expertinnen und Experten oder Nichtexpertinnen und -experten – der jeweiligen Umfrage zugeschnitten.

Die Entwicklung der Fragen erfolgte in mehreren Schritten. Zunächst wurde in einem Workshop mit Vertreterinnen und Vertretern aus Wirtschaft, Politik und Nichtregierungsorganisationen ein umfangreicher Fragenkatalog erarbeitet. Aus der Sammlung der Fragen wurden gezielt diejenigen Fragen ausgewählt, deren Beantwortung dazu beiträgt, die Forschungsfragen zu beantworten. Durch die Auswahl sollte sichergestellt werden, dass die Ergebnisse der Umfrage effektiv zur Beantwortung der zentralen Fragen herangezogen werden können. Die ausgewählten Fragen wurden auf der EcoCompute-Konferenz, einer Veranstaltung zur Förderung nachhaltiger und umweltfreundlicher Technologien in der IT-Branche, in einem Workshop vorgestellt, diskutiert und weiter verfeinert. Die überarbeitete Fragensammlung wurde dem Konsortium vorgestellt und das Feedback wurde eingearbeitet.

3.2 Befragung von Expertinnen und Experten und offene Umfrage

Die offene Umfrage wurde am 22. Mai 2024 über mehrere Netzwerke verteilt, darunter das der Green Coding Solutions GmbH, das Bits&Bäume-Netzwerk, das Netzwerk der EcoCompute-Konferenz, das Netzwerk der ClimateActionTech Community sowie der LinkedIn-Kanal des Future Energy Lab der dena. Der Link zur Expertenbefragung wurde am selben Tag direkt an die ausgewählten Expertinnen und Experten versendet. Beide Umfragen waren für einen Zeitraum von zwei Wochen (bis zum 5. Juni 2024) online zugänglich.

Zusätzlich zu der Expertenbefragung und der offenen Umfrage wurden Interviews mit fünf Expertinnen und Experten aus den Bereichen Technik, NGOs, Politik, Wirtschaft und Wissenschaft geführt. Diese Interviews dienten dazu, tiefere Einblicke zu gewinnen und zusätzliche Informationen zu sammeln, um realistische und praxisnahe Handlungsempfehlungen zu entwickeln. In diesen Gesprächen wurden dieselben Fragen gestellt wie in der Expertenbefragung, sodass die Ergebnisse der Interviews gemeinsam mit den Daten aus der Expertenbefragung ausgewertet werden konnten.

Die Expertenbefragung wie auch die offene Umfrage umfassten zwei verschiedene Fragetypen:

- 1. Multiple-Choice-Fragen mit Freitext-Option:** Zu den Fragen wurden den Teilnehmerinnen und Teilnehmern vorgegebene Antwortmöglichkeiten angeboten, von denen jeweils nur eine ausgewählt werden konnte. Eine Ausnahme bildete eine Option mit Freitext-Feld, bei der zusätzliche Informationen oder Kommentare hinzugefügt werden konnten. Diese Struktur ermöglichte eine Balance zwischen standardisierten Antworten und individueller Flexibilität.
- 2. Freitext-Fragen:** Um die Thematik eingehend zu erfassen, wurden Fragen als offene Freitext-Fragen gestaltet, die anschließend manuell kategorisiert wurden. Dieser Ansatz gewährleistete, dass die Befragten unvoreingenommen antworten konnten.

Auswahl des Umfrage-Tools

Für die Erhebung der Daten wurde LimeSurvey (www.limesurvey.org/de) ausgewählt, ein Tool, das sowohl offene als auch zugangsbasierte Fragen mit Einladung unterstützt und eine effiziente Datenauswertung ermöglicht. LimeSurvey wurde aufgrund seiner Flexibilität, der hohen Datenschutzstandards – inklusive deutschen Hosting-Standorts – und der Benutzerfreundlichkeit ausgewählt, um eine möglichst breite Beteiligung sicherzustellen.

Auswertung der Antworten

Die Erhebung bestand aus drei Teilen:

- 1. Offene Umfrage:** Die offene Umfrage bestand aus 17 Fragen (OU1 bis OU17). Insgesamt wurden 109 vollständige Antworten von den Teilnehmerinnen und Teilnehmern verzeichnet. Es wurden 94 Umfragen in Deutsch und 15 in Englisch ausgefüllt.
- 2. Expertenbefragung:** Die Umfrage für die Expertinnen und Experten bestand aus den 19 Fragen (EU1 bis EU19). Es wurden 27 vollständige Antworten von Expertinnen und Experten aus verschiedenen Bereichen, darunter akademische Bildung/Forschung, Privatwirtschaft, Beratung, Entwicklung, Zivilgesellschaft und Politik, verzeichnet. 23 Umfragen wurden in Deutsch und 4 in Englisch ausgefüllt.
- 3. Persönliche Interviews:** Diese Befragung bestand aus einstündigen persönlichen Interviews mit fünf Expertinnen und Experten, darunter jeweils eine Vertreterin oder ein Vertreter aus dem Bereich Technik, Wirtschaft, Wissenschaft, Politik und NGOs. Die Befragung umfasste alle Fragen, die auch in der Expertenbefragung zum Einsatz kamen. Die Ergebnisse dieser Gespräche sind als Datenpunkte in die Expertenbefragung mit eingeflossen.

Sowohl die offene Umfrage als auch die Expertenbefragung und die persönlichen Interviews wurden in einem Zeitraum von zwei Wochen durchgeführt. Unvollständige Einreichungen wurden von der Analyse ausgenommen, um die Datenqualität zu gewährleisten.

Unabhängigkeitsbetrachtung und Daten-Anonymisierung

Ein Teil der Expertinnen und Experten, die an der Expertenbefragung teilgenommen haben, stammt aus dem Konsortium. Die Aussagen und Meinungen der Mitglieder des Konsortiums wurden dabei nicht anders gewichtet als die der weiteren befragten Personen. Die Gleichbehandlung der Antworten wurde durch ihre Anonymisierung sichergestellt. Die Auswertung der Fragen wurde ausschließlich durch die Green Coding Solutions GmbH durchgeführt, die selbst nicht an der Umfrage teilnahm.

Kategorisierung der Freitext-Antworten

Die Kategorisierung der Freitext-Antworten erfolgte durch die Zuordnung jeder Antwort zu einer einzigen Hauptkategorie, um eine klare und fokussierte Analyse sicherzustellen. Die Methode der Zusammenfassung zu einer Hauptkategorie ermöglicht eine systematische und konsistente Strukturierung der Daten, indem der dominante Aspekt jeder Antwort hervorgehoben wird. Sie bietet den Vorteil der methodischen Vereinfachung und fördert die Eindeutigkeit der Ergebnisse. Allerdings ist dieses Vorgehen auch mit Herausforderungen verbunden, da die Komplexität und Mehrdimensionalität vieler Antworten reduziert werden können. Zudem besteht die Gefahr, dass relevante sekundäre Themen übersehen und subjektive Entscheidungen bei der Auswahl der Hauptkategorie getroffen werden, was potenziell den Verlust von Nuancen in den Daten zur Folge hat. Um diese Risiken zu minimieren, wurden die Kategorien mehrfach überarbeitet und angepasst, um sicherzustellen, dass die Antworten präzise wiedergespiegelt werden. Darüber hinaus wurde eine regelmäßige Überprüfung durch mehrere Codierer durchgeführt, um die Zuverlässigkeit und Konsistenz der Zuordnungen zu erhöhen.

Das methodische Vorgehen der Kategorisierung wurde bei allen Freitext-Fragen angewendet. Konnten Antworten zu einer eindeutigen Hauptkategorie unter einem Namen zusammengefasst werden, wurde die Kategorie mit dem entsprechenden Namen benannt. Konnte ein solcher Oberbegriff nicht eindeutig vergeben werden, so wurde die Kategorie mit der Kombination der identifizierten Begriffe – separiert durch einen Schrägstrich (/) – benannt. So wurden beispielsweise bei einer Frage als Antwort die Begriffe „nachhaltig“, „ressourcenschonend“ und „ökologisch“ genannt. Diese Antworten wurden in einer Kategorie zusammengefasst, die den Namen „Nachhaltig/Ressourcenschonend/Ökologisch“ trägt, da kein eindeutiger Oberbegriff identifiziert werden konnte. Zu beachten ist, dass Antworten auch mehreren Kategorien gleichzeitig zugeordnet werden konnten, wenn dies einen Mehrwert im Erkenntnisgewinn darstellte. Das Vorkommen wurde dann für beide Kategorien gezählt.

Limitationen

Obwohl dieser Report wertvolle Erkenntnisse über die Möglichkeiten und Herausforderungen von Green Coding liefert, sind einige Limitationen zu berücksichtigen. Die befragten Expertinnen und Experten stammen größtenteils aus dem bestehenden Netzwerk der Forscherinnen und Forscher, was zu einer Verzerrung (Bias) der Ergebnisse führen könnte. Darüber hinaus wäre eine größere Anzahl an Befragten wünschenswert gewesen, um ein breiteres Spektrum an Perspektiven zu erfassen. Ferner bezieht sich die Untersuchung primär auf den deutschen Kontext. Eine Ausweitung des Reports auf europäische oder globale Ebene könnte zusätzliche Erkenntnisse liefern und die Generalisierbarkeit der Ergebnisse verbessern. Zudem könnten alternative Methoden zur Auswertung der Umfragedaten in zukünftigen Untersuchungen genutzt werden, um noch präzisere Ergebnisse zu erzielen.

3.3 Auswertung der Umfrageergebnisse

Im Folgenden werden die Antworten auf die Fragen aus der Umfrage ausgewertet. Alle Fragen der offenen Umfrage (OU1 bis OU17) und der Expertenumfrage (EU1 bis EU19) wurden ausgewertet. In diesem Report werden die Ergebnisse der offenen Umfrage zu den Fragen OU2, OU3, OU5, OU6, OU7, OU8, OU13 und OU14 und aus der Expertenumfrage die Ergebnisse zu den Fragen EU2, EU3, EU5, EU6, EU7, EU8, EU9, EU10, EU13 und EU14 dargestellt und diskutiert. Die Diskussion der nicht genannten Fragen wurde aus den folgenden Gründen nicht in den Report integriert. (1) Zu geringe Datenbasis: Auch wenn eine Umfrage an sich vollständig ausgefüllt wurde, konnte im Verlauf der Fragen die Tendenz erkannt werden, dass die Antworten von Teilnehmerinnen und Teilnehmern zunehmend kürzer wurden oder auch einmal eine Frage ausgelassen wurde. Bei einer Menge von weniger als 10 verwertbaren Antworten wurde die Frage exkludiert. (2) Überschneidung mit anderen Fragen: Es zeigte sich, dass sich einige Fragen mit anderen Fragen der Umfrage überschneiden haben bzw. dass die Antworten der Teilnehmerinnen und Teilnehmer darauf hinwiesen, dass dies so wahrgenommen wurde. (3) Strategische Auswahl für die Analyse: Sowohl die offene Umfrage als auch die Expertenumfrage waren so konzipiert, dass neue Erkenntnisse gewonnen werden sollten. Das Ergebnis der Umfrage war demnach offen und nicht vorhersehbar. Es wurde daher auf Fragen fokussiert, bei denen die Antworten für die Beantwortung der Forschungsfragen herangezogen werden konnten. Eine vollständige Übersicht über alle Fragen der offenen Umfrage und der Expertenumfrage inklusive aller Antwortmöglichkeiten bei Multiple-Choice-Fragen ist im Anhang zu finden.

Begriff Green Coding

Frage: Wie gut sind Sie mit dem Thema Green Coding vertraut? (EU2, OU2)

Typ: Multiple-Choice-Frage

Zielgruppe: Expertinnen und Experten sowie offene Umfrage

Diese initiale Fragestellung sollte klären, inwieweit „Green Coding“ als Begriff in den verschiedenen Umfeldern etabliert ist.

Während sich ein Großteil der Expertinnen und Experten bereits aktiv mit dem Thema Green Coding beschäftigt hat, zeigt sich, dass fast ein Fünftel der Befragten der offenen Umfrage den Begriff noch nie gehört hat. Zudem haben fast 25 % der Teilnehmerinnen und Teilnehmer schon von Green Coding gehört, sind aber nicht ausreichend vertraut, um ein Beispiel nennen zu können.

Verständnis von Green Coding

Frage: Was verstehen Sie unter Green Coding? (EU3, OU3)

Typ: Freitext-Frage

Zielgruppe: Expertinnen und Experten sowie offene Umfrage

Diese Frage hatte das Ziel, zu ermitteln, ob es ein einheitliches Verständnis des Begriffs gibt und, falls ja, welches dies ist. Sie wurde nur den Teilnehmerinnen und Teilnehmern gestellt, die in der vorangegangenen Frage angegeben hatten, mit dem Thema Green Coding vertraut zu sein. Den Befragten stand ein offenes Textfeld zur Verfügung, in dem Antworten eingetragen werden konnten. Die folgenden Hauptkategorien wurden identifiziert: (1) Technische Lösungen, (2) Ökologisch/Ressourcenschonend/Nachhaltig, (3) Organisatorische/Management-Lösungen und (4) Alternative Lösungen. Jede der Kategorien enthält die genannten Begriffe, die zur Hauptkategorie zusammengefasst wurden. Darüber hinaus geben die Klammern hinter den einzelnen Begriffen den prozentualen Anteil der Antworten der Expertinnen und Experten sowie der Befragten der offenen Umfrage an, die dieser Kategorie zuzuordnen waren.

Technische Lösungen (100 % Expertenbefragung/95 % offene Umfrage):

Ein Großteil beider Befragtengruppen versteht unter Green Coding technische Lösungen. Insbesondere wurden hier die Programmierung, Algorithmen, die Effizienz des Codes und das Vermeiden von Overhead genannt (70 %/34 %). Weitere Punkte waren die Verlängerung der Hardwarenutzung (0 %/4 %), der Fokus auf effiziente Hardware (15 %/2 %), das Messen der Software in Bezug auf den Stromverbrauch der Hardware (0 %/2 %) sowie die Einsparung von Strom (30 %/22 %) und CO₂ (19 %/5 %). Ein Teil der Befragten gab ebenfalls die Auswahl einer geeigneten Architektur und Programmiersprache (19 %/5 %) und Carbon Aware Computing (7 %/0 %) an.

Ökologisch/ Ressourcenschonend/Nachhaltige Lösungen (67 % Expertenbefragung/46 % offene Umfrage):

67 % der Expertinnen und Experten und 46 % der Befragten der offenen Umfrage gaben Schlagwörter wie „ökologisch“, „ressourcenschonend“ und „nachhaltig“ als Verständnis von Green Coding an.

Organisatorische/Management-Lösungen (11 % Expertenbefragung/18 % offene Umfrage):

11 % der Expertinnen und Experten und 18 % der Befragten der offenen Umfrage verstehen unter Green Coding organisatorische und Management-Lösungen. Dazu zählt im Spezifischen das Vermeiden von sogenannten Rebound-Effekten (0 %/1 %).

Ein Rebound-Effekt tritt auf, wenn eine Steigerung der Effizienz dazu führt, dass die eingesparten Ressourcen dazu genutzt werden, die Nachfrage nach einer Dienstleistung oder einem Produkt zu steigern, was den Gesamtnutzen der Effizienzsteigerung verringert oder sogar ins Gegenteil verkehrt. Im Bereich Green Coding bedeutet dies, dass der aufgrund des optimierten Codes erzielte Gewinn durch reduzierte Strom- und Hardwarenutzung infolge einer erhöhten Nutzung oder zusätzlicher Features kompensiert werden könnte, was den positiven Effekt mindert oder aufhebt. Des Weiteren wurden der sparsame Umgang mit Daten und die Reduzierung der Nutzung von Ressourcen genannt, zum Beispiel durch das Abschalten ungenutzter Systeme (4 %/7 %).

Alternative Lösungen (4 % Expertenbefragung/5 % offene Umfrage):

4 % der Expertinnen und Experten und 5 % der Befragten der offenen Umfrage verstehen unter Green Coding alternative Lösungen, zu denen freie Open Source Software (FOSS) (0 %/1 %), Beschreibungen wie sozial (4 %/1 %) und transparent (0 %/1 %) sowie Analogien zum Aspekt Sicherheit (0 %/1 %) in der Softwareentwicklung zählen.

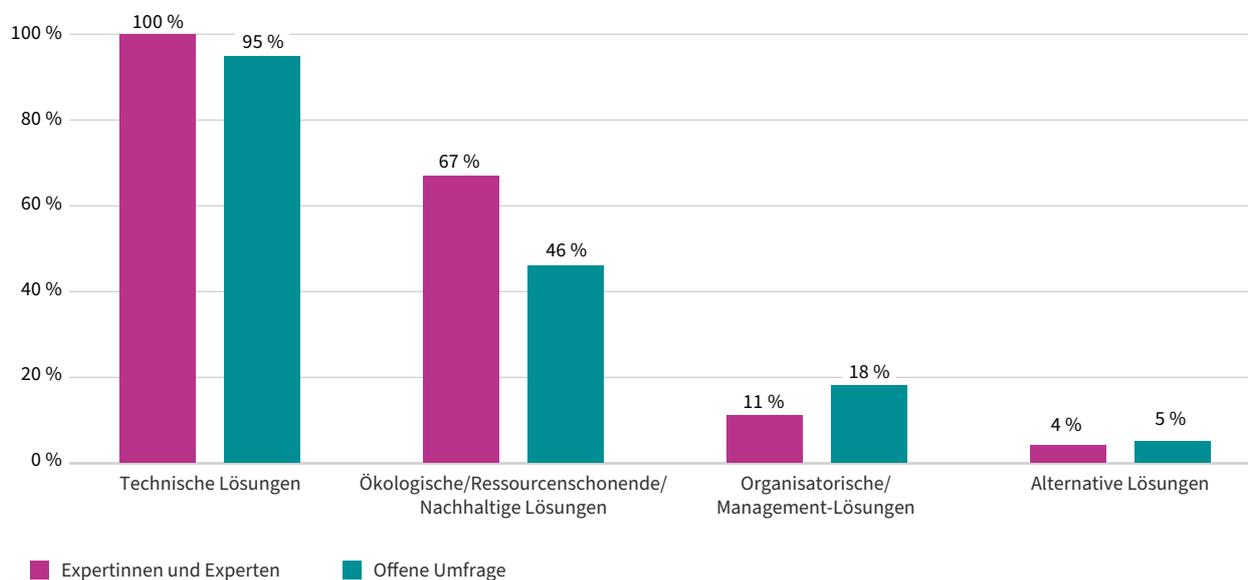


Abbildung 2: Verständnis von Green Coding. Quelle: Eigene Darstellung

Expertenbefragung

In der Expertenbefragung zeigt sich das Bild, dass Green Coding von allen Expertinnen und Experten zum Teil als technische Lösung wahrgenommen wird (100 % Zustimmung). Hinsichtlich der spezifischen Techniken und Maßnahmen besteht jedoch Uneinigkeit. Zwar dominiert die Auffassung, dass Ressourcen geschont werden müssen und Strom eingespart wird, doch wird der Begriff der Ressource nicht klar definiert. Darüber hinaus werden nur sehr selten konkrete Methodiken und Hebelpunkte angesprochen, was auf das Potenzial einer Einführung von klaren Definitionen, Methodiken und Begriffen in der Diskussion hinweist.

Offene Umfrage

Die Analyse der Antworten der offenen Umfrage führte ebenfalls zu wesentlichen Erkenntnissen. Der überwiegende Teil der Befragten assoziiert Green Coding mit technischen Lösungen und Programmierung (95 %). Organisatorische Maßnahmen werden im Vergleich dazu seltener mit Green Coding in Verbindung gebracht und wurden nur von 18 % der Befragten

erwähnt. Die verschiedenen Techniken, die genannt wurden, entsprechen nicht den in der Literatur als wirksam identifizierten Hebeln wie Carbon Aware Computing oder dem dynamischen Skalieren und Abschalten von Systemen.

22 % der Befragten nannten Stromeinsparungen als Ziel. Die CO₂-Reduktion wurde hingegen nur in 5 % der Antworten erwähnt. Zudem wurden oft vage Begriffe wie „nachhaltig“ oder „ressourcenschonend“ verwendet, ohne dass klargestellt wurde, welche spezifischen Ressourcen genau gemeint sind. Konkrete Techniken mit nachweislich hohen Einsparpotenzialen in Bezug auf Stromverbrauch und CO₂-Emissionen, wie beispielsweise dynamisches Skalieren und Abschalten von Hardware oder die längere Nutzung von Geräten, wurden nur sporadisch genannt.

Auf das präzise Messen des Stromverbrauchs und der CO₂-Emissionen wurde ebenfalls kaum eingegangen, obwohl diese Messungen eine grundlegende Voraussetzung für viele der vorgeschlagenen Maßnahmen darstellen. Nur 2 % der befragten

Personen gaben „Messen“ als Antwort. Auch hier scheint der Begriff der Ressource in vielen Kontexten unklar definiert zu sein, was zu den gleichen Schlussfolgerungen wie bei der Expertenbefragung führt, dass der Begriff definiert werden muss, zudem aber auch Potenziale und ihre Auswirkungen auf die Ressourceneinsparung dargestellt werden.

Gemeinsames Verständnis des Begriffs Green Coding im Kontext der Umfrage

Für die weiteren Fragen der Umfrage war es von Bedeutung, dass die Teilnehmerinnen und Teilnehmer ein gemeinsames Verständnis des Begriffs „Green Coding“ haben, um die Fragen im Kontext dieses gemeinsamen Verständnisses beantworten zu können. Die Beschreibung basiert auf den Erkenntnissen aus der Literaturrecherche und umfasst zudem die Darstellung von anschaulichen Beispielen für die Teilnehmerinnen und Teilnehmer. Ziel war es, unterschiedliche Interpretationen zu normieren und insbesondere das Verständnis der Beteiligten der Umfragen in Bezug auf Green Coding dahingehend zu erweitern, dass Green Coding maßgeblich im Kontext von Algorithmus-Optimierungen zu verstehen ist. Darüber hinaus wurde darauf hingewiesen, dass der Begriff in der Literatur durchaus weiter gefasst sein kann und weitere Aspekte einbezieht, wie beispielsweise die Vermeidung von Hardware-Obsoleszenz durch Softwareoptimierung.

Der folgende Text wurde den Teilnehmerinnen und Teilnehmern eingeblendet:

„Für den Kontext des Reports würden wir die Bedeutung des Begriffs ‚Green Coding‘ gerne normieren und legen daher eine Definition fest, die gegenüber Ihrem Verständnis möglicherweise eingeschränkt ist. Unter Green Coding verstehen wir im Folgenden Softwareentwicklung mit Fokus auf die Reduktion des Stromverbrauchs und der CO₂-Emissionen. Beispiele hierfür sind die Nutzung von Algorithmen, die besonders wenig Strom verbrauchen, ebenso KI-Modelle, die speziell auf den Stromverbrauch optimiert wurden, oder die Anpassung der Software, sodass diese auch auf alter Hardware lauffähig ist (um Hardware-Obsoleszenz zu vermeiden), sowie das Prinzip, Rechenoperationen bevorzugt zu Zeiten auszuführen, zu denen ein Server mit grünem Strom betrieben werden kann. Typischerweise nicht zu Green Coding gehören Praktiken, die durch Software CO₂-Einsparungen außerhalb der digitalen Welt erzeugen. Zum Beispiel würde eine Software, die Staus auf Autobahnen reduziert, nicht aufgrund dieser Wirkung unter Green Coding fallen. Bei Green Coding geht es um die Reduktion von CO₂-Emissionen durch effiziente Software, das heißt Software, die Hardware und Strom effizient so nutzt, dass möglichst wenig Emissionen erzeugt werden.“

Beschäftigung mit dem Thema Green Coding

Frage: Wie haben Sie sich schon mit dem Thema Green Coding beschäftigt? (EU5, OU5)

Typ: Freitext-Frage

Zielgruppe: Expertinnen und Experten sowie offene Umfrage

Ziel der Freitext-Frage „Wie haben Sie sich schon mit dem Thema *Green Coding* beschäftigt?“ war es, herauszufinden, in welchem Kontext sich die Befragten mit Green Coding beschäftigt haben. Die Ergebnisse zeichnen ein vielfältiges Bild der Auseinandersetzung mit dem Thema Green Coding.

Expertenbefragung

38 % der Expertinnen und Experten haben sich im Zusammenhang mit ihrer Erwerbsarbeit mit Green Coding auseinandergesetzt und 29 % haben sich im Kontext von Wissenschaft und Bildung mit Green Coding beschäftigt. Dazu zählen beispielsweise das Besuchen von Vorlesungen, das Verfassen von Haus- und Abschlussarbeiten sowie wissenschaftliche Publikationen. Workshops und Konferenzen waren für 17 % der Expertinnen und Experten eine wichtige Informationsquelle, während 8 % durch eigenständige Weiterbildung und weitere 8 % durch andere, nicht weiter spezifizierte Quellen an Informationen zum Thema Green Coding gelangt sind.

Insgesamt sticht hervor, dass ein großer Teil der Befragten sich eigenständig mit dem Thema Green Coding auseinandergesetzt hat. Dies könnte auf ein großes individuelles Interesse und eine proaktive Auseinandersetzung hindeuten, während die geringe Beteiligung durch firmeninterne Veranstaltungen und die Presse darauf hinweist, dass es hier Potenzial für eine verstärkte institutionelle und mediale Aufklärung geben könnte.

Offene Umfrage

Bei der offenen Umfrage zeigt sich ein anderes Bild. 26 % der Befragten gaben an, sich durch eigenständige Erarbeitung des Themas mit Green Coding beschäftigt zu haben, was auf ein starkes Eigeninteresse an Green Coding hinweist. Weitere 21 % der Befragten antworteten, sich im wissenschaftlichen Kontext mit dem Thema auseinandergesetzt zu haben, was auf eine gewisse akademische Relevanz und Einbindung in wissenschaftliche Diskurse hinweist.

15 % haben das Thema im Kontext der Erwerbsarbeit kennengelernt. Workshops und Konferenzen wurden von 14 % der Befragten als Quelle ihrer Auseinandersetzung damit genannt, was ein Zeichen für die Bedeutung von Fachveranstaltungen und Netzwerken sein kann. 15 % der Teilnehmerinnen und Teilnehmer gaben nicht näher spezifizierte Informationsmaterialien als Quelle an.

Jedoch ist zu beachten, dass nur 7 % der Befragten sich über die Presse informiert haben, was im Vergleich relativ gering erscheint und auf eine möglicherweise unzureichende mediale Präsenz des Themas schließen lässt. Am unteren Ende des Spektrums stehen firmeninterne Bildungs- und Weiterbildungsveranstaltungen, die nur 1 % der Befragten als Quelle angegeben hat. Dies könnte auf eine ausbaufähige interne Kommunikation und Weiterbildung in Unternehmen hinweisen, die Green Coding bisher nicht ausreichend in ihre Bildungsprogramme integriert haben.

Es ist wichtig, zu beachten, dass die Teilnehmerinnen und Teilnehmer der Umfrage wahrscheinlich ein grundsätzliches Interesse am Thema Green Coding haben, was zu einem gewissen Bias in den Ergebnissen führt. Die Umfrage wurde über einschlägige Kanäle (siehe Kapitel 3.2) zum Thema Nachhaltigkeit und Software geteilt, die Personen erreichen, die sich für das spezifische Thema interessieren. Dennoch sind die Antworten auf die gestellte Frage interessant, da sie wertvolle Einblicke bieten, wie sich diese Personen informieren und bilden, um Wissen in ihre beruflichen, wissenschaftlichen oder persönlichen Kontexte zu tragen. Die Umfrage zeigt, welche Informationskanäle effektiv genutzt werden und wo Lücken in der Verbreitung bestehen. Dies kann wichtige Hinweise für die zukünftige Verankerung und Weiterverbreitung des Themas Green Coding liefern.

Relevanz

Frage: Welche Relevanz hat Green Coding Ihrer Meinung nach in Bezug auf die Reduktion des Stromverbrauchs digitaler Systeme? (EU6, OU6)

Typ: Multiple-Choice-Frage

Zielgruppe: Expertinnen und Experten sowie offene Umfrage

Wie die Literaturrecherche gezeigt hat, weist moderne IT-Infrastruktur einen großen Strom- und Hardwareressourcenbedarf auf. Vor diesem Hintergrund sollte die Multiple-Choice-Frage „Welche Relevanz hat Green Coding Ihrer Meinung nach in Bezug auf die Reduktion des Stromverbrauchs digitaler Systeme?“ ermitteln, wie das Potenzial von Green Coding eingeschätzt wird, den Stromverbrauch digitaler Systeme zu senken.

Expertenbefragung

Bei der Expertenbefragung zeigt sich ein klareres Bild: 63 % der Expertinnen und Experten stufen Green Coding als „sehr relevant“ für die Reduktion des Stromverbrauchs digitaler Systeme ein, während 37 % es als „eher relevant“ betrachten. Keine der befragten Personen aus dieser Gruppe sieht Green Coding als „eher nicht relevant“ oder „nicht relevant“ an und ebenfalls keine gab an, keine Meinung dazu zu haben. Dies unterstreicht die hohe Relevanz, die Expertinnen und Experten dem Green Coding beimessen.

Offene Umfrage

Die offene Umfrage ergab, dass 48 % der Befragten Green Coding als „eher relevant“ in Bezug auf die Reduktion des Stromverbrauchs digitaler Systeme einschätzen. Ein Anteil von 39 % stuft Green Coding als „sehr relevant“ ein. Dies zeigt, dass ein großer Teil der Befragten die Bedeutung von Green Coding anerkennt. Allerdings sehen 7 % der Befragten Green Coding als „eher nicht relevant“ an und 3 % glauben, dass es „keine Relevanz“ hat. Weitere 3 % der Befragten haben „keine Meinung“ oder haben keine Angabe zur Relevanz von Green Coding gemacht. Diese Ergebnisse deuten darauf hin, dass, obwohl eine Mehrheit der Befragten die Bedeutung von Green Coding erkennt, die Relevanz von einigen wenigen in Frage gestellt wird.

Wie im nachfolgenden Diagramm graphisch dargestellt, zeigen die Umfrageergebnisse, dass sowohl innerhalb der offenen Umfrage als auch unter den Expertinnen und Experten eine Zustimmung zu der Aussage besteht, dass Green Coding zur Reduktion des Stromverbrauchs digitaler Systeme beitragen kann. Während die Beteiligten der offenen Umfrage überwiegend die Relevanz von Green Coding anerkennen, aber auch einige skeptische oder unentschlossene Stimmen bestehen, ist die Expertenmeinung deutlich einheitlicher und positiver gegenüber Green Coding. Dies deutet darauf hin, dass Expertinnen und Experten von der Notwendigkeit und dem Nutzen von Green Coding eher überzeugt und möglicherweise besser über die Vorteile des Einsatzes von Green Coding informiert sind. Gleichzeitig muss aber auch wiederholt klargestellt werden, dass hier ein gewisses Bias besteht, da die Expertenbefragung innerhalb des eigenen Netzwerks verbreitet wurde.

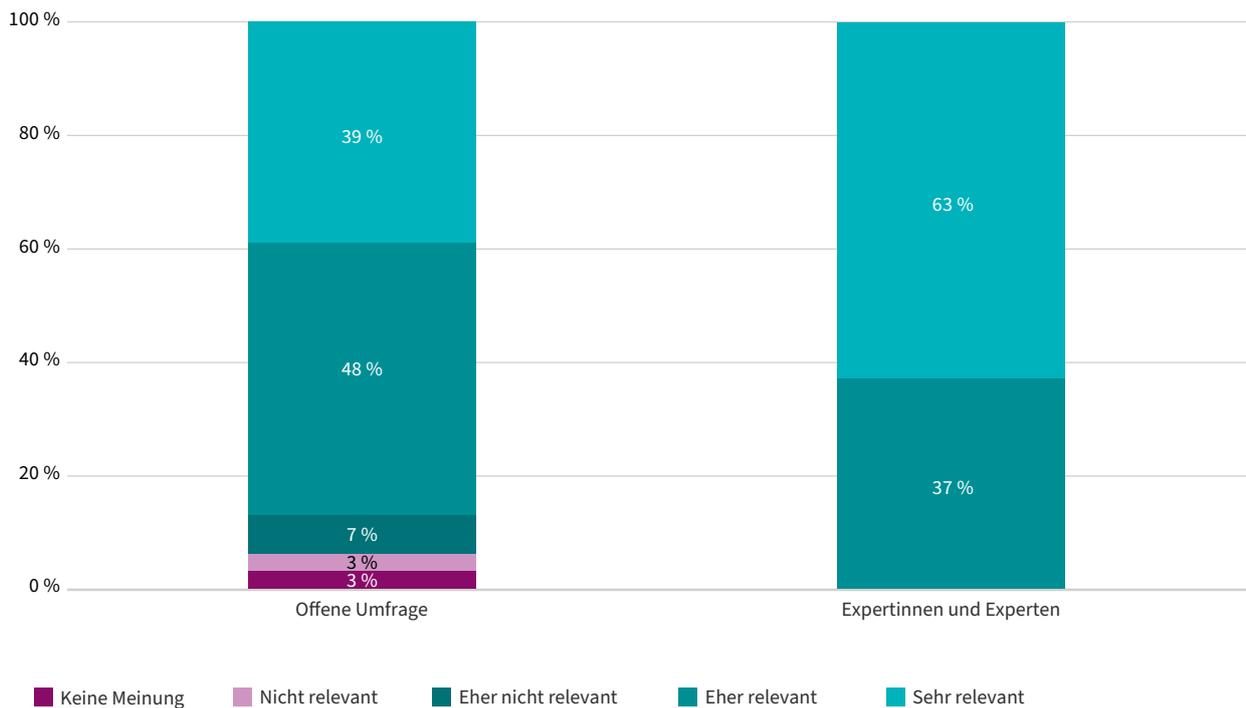


Abbildung 3: Relevanz von Green Coding in Bezug auf die Reduktion des Stromverbrauchs digitaler Systeme. Quelle: Eigene Darstellung

Hindernisse bei der Nutzung oder Implementierung von Green Coding

Frage: Welche Hindernisse bei der Nutzung oder Implementierung von Green Coding sind Ihnen bekannt? (EU7)

Typ: Freitext-Frage

Zielgruppe: Expertinnen und Experten

Durch die Untersuchung der Frage „Welche Hindernisse bei der Nutzung oder Implementierung von Green Coding sind Ihnen bekannt?“ sollte ein tieferes Verständnis der praktischen Herausforderungen bei der Einführung von Green Coding gewonnen werden. Die Freitext-Antworten der Expertinnen und Experten wurden in verschiedene Kategorien unterteilt, die jeweils spezifische Aspekte beleuchten. Da es hier sowohl eine sehr breite Streuung als auch eine Überschneidung der verschiedenen Antworten gab, wurden manche Antworten mehreren Kategorien zugeordnet (siehe Kapitel 3.2, Kategorisierung der Freitext-Antworten).

Wissen und Schulung (70 %): 70 % der Befragten identifizierten mangelndes Wissen und unzureichende Schulungen als Herausforderung für die Anwendung von Green Coding. Dazu zählen beispielsweise das geringe Angebot an Lehrbüchern und Tutorials. Auch das fehlende Wissen zu Messmethodik wurde genannt. Dies deutet darauf hin, dass es einen erheblichen Bedarf an Bildung und Weiterbildung gibt, um das Bewusstsein und die Fähigkeiten im Bereich Green Coding zu verbessern.

Fehlende Standards und Messmethoden (56 %): 56 % der Befragten nannten fehlende Standards und Messmethoden als Hindernis. Ohne klare Richtlinien und Messinstrumente sei es schwierig, die Wirksamkeit von Green-Coding-Maßnahmen zu bewerten und zu vergleichen, was die Implementierung und Akzeptanz erschwert.

Technische und organisatorische Herausforderungen (56 %): Technische und organisatorische Herausforderungen wurden ebenfalls von 56 % der Befragten als Hindernis genannt. Hier wurde unter anderem angeführt, dass Expertise sowie einfache technische Lösungen für die Umsetzung fehlten und dass bestehende organisatorische Strukturen einen Eingriff in Systeme verhinderten, die im Unternehmen genutzt werden.

Ökonomische Prioritäten und mangelnde Anreize (52 %): Ökonomische Prioritäten und fehlende Anreize stellen für 52 % der Befragten ein wesentliches Hindernis dar. Häufig stünden kurzfristige wirtschaftliche Ziele im Vordergrund, wodurch nachhaltige Praktiken wie Green Coding vernachlässigt werden. Zudem fehlten finanzielle oder andere Anreize, um Unternehmen zur Umsetzung von Green Coding zu motivieren.

Regulatorische und gesellschaftliche Aspekte (26 %): Regulatorische und gesellschaftliche Aspekte werden von 26 % der Befragten als Hindernisse betrachtet. Dies schließt das Fehlen von gesetzlichen Vorgaben und gesellschaftlicher Unterstützung ein, die notwendig wären, um Green Coding breiter zu etablieren.

Keine einheitliche Definition (4 %): Ein kleinerer Anteil der Befragten (4 %) sieht das Fehlen einer einheitlichen Definition von Green Coding als Problem an. Ohne eine klare und allgemein akzeptierte Definition sei es schwierig, ein gemeinsames Verständnis und eine gemeinsame Basis für Maßnahmen zu schaffen.

Fehlende Cases (4 %): Ebenfalls 4 % der Befragten nannten fehlende Anwendungsfälle als Hindernis. Ohne praktische Beispiele und Erfolgsgeschichten sei es schwierig, die Vorteile von Green Coding zu demonstrieren und andere zur Nachahmung zu motivieren.

Label/Siegel (4 %): Schließlich sehen 4 % der Befragten das Fehlen von Labels oder Siegeln als Herausforderung an. Solche Zertifikate könnten helfen, Green-Coding-Initiativen sichtbar zu machen und ihre Akzeptanz und Glaubwürdigkeit zu erhöhen.

Zusammenfassend lässt sich feststellen, dass eine Vielzahl von Hindernissen die Implementierung von Green Coding erschweren. Hauptbarrieren sind ein Mangel an Wissen und Schulung, fehlende Standards und Messmethoden sowie technische und organisatorische Herausforderungen. Ökonomische Prioritäten und fehlende Anreize spielen ebenfalls eine bedeutende Rolle sowie regulatorische und gesellschaftliche Aspekte. Weniger häufig, aber dennoch relevant sind das Fehlen einer einheitlichen Definition, fehlende Anwendungsfälle und das Fehlen von Labels oder Siegeln. Gerade der letzte Punkt ist interessant, da es bestehende Labels wie zum Beispiel den Blauen Engel für Software, Green Software Design – Certificate und die Green Certified Site gibt, die bei den Befragten entweder nicht ausreichend oder gar nicht bekannt sind. Das nachfolgende Diagramm stellt die einzelnen Hindernisse noch einmal graphisch dar.

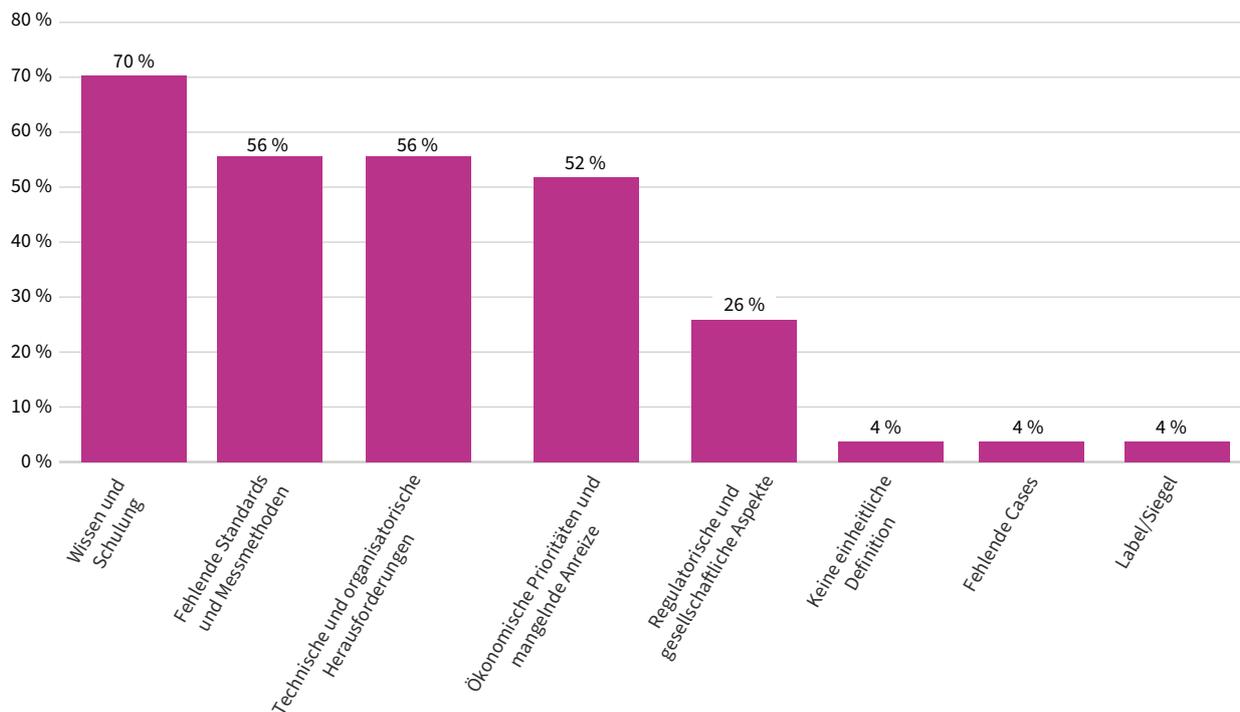


Abbildung 4: Hindernisse bei der Nutzung oder Implementierung von Green Coding. Quelle: Eigene Darstellung

Green Coding als Standard in der Softwareentwicklung

Frage: Welche Gründe sehen Sie dafür, dass Green Coding aktuell nicht der Standard in der Softwareentwicklung ist? (EU8)

Typ: Freitext-Frage

Zielgruppe: Expertinnen und Experten

Um zu verstehen, warum Green Coding derzeit nicht als Standard in der Softwareentwicklung etabliert ist, wurden die Expertinnen und Experten gefragt: „Welche Gründe sehen

Sie dafür, dass Green Coding aktuell nicht der Standard in der Softwareentwicklung ist?“ Die folgenden Gründe wurden genannt:

Mangel an Bewusstsein und Wissen (41 %): 41 % der Befragten sehen den Grund in einem Mangel an Bewusstsein und Wissen über Green Coding. Entwicklerinnen und Entwickler sowie Entscheidungsträgerinnen und -träger seien sich möglicherweise der Vorteile und Praktiken des Green Coding nicht ausreichend bewusst. Dieser Mangel an Information und Verständnis verhindere, dass Green Coding in der Breite implementiert wird.

Wirtschaftliche Prioritäten und fehlender Druck (33 %):

Weitere große Hindernisse stellen die Priorisierung wirtschaftlicher Ziele und der fehlende Druck zur Implementierung nachhaltiger Praktiken dar. 33 % der Befragten gaben an, dass ökonomische Prioritäten wie Kosteneffizienz und schnelle Markteinführung oft über ökologische Überlegungen gestellt werden. Zudem fehle häufig der externe Druck durch Regulierungen oder Marktanforderungen, der Unternehmen zur Umsetzung von Green Coding bewegen würde.

Fehlende Standards und Best Practices (15 %): 15 % der Befragten haben das Fehlen von Standards und Best Practices als Hindernis identifiziert. Ohne etablierte Richtlinien und bewährte Verfahren sei es schwierig, Green Coding systematisch und konsistent zu implementieren. Standards und Best Practices könnten Klarheit und Orientierung bieten und die Verbreitung von Green Coding fördern.

Kulturelle und soziale Faktoren (7 %): Kulturelle und soziale Faktoren sowie übergreifende gesellschaftliche Aspekte spielen für 7 % der Befragten eine Rolle. So gab eine befragte Person beispielsweise an, dass ein Grund darin bestehe, dass Ökologie im Allgemeinen nicht handlungstreibend sei. Eine weitere Person antwortete, dass die Gründe die gleichen seien, die die Menschheit davon abhalten würden, der Klimakrise zu begegnen. Diese kulturellen Barrieren können die Akzeptanz und Umsetzung von Green Coding erheblich erschweren.

Technologische und organisatorische Hürden (4 %): Schließlich sehen 4 % der Befragten technologische und organisatorische Hürden als Grund dafür, dass Green Coding nicht der Standard ist. Dazu zählen etwa das Fehlen von einfach einsetzbaren Messwerkzeugen sowie organisatorische Strukturen und Prozesse, die beispielsweise die Anpassung von Entwicklungsprozessen verhindern.

Die Umfrageergebnisse zeigen, dass eine Vielzahl von Gründen darauf einzahlen, dass Green Coding derzeit nicht als Standard in der Softwareentwicklung etabliert ist. Dabei wird der Mangel an Bewusstsein und Wissen am häufigsten als Grund genannt, gefolgt von wirtschaftlichen Prioritäten und fehlendem Druck zur Umsetzung nachhaltiger Praktiken. Das Fehlen von Standards und Best Practices, kulturelle und soziale Faktoren sowie technologische und organisatorische Hürden würden ebenfalls zur aktuellen Situation beitragen. Das nachstehende Diagramm visualisiert diese Gründe.

Die Erkenntnisse verdeutlichen die Notwendigkeit eines umfassenden Ansatzes, um Green Coding zu fördern. Aufklärung und Bildung sollten dazu beitragen, das Bewusstsein und Wissen über Green Coding zu erweitern. Gleichzeitig sollten wirtschaftliche Anreize und Regulierungen geschaffen werden, um den Druck zur Anwendung zu verstärken. Die Entwicklung und Verbreitung von Standards und Best Practices können zudem Klarheit und Orientierung bieten, während kulturelle Veränderungen und die Überwindung technologischer und organisatorischer Hürden die Akzeptanz erhöhen und die Umsetzung von Green Coding erleichtern können. Nur durch die Adressierung der diversen Herausforderungen kann Green Coding als Standard in der Softwareentwicklung etabliert und das volle Potenzial zur Reduktion des Stromverbrauchs und zur Förderung der Nachhaltigkeit ausgeschöpft werden.

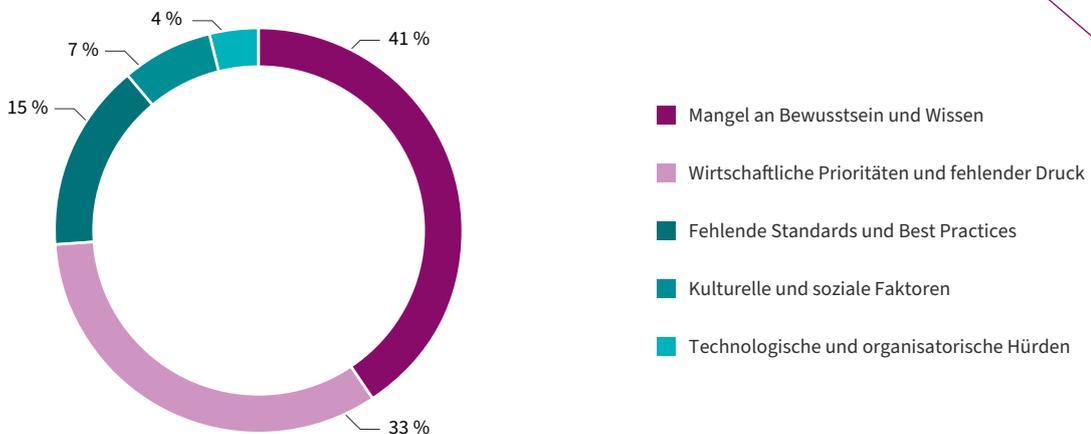


Abbildung 5: Hemmnisse für Green Coding als Standard in der Softwareentwicklung. Quelle: Eigene Darstellung

Potenzielle Nachteile von Green Coding

Frage: Denken Sie, dass Green Coding auch Nachteile haben könnte? (EU9)

Typ: Multiple-Choice-Frage mit Freitext-Option

Zielgruppe: Expertinnen und Experten

Um zu untersuchen, ob Green Coding auch Nachteile mit sich bringen kann, stellten wir den Expertinnen und Experten die Frage: „Denken Sie, dass Green Coding auch Nachteile haben könnte?“ Die Befragten hatten die Möglichkeit, mit „Nein“ zu antworten oder bei Auswahl von „Ja“ ein Freitext-Feld für zusätzliche Erläuterungen auszufüllen. Die Antworten der Befragten wurden in verschiedene Kategorien eingeteilt, um Bedenken zu und potenzielle Nachteile von Green Coding näher zu beleuchten.

Technische Herausforderungen und Kompromisse (32 %):

Der am häufigsten genannte Nachteil sind technische Herausforderungen und Kompromisse, die von 32 % der Befragten erwähnt wurden. Dies umfasst Schwierigkeiten bei der Implementierung von Green-Coding-Praktiken in bestehende Systeme sowie die Notwendigkeit, technische Lösungen zu finden, die sowohl nachhaltig als auch leistungsfähig sind. Es könnten auch Kompromisse bei der Systemleistung oder -stabilität erforderlich sein, um Stromeinsparungen zu erzielen.

Zeit- und Kostenaufwand (24 %): Ein weiterer relevanter Nachteil wird im erhöhten Zeit- und Kostenaufwand gesehen, der von 24 % der Befragten genannt wurde. Die Implementierung von Green Coding erfordert Schulungen und Investitionen in neue Technologien oder Tools. Diese zusätzlichen Aufwände könnten insbesondere für kleinere Unternehmen oder Projekte eine erhebliche Hürde darstellen.

Markt- und wirtschaftliche Auswirkungen (11 %): 11 % der Befragten befürchten markt- und wirtschaftliche Auswirkungen. Wenn nachhaltige Praktiken die Entwicklungszeit verlängern oder die Kosten erhöhen, könnte sich dies negativ auf die Wettbewerbsfähigkeit auswirken. Zudem wäre der Verlust von Marktanteilen möglich, wenn sich die Produktentwicklung verlangsamt.

Einschränkungen bei der Funktionalität und Usability (7 %):

Einschränkungen bei der Funktionalität und Benutzerfreundlichkeit wurden von 7 % der Befragten als potenzieller Nachteil genannt. Die Umsetzung von Green-Coding-Praktiken könnte dazu führen, dass bestimmte Funktionen eingeschränkt werden oder die Benutzererfahrung beeinträchtigt wird, um den Stromverbrauch zu reduzieren.

Komplexität und fehlendes Wissen (7 %):

7 % der Befragten gaben an, dass die Entwicklung durch den Einsatz von Green-Coding-Praktiken an Komplexität gewinnt. Green Coding erfordere spezialisierte Kenntnisse, die möglicherweise zunächst erworben werden müssten und daher mehr Aufwand in der Entwicklung verursachen würden. Darüber hinaus würde das Aufsetzen und Bedienen der erforderlichen Messinfrastruktur ebenfalls die Entwicklungszeit verlängern.

Kein Nachteil (15 %): 15 % der Befragten sehen keine Nachteile von Green Coding.

Die Umfrageergebnisse zeigen, dass verschiedene potenzielle Nachteile von Green Coding wahrgenommen werden, die technischen Herausforderungen und Kompromisse sowie Zeit- und Kostenaufwand mit den einhergehenden wirtschaftlichen Auswirkungen umfassen. Einschränkungen bei der Funktionalität und Usability sowie die Komplexität und das fehlende Wissen sind weitere Bedenken, die von den Befragten geäußert wurden. Gleichzeitig gibt es auch eine Gruppe von Befragten, die aktuell keine Nachteile sieht. Diese Tendenzen werden im nachstehenden Diagramm noch einmal verdeutlicht.

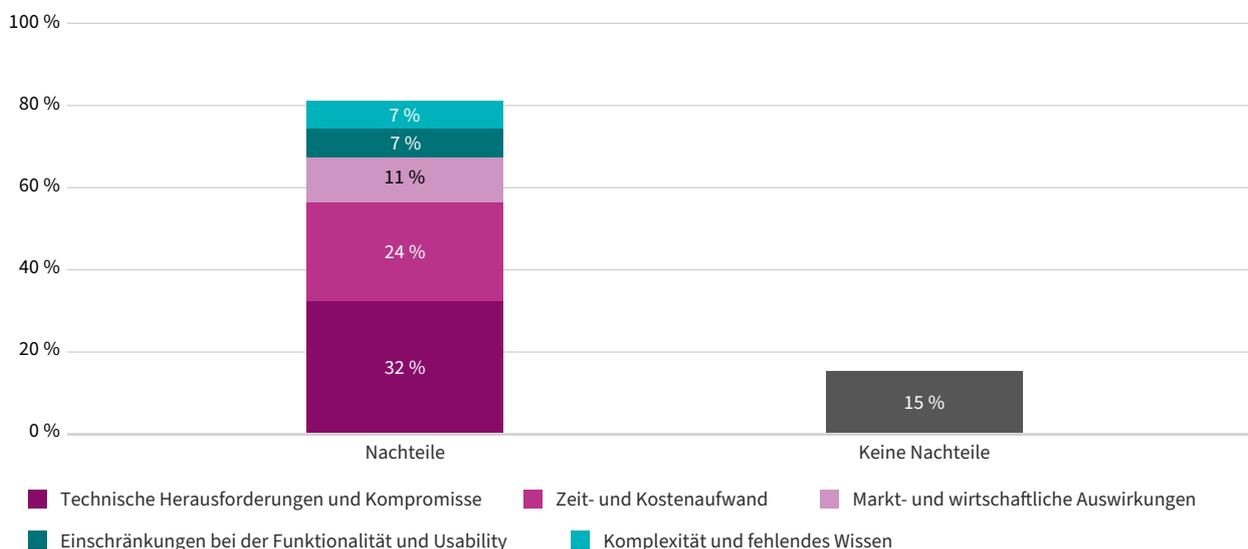


Abbildung 6: Potenzielle Nachteile von Green Coding. Quelle: Eigene Darstellung

Relevante Maßnahmen zur besseren Positionierung von Green Coding im professionellen Umfeld

Frage: Welche relevanten Maßnahmen würden das Thema Green Coding Ihrer Ansicht nach in Ihrem professionellen Umfeld besser positionieren? (EU10)

Typ: Freitext-Frage

Zielgruppe: Expertinnen und Experten

Um zu verstehen, welche Maßnahmen das Thema Green Coding im professionellen Umfeld besser positionieren könnten, stellten wir die Frage: „Welche relevanten Maßnahmen würden das Thema Green Coding Ihrer Ansicht nach in Ihrem professionellen Umfeld besser positionieren?“ Die Antworten der Befragten wurden wiederum in verschiedene Kategorien eingeteilt, die spezifische Maßnahmen und Bedürfnisse hervorheben, um Bedenken zu und potenzielle Nachteile von Green Coding näher zu beleuchten.

Schulungen und Weiterbildung (38 %): Der am häufigsten genannte Bedarf sind Schulungen und Weiterbildung, die von 38 % der Befragten als entscheidend angesehen werden. Durch gezielte Bildungsmaßnahmen und Schulungsprogramme könnten Fachleute die notwendigen Kenntnisse und Fähigkeiten erwerben, um Green Coding effektiv zu implementieren. Dies umfasst sowohl theoretische als auch praktische Schulungen, die ein tieferes Verständnis und die Anwendung nachhaltiger Softwareentwicklungspraktiken fördern.

Transparenz und Bewusstsein (35 %): Ein weiterer wesentlicher Aspekt ist die Förderung von Transparenz und Bewusstsein, die von 35 % der Befragten als wichtig erachtet wird. Dies könnte durch Aufklärungsprogramme, Informationsmaterialien und die regelmäßige Kommunikation über die Vorteile und die Notwendigkeit von Green Coding erreicht werden. Eine stärkere Sensibilisierung könnte dazu beitragen, das Engagement und die Unterstützung im gesamten Unternehmen zu erhöhen.

Standards und Regulierung (15 %): 15 % der Befragten sehen die Etablierung von Standards und Regulierung als notwendig an. Klare Richtlinien und normative Vorgaben könnten einen Rahmen schaffen, innerhalb dessen Green-Coding-Praktiken angewendet werden. Dies würde nicht nur die Implementierung erleichtern, sondern auch eine Vergleichbarkeit und Bewertung der Maßnahmen ermöglichen.

Management und organisatorische Unterstützung (8 %): Management und organisatorische Unterstützung wurden von 8 % der Befragten als relevante Maßnahmen genannt. Führungskräfte und Managementebenen müssten Green Coding aktiv unterstützen und fördern. Dies könnte durch die Bereitstellung von Ressourcen, die Schaffung einer entsprechenden Unternehmenskultur und die Integration von Nachhaltigkeitszielen in die Unternehmensstrategie geschehen.

Tools und Technologien (4 %): Schließlich sehen 4 % der Befragten den Bedarf an spezifischen Tools und Technologien, die Green Coding erleichtern. Dies umfasst Softwarewerkzeuge, die den Stromverbrauch messen und optimieren können, sowie technische Lösungen, die nachhaltige Softwareentwicklungspraktiken unterstützen.

Die Umfrageergebnisse zeigen, dass eine Vielzahl von Maßnahmen getroffen werden könnten, um Green Coding im professionellen Umfeld besser zu positionieren. Schulungen und Weiterbildung sind entscheidend, um das notwendige Wissen und die notwendigen Fähigkeiten zu vermitteln. Transparenz und Bewusstsein sollten gefördert werden, um das Engagement für nachhaltige Praktiken zu erhöhen. Die Etablierung von Standards und Regulierung kann einen klaren Rahmen bieten, während Management und organisatorische Unterstützung die Implementierung erleichtern können. Schließlich sind spezifische Tools und Technologien erforderlich, um Green Coding praktisch umzusetzen.

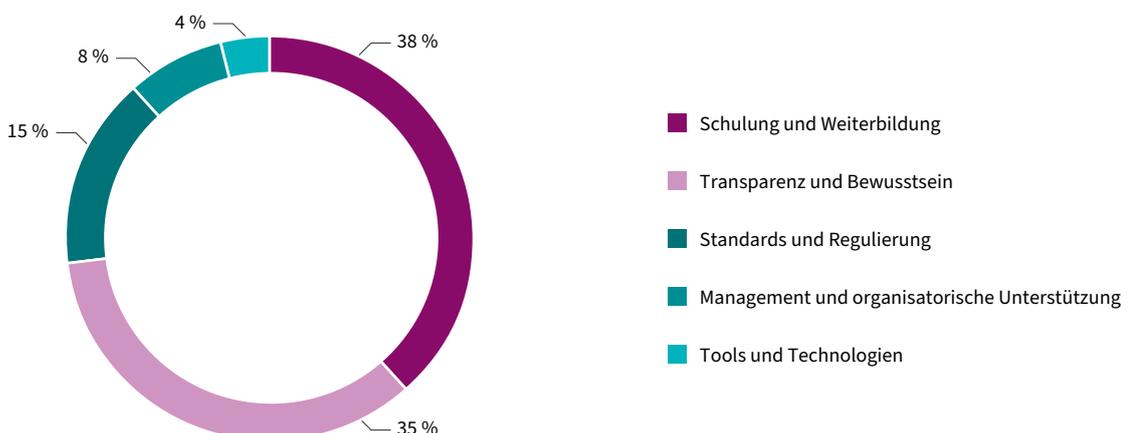


Abbildung 7: Relevante Maßnahmen zur besseren Positionierung von Green Coding. Quelle: Eigene Darstellung

Messen und Kennen des Stromverbrauchs von genutzter Software

Frage: Messen/Kennen Sie bereits den Stromverbrauch der Software, die Sie nutzen (privat/professionell)? (OU7)

Typ: Multiple-Choice-Frage mit Freitext-Option

Zielgruppe: Offene Umfrage

Um zu verstehen, ob und wie der Stromverbrauch von Software im privaten und professionellen Umfeld gemessen wird, stellten wir in der offenen Umfrage die Frage: „Messen/Kennen Sie bereits den Stromverbrauch der Software, die Sie nutzen (privat/professionell)?“ Die Frage wurde so gestellt, dass man „Nein“ auswählen konnte oder bei Auswahl von „Ja“ ein Freitext-Feld angezeigt wurde. Bezüglich dieses Freitext-Feldes wurde nach den Gründen gefragt, warum der Stromverbrauch der Software nicht gemessen bzw. gekannt wird. Die Antworten wurden in verschiedene Kategorien eingeordnet, die die Herausforderungen beim Messen des Stromverbrauchs beleuchten.

Ja (31 %): 31 % der Befragten gaben an, dass sie den Stromverbrauch der von ihnen genutzten Software bereits messen oder kennen. Diese Gruppe ist sich der Methoden zur Überwachung des Stromverbrauchs bewusst und nutzt möglicherweise bereits Tools oder Praktiken, um den Stromverbrauch zu verfolgen und zu optimieren.

Nein (69 %): 69 % beantworteten die Frage mit „Nein“. Die Befragten gaben die folgenden Gründe für die Auswahl dieser Antwort an.

Fehlendes Wissen (61 %): Der größte Teil der Befragten (61 %) gab an, nicht zu wissen, wie sie den Stromverbrauch ihrer Software messen sollen. Dies zeigt ein deutliches Informationsdefizit und verdeutlicht den Bedarf an Aufklärung und Schulung darüber, welche Methoden und Werkzeuge zur Messung des Stromverbrauchs zur Verfügung stehen und wie sie effektiv eingesetzt werden können.

Kein Sinn/Kein Bedarf (19 %): Ein weiterer Anteil der Befragten (19 %) sieht keinen Sinn oder Bedarf darin, den Stromverbrauch ihrer Software zu messen. Diese Haltung kann auf ein mangelndes Bewusstsein für die Bedeutung des Stromverbrauchs im Kontext von Nachhaltigkeit und Kosteneffizienz hinweisen. Hier könnte Aufklärungsarbeit zur Vermittlung der Relevanz dieser Messungen beitragen.

Fehlende Auseinandersetzung mit der Thematik (13 %): 13 % der Befragten haben sich bislang keine Gedanken darüber gemacht, warum oder wie sie den Stromverbrauch ihrer Software messen sollten. Diese Gruppe ist möglicherweise noch nicht mit den potenziellen Vorteilen und Notwendigkeiten des Messens vertraut.

Zu hoher Aufwand (7 %): Schließlich empfinden 7 % der Befragten den Aufwand zur Messung des Stromverbrauchs als zu hoch. Diese Wahrnehmung kann auf fehlende einfache und zugängliche Werkzeuge oder auf eine mangelnde Integration solcher Messungen in bestehende Arbeitsprozesse zurückzuführen sein. Hier könnten technologische Innovationen und benutzerfreundliche Tools Abhilfe schaffen.

Die Umfrageergebnisse verdeutlichen, dass es verschiedene Gründe gibt, warum der Stromverbrauch von Software nicht gemessen wird. Der Grund, der am häufigsten genannt wurde, ist die Unklarheit darüber, wie eine solche Messung durchgeführt werden kann, gefolgt von der Ansicht, dass eine Messung keinen Sinn ergibt oder nicht erforderlich ist. Ein Teil der Befragten hat sich bislang keine Gedanken darüber gemacht oder empfindet den Aufwand als zu hoch.

Diese Erkenntnisse weisen darauf hin, dass ein umfassender Ansatz erforderlich ist, um das Messen des Stromverbrauchs von Software zu fördern. Aufklärung und Schulungen sind entscheidend, um das notwendige Wissen zu vermitteln und das Bewusstsein für die Bedeutung dieser Messungen zu erhöhen. Darüber hinaus könnten die Entwicklung und die Bereitstellung benutzerfreundlicher Werkzeuge und Methoden den Prozess der Messung erleichtern und dazu beitragen, dass mehr Nutzerinnen und Nutzer den Stromverbrauch ihrer Software berücksichtigen. Indem diese Barrieren überwunden werden, können ein breiteres Verständnis und eine bessere Kontrolle des Stromverbrauchs erreicht werden, was letztlich zu einer nachhaltigeren Nutzung digitaler Technologien beiträgt.

Auswahl von Software/KI-Modellen basierend auf dem Stromverbrauch

Frage: Würden Sie eine Auswahl Ihrer genutzten Software/KI-Modelle basierend auf dem Stromverbrauch durchführen statt allein basierend auf der Leistungsfähigkeit? (OU8)

Typ: Multiple-Choice-Frage mit Freitext-Option

Zielgruppe: Offene Umfrage

Diese Frage wurde gestellt, um zu verstehen, ob Nutzerinnen und Nutzer bereit wären, ihre Auswahl an Software oder KI-Modellen basierend auf dem Stromverbrauch statt allein basierend auf der Leistungsfähigkeit zu treffen. Die Freitext-Antworten der Befragten, die „Nein“ gewählt hatten, wurden verschiedenen Kategorien zugeordnet, die ihre Beweggründe und Bedenken beleuchten.

Ja (72 %): Eine Mehrheit der Befragten gab an, dass sie bereit wäre, ihre Auswahl an Software oder KI-Modellen basierend auf dem Stromverbrauch zu treffen. Diese hohe Zustimmung zeigt eine starke Bereitschaft, stromsparende Praktiken in den Auswahlprozess von Software zu integrieren.

Nein (28 %): 28 % beantworteten die Frage mit „Nein“. Die Befragten gaben die folgenden Gründe für die Auswahl dieser Antwort an.

Unzureichende Datenbasis/Fehlende Informationen (6 %): 6 % der Befragten gaben an, dass sie nicht über genügend Daten oder Informationen verfügen, um eine fundierte Entscheidung basierend auf dem Stromverbrauch zu treffen. Dies deutet darauf hin, dass bessere Informationen und mehr Transparenz bezüglich des Stromverbrauchs von Software und KI-Modellen erforderlich sind.

Priorität liegt auf anderen Aspekten/Fehlender Anreiz (6 %): Ebenfalls 6 % der Befragten sehen andere Faktoren als wichtiger an oder haben keine Anreize, den Stromverbrauch in ihre Entscheidung einzubeziehen. Dies könnte auf eine Priorisierung von Leistungsfähigkeit, Kosten oder anderen Kriterien hinweisen, die derzeit als wichtiger erachtet werden.

Fehlende Entscheidungsbefugnis (5 %): Für 5 % der Befragten besteht der Grund darin, nicht über die erforderliche Entscheidungsbefugnis zu verfügen, um die Auswahl von Software oder KI-Modellen zu beeinflussen. Dies betrifft Angestellte, die nicht in relevante Entscheidungsprozesse eingebunden sind, und deutet auf die Notwendigkeit hin, Entscheidungsträgerinnen und -träger sowie relevante Stakeholder zu sensibilisieren.

Eigener Einfluss zu gering (5 %): Genauso gaben 5 % der Befragten an, dass ihre individuellen CO₂-Emissionen zu gering seien, um durch Einsparen einen relevanten Beitrag zur Vermeidung des Klimawandels leisten zu können. Hier zeigt sich erneut der Bedarf an Aufklärung und der Beseitigung von Missverständnissen, um die Wirksamkeit individueller Entscheidungen zu verdeutlichen.

Stromverbrauch als weniger relevant erachtet (4 %): 4 % der Befragten erachten den Stromverbrauch als weniger relevant. Diese Haltung zeigt, dass weitere Sensibilisierungsmaßnahmen notwendig sind, um die Bedeutung des Stromverbrauchs im Kontext von Software hervorzuheben.

Die Umfrageergebnisse zeigen, dass eine große Mehrheit der Befragten bereit wäre, ihre Auswahl an Software oder KI-Modellen basierend auf dem Stromverbrauch zu treffen. Dies kann als ein ausgeprägtes Bewusstsein und eine Bereitschaft interpretiert werden, nachhaltige Entscheidungen zu fördern. Gleichzeitig gibt es jedoch Barrieren wie mangelnde Informationen, fehlende Anreize, eingeschränkte Entscheidungsbefugnis und unzureichender Kenntnisstand, die die Umsetzung solcher Entscheidungen erschweren.

Bereitschaft zur Änderung von Praktiken zur Stromeinsparung

Frage: Würden Sie die Praktiken für die Entwicklung Ihrer Software ändern, wenn verifizierte Praktiken veröffentlicht würden, die nachweislich Strom einsparen? (EU13, OU13)

Typ: Multiple-Choice-Frage mit Freitext-Option

Zielgruppe: Developer

Um zu verstehen, ob die Befragten bereit wären, ihre Entwicklungspraktiken zu ändern, wenn verifizierte und nachweislich stromsparende Praktiken veröffentlicht würden, stellten wir Teilnehmerinnen und Teilnehmern beider Umfragen, die für sich die Angabe „Developer“ gemacht hatten, die Frage: „Würden Sie die Praktiken für die Entwicklung Ihrer Software ändern, wenn verifizierte Praktiken veröffentlicht würden, die nachweislich Strom einsparen?“ Die Antworten der Befragten zeigen eine überwältigende Zustimmung.

Insgesamt gaben 92 % der Befragten an, dass sie dafür offen wären, ihre Entwicklungspraktiken zu ändern, sofern verifizierte und empirisch fundierte stromeinsparende Praktiken bereitgestellt werden. Diese hohe Zustimmungsrates weist auf ein ausgeprägtes Bewusstsein für nachhaltige Entwicklung und die Möglichkeit hin, solche Praktiken in den Softwareentwicklungsprozess zu integrieren, vorausgesetzt ihre Vorteile sind klar nachvollziehbar.

Es sollte jedoch angemerkt werden, dass die Formulierung der Frage möglicherweise einen Einfluss auf die Antworten der Befragten hatte. Der Bezug auf *verifizierte Praktiken* und „nachweisliche Stromeinsparungen“ könnte dazu geführt haben, dass die Teilnehmerinnen und Teilnehmer sich eher positiv äußern, da diese Formulierungen den Eindruck vermitteln könnten, dass die Verantwortung für etwaige Änderungen in ihrem Einflussbereich liegt und dass solche Maßnahmen als vorteilhaft erachtet werden. Lediglich 8 % der Befragten äußerten, dass sie ihre Entwicklungspraktiken nicht anpassen würden, selbst wenn solche verifizierten und effizienzsteigernden Praktiken zur Verfügung stünden.

Dennoch kann die überwiegende Befürwortung der Befragten als Indiz für transformative Veränderungen in der Softwareentwicklung interpretiert werden, sofern praktikable und wissenschaftlich fundierte Ansätze zur Stromeinsparung zur Verfügung stehen.

Bereitschaft zur Änderung von Entwicklungspraktiken zur Stromeinsparung trotz verlängerter Entwicklungszeit

Frage: Würden Sie diese Praktiken auch anwenden, wenn sich dadurch die Entwicklungszeit der Software verlängert? (EU14, OU14)

Typ: Multiple-Choice-Frage mit Freitext-Option

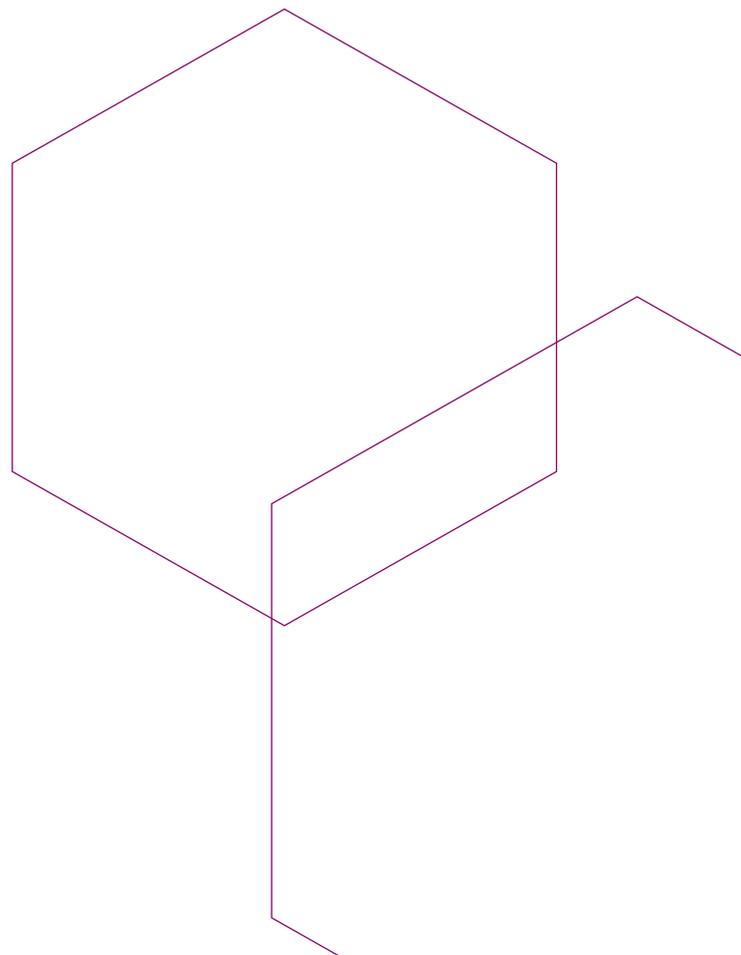
Zielgruppe: Developer

Wie auch bei der vorhergehenden Frage waren die Zielgruppe hier die Developer beider Umfragen. In der vorangegangenen Frage wurden die Teilnehmerinnen und Teilnehmer gefragt, ob sie ihre Praktiken ändern würden, wenn verifizierte und stromsparende Praktiken veröffentlicht würden. 92 % bejahten dies. In dieser Folgefrage wollten wir wissen: „Würden Sie diese Praktiken auch anwenden, wenn sich dadurch die Entwicklungszeit der Software verlängert?“ Die Antworten der Befragten zeigen eine hohe, aber nicht uneingeschränkte Bereitschaft.

66 % der Befragten gaben an, dass sie bereit wären, stromsparende Praktiken anzuwenden, selbst wenn dies die Entwicklungszeit der Software verlängern würde. Dies unterstreicht das starke Engagement für Nachhaltigkeit und die Anerkennung der langfristigen Vorteile von Stromeinsparungen, auch wenn diese Herausforderungen wie längere Entwicklungszyklen mit sich bringen.

34 % der Befragten beantworteten die Frage mit „Nein“ und hoben hervor, dass Zeit bereits jetzt eine knappe Ressource ist. Diese Gruppe könnte unter bestimmten Bedingungen bereit sein, die Praktiken zu nutzen, jedoch nicht uneingeschränkt. Mögliche Einschränkungen sind etwa, dass eine Verlängerung der Entwicklungszeit nur bis zu einem gewissen Punkt akzeptabel ist oder dass andere Faktoren wie Budget, Projektforderungen oder Kundenwünsche eine entscheidende Rolle spielen.

In der vorherigen Frage gab die deutliche Mehrheit der Befragten an, dass sie bereit wären, ihre Entwicklungspraktiken zu ändern, wenn verifizierte stromsparende Praktiken verfügbar wären. Diese hohe Zustimmungsrates zeigt ein starkes Grundbewusstsein und Engagement für Nachhaltigkeit. Die Antworten auf die aktuelle Frage erweitern dieses Bild, indem sie zeigen, dass selbst bei zusätzlichen Herausforderungen wie längeren Entwicklungszeiten eine Mehrheit weiterhin bereit wäre, diese Praktiken anzuwenden.





4.

**Handlungs-
empfehlungen**

Die Entwicklung der Handlungsempfehlungen war ein zentraler Bestandteil dieses Reports. Sie werden zunächst im Unterkapitel Maßnahmenkatalog klar und übersichtlich zusammengefasst. Im darauffolgenden Unterkapitel werden Ansatz und

Methodik des Erstellungsprozesses erläutert, während das Unterkapitel Erarbeitung der Handlungsempfehlungen einen detaillierten Ablauf zur Entwicklung jeder einzelnen Empfehlung beschreibt.

4.1 Maßnahmenkatalog

Die folgenden Tabellen geben einen Überblick über Handlungsempfehlungen und konkrete Maßnahmen für verschiedene Stakeholdergruppen, um die Implementierung von Green Coding effizient voranzutreiben. Ziel ist es, eine klare und strukturierte Übersicht bereitzustellen, die die Umsetzung erleichtert und zur nachhaltigen Gestaltung der Digitalisierung

beiträgt. Die Empfehlungen adressieren Politik, Unternehmen, Bildungseinrichtungen sowie ein Zusammenspiel aller Akteure im Bereich Kommunikation und Zusammenarbeit und zeigen praxisorientierte Ansätze auf, um den Stromverbrauch und die CO₂-Emissionen im IT-Sektor zu reduzieren.

Handlungsempfehlung für die Politik

Einheitliche Definition von Green Coding

Die Bundesregierung sollte einen Stakeholder-Prozess initiieren, um eine einheitliche Definition von Green Coding zu entwickeln, die bundesweit Anwendung finden und weiteren Politiken zugrunde liegen kann.

Maßnahmen:

- Die Bundesregierung sollte relevante Stakeholder wie Vertreterinnen und Vertreter der Wissenschaft und Organisationen wie die Gesellschaft für Informatik, das Umweltbundesamt, den Bundesverband Green Software, das International Requirements Engineering Board (IREB) und die Internationale Organisation für Normung (ISO) sowie potenziell weitere relevante Akteure einbinden, um eine verbindliche Definition des Begriffs „Green Coding“ zu finden. Die Definition sollte spezifische technische und organisatorische Praktiken festlegen und messbare Kriterien für Green Coding definieren.

Transparenzvorschriften

Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von eingebetteten Emissionen bei Hardware entlang ihres gesamten Lebenszyklus einführen.

Maßnahmen:

- Die Bundesregierung sollte eine Expertenkommission einsetzen, die eine Richtlinie und/oder einen Standard für die transparente Berichterstattung von eingebetteten Emissionen erarbeitet.
- Die Bundesregierung sollte gesetzliche Vorgaben entwickeln, die Hardwarelieferanten verpflichten, die eingebetteten Emissionen ihrer Produkte offenzulegen. Diese Angaben sollten den gesamten Lebenszyklus der Hardware umfassen, von der Produktion über den Transport bis zum Recycling bzw. zur Entsorgung.

Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von operativen und eingebetteten Emissionen durch IT-Dienstleistungen und Cloud-Services entlang ihres Lebenszyklus einführen.

Maßnahmen:

- Die Bundesregierung sollte die Entwicklung und Implementierung von standardisierten Berichtsformaten in Auftrag geben, die es der Nutzerschaft ermöglichen, den Stromverbrauch und die Emissionen ihrer genutzten Dienstleistungen zu verstehen und zu vergleichen. Hier könnten schon bestehende Berichte erweitert werden.
- Die Bundesregierung sollte gesetzliche Vorgaben einführen, die IT-Dienstleister verpflichten, ihre operativen und eingebetteten Emissionen je Dienstleistung transparent zu machen.
- Die Europäische Kommission sollte die EU-Taxonomie erweitern, um nicht nur Rechenzentren, sondern auch Softwarelösungen in die Klassifizierung mit einzubeziehen.

Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von operativen und eingebetteten Emissionen bei Software entlang ihres Lebenszyklus einführen.

Maßnahmen:

- Die Bundesregierung sollte die Entwicklung und Implementierung eines standardisierten Berichtsformats in Auftrag geben, um Emissionen von Software entlang des gesamten Lebenszyklus zu erfassen und transparent öffentlich zu machen.
- Die Bundesregierung sollte gesetzliche Vorgaben einführen, die Softwarehersteller verpflichten, die operativen und eingebetteten Emissionen ihrer Software entlang des Lebenszyklus transparent zu machen.

Handlungsempfehlung für die Politik

Schaffung von Anreizen für Unternehmen

Die Bundesregierung sollte eine Förderung für Green-Coding-Pilotprojekte einsetzen.

Maßnahmen:

- Die Bundesregierung sollte finanzielle Mittel zur Förderung von Pilotprojekten bereitstellen, die Green-Coding-Praktiken in der Praxis einsetzen und evaluieren.
- Die Bundesregierung sollte die Geförderten im Rahmen der Förderung verpflichten, Erkenntnisse dieser Pilotprojekte zu veröffentlichen, um Best Practices und erfolgreiche Ansätze öffentlich zu teilen

Die Bundesregierung sollte Maßnahmen zur Optimierung von Software für die Zertifizierung mit dem Blauen Engel finanziell fördern.

Maßnahmen:

- Die Bundesregierung sollte ein spezielles Förderprogramm einführen, das moderate finanzielle Zuschüsse für Unternehmen bereitstellt, um ihre Software an die Anforderungen des Blauen Engels anzupassen.
- Die Bundesregierung sollte moderate finanzielle Unterstützung für Unternehmen bereitstellen, um die Kosten für externe Audits und Zertifizierungsverfahren zu senken, die notwendig sind, um die Konformität ihrer Software mit den Kriterien des Blauen Engels nachzuweisen.

Die Bundesregierung sollte den Erwerb von Software (bzw. Softwarelizenzen), welche mit dem Blauen Engel zertifiziert ist, finanziell fördern, da dies nachhaltige Entwicklungspraktiken stärkt, langfristig Energie- und Ressourceneinsparungen ermöglicht und die Wettbewerbsfähigkeit der deutschen IT-Branche fördert.

Maßnahmen:

- Die Bundesregierung sollte moderate finanzielle Zuschüsse oder Fördermittel für Unternehmen bereitstellen, die Software erwerben möchten, die den Blauen Engel trägt. Diese Programme könnten prozentuale Rabatte oder fixe Beträge anbieten, um die Kosten für die Anschaffung von zertifizierter, umweltfreundlicher Software zu senken.
- Die Bundesregierung sollte steuerliche Vergünstigungen einführen, wie zum Beispiel Abschreibungsmöglichkeiten oder Steuererleichterungen, für Unternehmen, die Software mit dem Blauen Engel erwerben.

Die Bundesregierung sollte gesetzlich vorschreiben, dass in der öffentlichen Beschaffung und im mittelfristigen Einsatz ausschließlich energie- und ressourceneffiziente Software zugelassen wird, um ökologische Nachhaltigkeitsziele wirksam zu unterstützen.

Maßnahmen:

- Die Bundesregierung sollte ein Expertengremium einsetzen, das verbindliche Nachhaltigkeitskriterien für Software definiert. Diese Vorgaben dienen als Grundlage für öffentliche Ausschreibungen, um Transparenz und Vergleichbarkeit zu gewährleisten.
- Die Bundesregierung sollte die öffentliche Hand dazu verpflichten, die identifizierten Nachhaltigkeitskriterien in den Beschaffungsprozess zu integrieren, sodass Softwareanbieter in öffentlichen Ausschreibungen nachweisen müssen, dass ihre Produkte die vorgegebenen Kriterien erfüllen.
- Die Bundesregierung sollte eine verbindliche Frist einführen, bis zu der alle neuen Softwarebeschaffungen im öffentlichen Sektor nur noch Produkte betreffen dürfen, die mit dem Blauen Engel zertifiziert sind.

Handlungsempfehlungen für Unternehmen

Stärkung des Bewusstseins

Unternehmen, die Software entwickeln, sollten Kennzahlen auf Softwareprodukt- und Softwareprojektebene einführen, die den Stromverbrauch und die operativen und eingebetteten Emissionen von Software entlang ihres Lebenszyklus erfassen.

Maßnahmen:

- Unternehmen, die Software entwickeln, sollten Kennzahlen definieren, um den Stromverbrauch und die operativen und eingebetteten Emissionen der Software entlang ihres Lebenszyklus zu messen. Die Kennzahlen sollten vom den Unternehmen in regelmäßigen Abständen bewertet werden.
- Unternehmen, die Software entwickeln, sollten die Kennzahlen pro Service bzw. pro Produkt in ihren Geschäftsberichten ausweisen.

Handlungsempfehlungen für Unternehmen

Unternehmen, die Software einkaufen, sollten Transparenz in Bezug auf den Stromverbrauch und die operativen und eingebetteten Emissionen für Hardware, IT-Dienstleistungen, Cloud-Services und Software bei den Herstellern einfordern.

Maßnahmen:

- Unternehmen, die Software einkaufen, sollten ausdrücklich Transparenz hinsichtlich des Stromverbrauchs und der operativen und eingebetteten Emissionen von erworbener und zu beschaffender Hardware, IT-Dienstleistungen, Cloud-Services und Software von den Herstellern einfordern.

Unternehmensinterne Ausbildungen und Schulungen

Unternehmen sollten interne und zielgruppengerechte Bildungsangebote zum Thema Green Coding bereitstellen, um unternehmensinternes Wissen im Bereich Green Coding aufzubauen.

Maßnahmen:

- Unternehmen, die Software entwickeln, sollten spezifische und auf die Zielgruppe zugeschnittene Schulungen anbieten, die sowohl technische als auch organisatorische Aspekte abdecken. Auf diese Weise kann Green Coding in Entwicklungsprogramme integriert werden, um eine nachhaltige Softwareentwicklung in den Entwicklungsprozessen zu verankern.
- Unternehmen, die Software einkaufen, sollten Schulungsprogramme entwickeln, die es den Beschäftigten ermöglichen, IT-Lieferanten anhand ihrer Green-Coding-Praktiken zu bewerten und so nachhaltigere Entscheidungen in der Beschaffung zu treffen.

Unternehmen sollten Angebote erarbeiten, um ihre Führungskräfte für Green Coding zu sensibilisieren.

Maßnahmen:

- Verbände und Beratungsunternehmen sollten Workshops und Seminare entwickeln, die gezielt Führungskräfte ansprechen und die Bedeutung von Green Coding für den langfristigen Geschäftserfolg sowie für die Erreichung der Umwelt- und Klimaziele verdeutlichen. Die Schulungen sollten auch darauf abzielen, Führungskräfte für die Integration von Green Coding in die unternehmensweite CSR-Strategie (Corporate Social Responsibility) zu gewinnen.

Handlungsempfehlungen für Bildungseinrichtungen

Integration von Green Coding in Curricula von Hochschulen

Hochschulen/Universitäten sollten Green Coding in die akademischen Curricula von informatikorientierten Studiengängen integrieren.

Maßnahmen:

- Hochschulen und Universitäten sollten Green Coding als festen Bestandteil in ihre Curricula von informatikorientierten Studiengängen integrieren, indem sie spezielle Module oder modulübergreifende Konzepte entwickeln.

Hochschulen/Universitäten sollten Kurse und Seminare anbieten, um auch Studentinnen und Studenten in nicht informatikorientierten Studiengängen für nachhaltige IT und Software zu sensibilisieren.

Maßnahmen:

- Hochschulen/Universitäten sollten interdisziplinäre Ansätze fördern, bei denen nachhaltige IT und Software auch in anderen Fachbereichen thematisiert wird.
- Hochschulen/Universitäten sollten Seminare, Workshops und Vorlesungen zu nachhaltiger IT und Software anbieten, um so zur Sensibilisierung eines breiteren Publikums angehender Führungskräfte und Expertinnen und Experten beizutragen.

Mehr Bildungsangebote

Volkshochschulen, Verbände und weitere Bildungseinrichtungen sollten Schulungsangebote zu Green Coding entwickeln und organisieren.

Maßnahmen:

- Fachverbände wie die Gesellschaft für Informatik (GI), Bitkom und der Bundesverband Green Software (BvGS) sollten Programme und Kurse zu Green Coding entwickeln und kostenlos zur Verfügung stellen.
- NGOs und Fachverbände sollten Online-Kurse, Webinare und Workshops zu Green Coding organisieren und sich dabei sowohl an Entwicklerinnen und Entwickler als auch an Führungskräfte und politische Entscheidungsträgerinnen und -träger richten.

Handlungsempfehlungen im Bereich Kommunikation und Zusammenarbeit

Erfolgsberichte

Verbände, NGOs und Behörden sollten konkrete Beispiele für die Anwendung von Green Coding sammeln, aufbereiten und öffentlich verfügbar machen.

Maßnahmen:

- Verbände wie zum Beispiel die Gesellschaft für Informatik oder der Bundesverband Green Software sollten Erfolgsgeschichten sammeln, aufbereiten und öffentlich zugänglich machen.

Unternehmen sollten eigene Erfolgsgeschichten öffentlich teilen und über soziale Medien und Informationskanäle eine breite Öffentlichkeit über die Erfolgsgeschichten informieren.

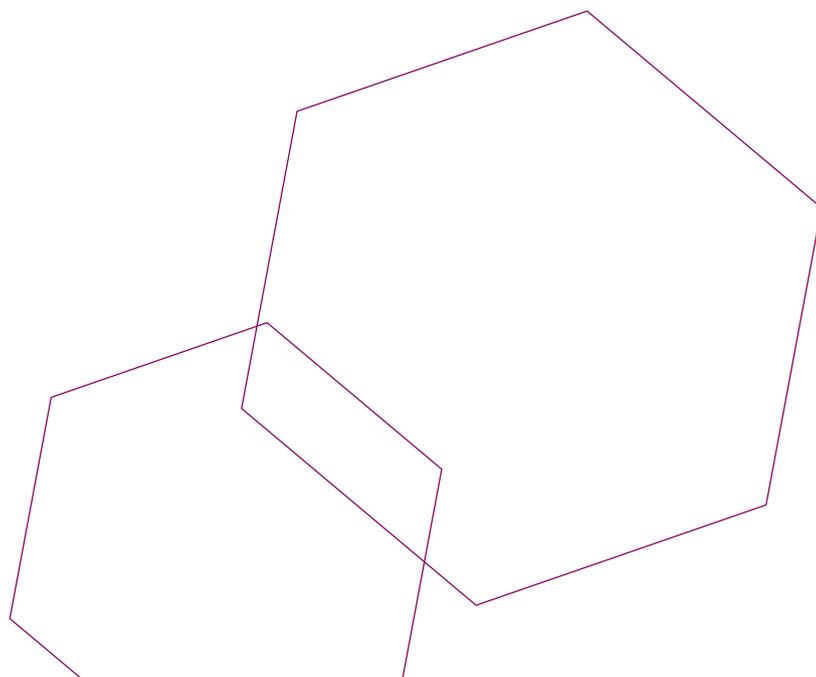
Maßnahmen:

- Unternehmen sollten ihre Erkenntnisse in Bezug auf die Anwendung von Green Coding veröffentlichen, beispielsweise durch eine Präsentation von Erfahrungen und Best Practices.
- Unternehmen sollten Fallstudien und Erfahrungsberichte veröffentlichen, die sowohl technische Details als auch messbare Ergebnisse darstellen.
- Unternehmen sollten soziale Medien, Pressemitteilungen und Branchenkonferenzen nutzen, um Erfolgsgeschichten öffentlichkeitswirksam zu teilen und eine breite Leserschaft zu erreichen.

Ministerien und Verbände sollten Preise für herausragende Green-Coding-Praktiken verleihen und sowohl den Preis als auch die Preisverleihung öffentlichkeitswirksam bewerben.

Maßnahmen:

- Ministerien sollten jährliche Preise für die besten Green-Coding-Initiativen und -Lösungen vergeben, um innovative Ansätze öffentlich zu honorieren.
- Ministerien sollten Informationen über die Preisträger in der Breite kommunizieren, um für Unternehmen mehr Anreize zu schaffen, mit dem Preis ausgezeichnet zu werden.
- Ministerien, die Preise verleihen, sollten die Preisverleihung öffentlichkeitswirksam gestalten, um die Reichweite und den Grad der Wahrnehmung des Themas zu maximieren.



4.2 Ansatz und Methodik

Die Implementierung von Green Coding als Standard in der Softwareentwicklung stellt eine komplexe Herausforderung dar, die auf verschiedenen Ebenen adressiert werden muss. Ein wesentlicher Schritt im Prozess war die systematische Einteilung der Stakeholder in drei zentrale Gruppen, die jeweils spezifische Rollen und Verantwortungen im Bereich Green Coding einnehmen:

Politik: Die Politik wurde als maßgeblicher Akteur identifiziert, um gesetzliche Rahmenbedingungen und Anreize für Green Coding zu schaffen.

Bildungseinrichtungen: Hochschulen, Universitäten sowie Bildungs- und Weiterbildungseinrichtungen inklusive der Volkshochschulen wurden als zentrale Akteure für die Aus- und Weiterbildung der (zukünftigen) Fachkräfte und die Verankerung von Green Coding in der (akademischen) Lehre und Weiterbildung definiert.

Unternehmen: IT-Produzenten und Unternehmen, die IT-Dienstleistungen nutzen, tragen die Hauptverantwortung für die praktische Umsetzung von Green Coding. Diese Gruppe wurde unterteilt in jene, die Software entwickeln, und jene, die sie einkaufen und nutzen. Beide Gruppen können durch nachhaltige Entscheidungen in ihrer IT-Strategie wesentlich zur Reduktion des CO₂-Ausstoßes beitragen.

Nach der Auswertung der Umfrageergebnisse wurden die gewonnenen Erkenntnisse in einem weiteren Workshop mit dem Konsortium und Vertreterinnen und Vertretern aus Politik, Forschung, Wirtschaft und Zivilgesellschaft diskutiert und verfeinert. Aus dieser intensiven Diskussion entstanden realistische und praxisnahe Handlungsempfehlungen, die im

Nachgang in individuellen Interviews mit jeweils einer Vertreterin oder einem Vertreter der Stakeholder-Gruppe hinsichtlich Formulierung und Anwendbarkeit finalisiert wurden.

Die entstandenen 19 Handlungsempfehlungen wurden entwickelt, um die vielfältigen Aspekte von Green Coding abzudecken und eine umfassende Strategie zur Förderung nachhaltiger Softwareentwicklung zu bieten. Die Handlungsempfehlungen wurden in vier Hauptkategorien unterteilt. Drei Kategorien decken dabei die relevantesten Stakeholder-Gruppen ab (Politik, Bildungseinrichtungen, Unternehmen). Zudem wurde eine Stakeholder-übergreifende Kommunikationskategorie hinzugefügt, um den Stakeholder-übergreifenden Charakter von Green Coding und die Notwendigkeit der Kooperation verschiedener Stakeholder-Gruppen zu unterstreichen.

Die Struktur der Handlungsempfehlungen folgt einem systematischen Ansatz, der eine klare und nachvollziehbare Umsetzung ermöglicht. Jede Empfehlung basiert auf einer Datengrundlage, die aus Literaturrecherchen, Umfrageergebnissen, Experten-Workshops und Experten-Interviews abgeleitet wurde. Auf dieser Basis werden spezifische Handlungsempfehlungen formuliert, die auf die identifizierten Herausforderungen eingehen. In der anschließenden Konkretisierung werden die Empfehlungen weiter präzisiert: Es werden konkrete Ziele definiert, spezifische Maßnahmen vorgeschlagen und die erwarteten Ergebnisse erläutert.

Diese strukturierte Herangehensweise stellt sicher, dass jede Empfehlung durch evidenzbasierte Handlungsansätze unterstützt wird und dabei sowohl die praktische Umsetzbarkeit als auch die Messbarkeit der Ergebnisse im Vordergrund stehen.

4.3 Erarbeitung der Handlungsempfehlungen

4.3.1 Politik

Die Politik spielt eine entscheidende Rolle bei der Förderung nachhaltiger Praktiken in der Softwareentwicklung. Insbesondere die Bundesregierung ist in der Lage, gesetzliche und andere politische Rahmenbedingungen zu schaffen, die die Integration von Green Coding als Standard in der Softwareentwicklung unterstützen. Durch klare Vorgaben sowie einheitliche und verbindliche Definitionen kann sie sicherstellen, dass Green Coding nicht nur als Konzept verstanden, sondern auch konsequent in der Praxis umgesetzt wird. Die Schaffung eines einheitlichen Rahmens ermöglicht es, unterschiedliche Interpretationen zu vermeiden, faire Ausgangsbedingungen für alle Stakeholder zu schaffen, die Effizienz zu steigern und die Potenziale für eine nachhaltige digitale Transformation voll auszuschöpfen.

Einheitliche Definition von Green Coding

Datengrundlage:

Während nur 4 % der Befragten das Fehlen einer Definition als Hindernis für die breitflächige Implementierung von Green Coding nannten (E07), zeigt die Umfrageanalyse zum „Verständnis von Green Coding“ (E03, U03) doch, dass die Definition des Begriffs stark variiert. Während einige Teilnehmerinnen und Teilnehmer Green Coding rein technisch und strombezogen verstehen, nutzen andere vage Begriffe wie „Nachhaltigkeit“. Auch unter Expertinnen und Experten besteht keine Einigkeit. Die geringe Anzahl der Befragten, die das Fehlen einer Definition als Hindernis nannten, kann auch darauf zurückgeführt werden, dass innerhalb der Umfrage eine einheitliche Definition präsentiert wurde.

Handlungsempfehlung:

- Die Bundesregierung sollte einen Stakeholder-Prozess initiieren, um eine einheitliche Definition von Green Coding zu entwickeln, die bundesweit Anwendung finden und weiteren Politiken zugrunde liegen kann.

Akteure:

Bundesregierung

Beschreibung:

Eine einheitliche Definition von Green Coding ist entscheidend, um klare Orientierung zu bieten und eine konsistente Anwendung zu fördern. Die Schaffung eines einheitlichen Rahmens ermöglicht es, unterschiedliche Interpretationen zu vermeiden, faire Ausgangsbedingungen für alle Stakeholder zu schaffen, die Effizienz zu steigern und die Potenziale für eine nachhaltige digitale Transformation voll auszuschöpfen. Eine klare Definition, die sowohl technische als auch organisatorische Aspekte berücksichtigt, stellt sicher, dass Green Coding nicht nur als Optimierung für Teams aus Entwicklerinnen und Entwicklern verstanden wird, sondern als ganzheitlicher Ansatz, der auch Management und Produktdesign einbezieht. Dies schafft ein Fundament für nachhaltige Softwareentwicklung und setzt verbindliche Standards.

Konkretisierung der Handlungsempfehlung:

Die Bundesregierung sollte einen Stakeholder-Prozess initiieren, um eine einheitliche Definition von Green Coding zu entwickeln, die bundesweit Anwendung finden und weiteren Politiken zugrunde liegen kann.

Ziel:

Die Erarbeitung einer Definition von Green Coding und ihre Verankerung in relevanten gesetzlichen Regelungen.

Maßnahmen:

- Die Bundesregierung sollte relevante Stakeholder wie Vertreterinnen und Vertreter der Wissenschaft und Organisationen wie die Gesellschaft für Informatik, das Umweltbundesamt, den Bundesverband Green Software, das International Requirements Engineering Board (IREB) und die Internationale Organisation für Normung (ISO) sowie potenziell weitere relevante Akteure einbinden, um eine verbindliche Definition des Begriffs „Green Coding“ zu finden. Die Definition sollte spezifische technische und organisatorische Praktiken festlegen und messbare Kriterien für Green Coding definieren.

Erwartetes Ergebnis:

Eine gesetzlich festgelegte Definition schafft Klarheit und Verbindlichkeit und kann als Grundlage für weitere Maßnahmen und Entwicklungen dienen. Dies fördert die breite Akzeptanz und Anwendung von Green Coding in der Industrie und Forschung und sorgt für ein gemeinsames Ziel unter allen Akteuren.

Transparenzvorschriften

Datengrundlage:

Sowohl die Umfrage als auch die Experten-Interviews zeigen, dass die fehlende Transparenz, das heißt der unzureichende Zugang zu Verbrauchsdaten von IT-Dienstleistungen und Hardware, eine erhebliche Hürde für die Umsetzung nachhaltiger IT-Praktiken wie Green Coding darstellen. Sowohl Unternehmen als auch Einzelpersonen haben Schwierigkeiten, den Stromverbrauch ihrer IT-Infrastruktur präzise zu erfassen. Ohne Daten über die Umweltauswirkungen der genutzten Software und Hardware bleibt der tatsächliche ökologische Fußabdruck unsichtbar, was eine zielgerichtete Reduktion der CO₂-Emissionen erschwert. Um diese Lücke zu schließen, ist die Einführung verpflichtender Transparenzvorschriften und Berichte notwendig. Diese sollten die Umweltauswirkungen von IT-Produkten und -Dienstleistungen über deren gesamten Lebenszyklus abdecken und eine fundierte Grundlage schaffen, um wirksame Maßnahmen zur Reduzierung von Emissionen zu implementieren.

Handlungsempfehlungen:

- Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von eingebetteten Emissionen bei Hardware entlang ihres gesamten Lebenszyklus einführen.
- Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von operativen und eingebetteten Emissionen durch IT-Dienstleistungen und Cloud-Services entlang ihres Lebenszyklus einführen.
- Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von operativen und eingebetteten Emissionen bei Software entlang ihres Lebenszyklus einführen.

Akteure:

Bundesregierung

Beschreibung:

Die drei genannten Handlungsempfehlungen bauen aufeinander auf. Die Grundlage bildet die Handlungsempfehlung, dass die eingebetteten Emissionen von Hardware entlang ihres Lebenszyklus transparent gemacht werden müssen. Erst die Transparenz dieser Werte ermöglicht es den Anbietern von IT-Dienstleistungen und Cloud-Services, ihre operativen und eingebetteten Emissionen zu berechnen und in einem nächsten Schritt transparent zu machen. Folglich können Softwarehersteller ihre operativen und eingebetteten Emissionen erfassen und transparent machen, wenn sie entweder direkt auf Hardware zugreifen oder IT-Dienstleistungen und Cloud-Services nutzen. Die fehlende Transparenz bezüglich der CO₂-Emissionen und des Stromverbrauchs von Hardware, von IT-Dienstleistungen und Cloud-Services sowie von Software erschwert eine fundierte Entscheidungsfindung in Unternehmen und im öffentlichen Sektor. Verbindliche politische Vorgaben zur Offenlegung dieser Daten können maßgeblich zur Verbreitung nachhaltiger Praktiken wie Green Coding beitragen. Durch gesetzliche Vorgaben zur Offenlegung von Emissionen

entlang des Lebenszyklus können Unternehmen und Einzelpersonen besser informierte Entscheidungen treffen und ihren ökologischen Fußabdruck selbst erfassen und durch bewusste Konsumententscheidungen reduzieren. Die Bundesregierung sollte hier eine zentrale Rolle spielen, indem sie gesetzliche Transparenzvorgaben für Hardwarehersteller, IT-Dienstleister und Cloud-Anbieter sowie Softwareproduzenten festlegt. Durch die Integration von Berichten und die Verpflichtung zur Angabe operativer und eingebetteter Emissionen kann der Markt gezielt in Richtung nachhaltiger IT-Lösungen gelenkt werden.

Konkretisierung der Handlungsempfehlungen:

Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von eingebetteten Emissionen bei Hardware entlang ihres gesamten Lebenszyklus einführen.

Ziel:

Unternehmen geben die eingebetteten Emissionen eines Produkts entlang seines gesamten Lebenszyklus transparent an.

Maßnahmen:

- Die Bundesregierung sollte eine Expertenkommission einsetzen, die eine Richtlinie und/oder einen Standard für die transparente Berichterstattung von eingebetteten Emissionen erarbeitet.
- Die Bundesregierung sollte gesetzliche Vorgaben entwickeln, die Hardwarelieferanten verpflichten, die eingebetteten Emissionen ihrer Produkte offenzulegen. Diese Angaben sollten den gesamten Lebenszyklus der Hardware umfassen, von der Produktion über den Transport bis zum Recycling bzw. zur Entsorgung.

Erwartetes Ergebnis:

Die verpflichtende Offenlegung der eingebetteten Emissionen von Hardware ermöglicht es Unternehmen sowie Verbraucherinnen und Verbrauchern, die ökologischen Auswirkungen der Produkte über deren gesamten Lebenszyklus genau zu bewerten. Dadurch können Kaufentscheidungen mit Fokus auf mehr Nachhaltigkeit auf fundierter Grundlage getroffen werden.

Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von operativen und eingebetteten Emissionen durch IT-Dienstleistungen und Cloud-Services entlang ihres Lebenszyklus einführen.

Ziel:

Die operativen und eingebetteten Emissionen von IT-Dienstleistungen, insbesondere bei Cloud-Anbietern, werden transparent dargestellt.

Maßnahmen:

- Die Bundesregierung sollte die Entwicklung und Implementierung von standardisierten Berichtsformaten in Auftrag geben, die es der Nutzerschaft ermöglichen, den Stromverbrauch und die Emissionen ihrer genutzten Dienstleistungen zu verstehen und zu vergleichen. Hier könnten schon bestehende Berichte erweitert werden.
- Die Bundesregierung sollte gesetzliche Vorgaben einführen, die IT-Dienstleister verpflichten, ihre operativen und eingebetteten Emissionen je Dienstleistung transparent zu machen.
- Die Europäische Kommission sollte die EU-Taxonomie erweitern, um nicht nur Rechenzentren, sondern auch Softwarelösungen in die Klassifizierung mit einzubeziehen.

Erwartetes Ergebnis:

Durch die verpflichtende Offenlegung der Emissionen von IT-Dienstleistungen und Cloud-Services erhalten Unternehmen sowie Verbraucherinnen und Verbraucher transparente Informationen zu operativen und eingebetteten Emissionen. Dies ermöglicht es ihnen, den Stromverbrauch und CO₂-Fußabdruck der genutzten Dienstleistungen zu vergleichen und emissionsärmere Anbieter zu wählen. Die erweiterte EU-Taxonomie schafft zusätzlich einen Rahmen, der auch Softwarelösungen umfasst, und fördert damit die Entwicklung nachhaltiger IT-Lösungen.

Die Bundesregierung sollte gesetzliche Verpflichtungen zur Transparenz von operativen und eingebetteten Emissionen bei Software entlang ihres Lebenszyklus einführen.

Ziel:

Die operativen und eingebetteten Emissionen von Software entlang ihres Lebenszyklus werden transparent dargestellt.

Maßnahmen:

- Die Bundesregierung sollte die Entwicklung und Implementierung eines standardisierten Berichtsformats in Auftrag geben, um Emissionen von Software entlang des gesamten Lebenszyklus zu erfassen und transparent öffentlich zu machen.
- Die Bundesregierung sollte gesetzliche Vorgaben einführen, die Softwarehersteller verpflichten, die operativen und eingebetteten Emissionen ihrer Software entlang des Lebenszyklus transparent zu machen.

Erwartetes Ergebnis:

Die verpflichtende Berichterstattung über Emissionen von Software entlang ihres Lebenszyklus erhöht das Bewusstsein für den ökologischen Fußabdruck der digitalen Infrastruktur bei Herstellern sowie bei Verbraucherinnen und Verbrauchern. Dies fördert nachhaltigere IT-Entscheidungen und schafft so Anreize zur Reduktion von CO₂-Emissionen.

Schaffung von Anreizen für Unternehmen

Datengrundlage:

Die Analyse zeigt, dass ein wesentlicher Grund für die mangelnde Verbreitung von Green Coding in der Softwareentwicklung in fehlenden ökonomischen Anreizen liegt (EU7). Ein Drittel der befragten Expertinnen und Experten nennt wirtschaftliche Prioritäten und den fehlenden Druck zur Implementierung nachhaltiger Praktiken als Hauptbarriere (EU8). Ohne finanzielle oder andere Anreize nehmen viele Unternehmen den unmittelbaren Nutzen der Anwendung von Green Coding zu selten wahr, was die Einführung und Akzeptanz erheblich erschwert. Expertinnen und Experten betonen, dass Unternehmen viel mehr auf klare Vorgaben reagieren, als freiwillig nachhaltige Maßnahmen zu ergreifen. Dabei kann die öffentliche Hand eine Vorreiterrolle einnehmen, indem sie gezielt Software erwirbt und einsetzt, die klar definierten Nachhaltigkeitskriterien entspricht. Dies kann die Nachfrage nach nachhaltiger Software anregen und so den Einsatz von Green Coding incentivieren. Auch geben die Befragten an, dass erfolgreiche Anwendungsfälle für Green Coding fehlen. Insbesondere für Unternehmen, die keine Vorreiterrolle einnehmen und die Anwendung von Methoden und Praktiken bevorzugen, deren Nutzen bereits gezeigt werden konnte, kann die nachvollziehbare und ansprechende Kommunikation erfolgreicher Anwendungsfälle zu einer Adaption von neuen Praktiken beitragen.

Handlungsempfehlungen:

- Die Bundesregierung sollte eine Förderung für Green-Coding-Pilotprojekte einsetzen.
- Die Bundesregierung sollte Maßnahmen zur Optimierung von Software für die Zertifizierung mit dem Blauen Engel finanziell fördern.
- Die Bundesregierung sollte den Erwerb von Software (bzw. Softwarelizenzen), die mit dem Blauen Engel zertifiziert ist, finanziell fördern.
- Die Bundesregierung sollte die Integration von Nachhaltigkeitskriterien für Software in der öffentlichen Beschaffung gesetzlich verpflichtend machen.

Akteure:

Bundesregierung

Beschreibung:

Die Schaffung gezielter ökonomischer Anreize ist von zentraler Bedeutung, um die Implementierung von Green Coding in der Softwareentwicklung zu beschleunigen. Viele Unternehmen sind zögerlich, nachhaltige Praktiken zu übernehmen, da sie die wirtschaftlichen Vorteile nicht hinreichend wahrnehmen. Durch die finanzielle Förderung und Zertifizierungen der Entwicklung und des Erwerbs nachhaltiger Software können nachhaltige Entwicklungen gefördert und zu einem Standard gemacht werden. Darüber hinaus kann der öffentliche Sektor eine Vorreiterrolle einnehmen. Durch die Integration von Nachhaltigkeitskriterien bei der Beschaffung von Software

können Softwareanbieter sensitiviert werden, Green Coding in der Softwareentwicklung anzuwenden. Darüber hinaus kann die mittelfristige Forderung, dass Software, die im öffentlichen Sektor erworben werden soll, mit dem Blauen Engel ausgezeichnet sein muss, die nachhaltige Softwareentwicklung verstetigen.

Konkretisierung der Handlungsempfehlungen:

Die Bundesregierung sollte eine Förderung für Green-Coding-Pilotprojekte einsetzen.

Ziel:

Verbundprojekte von Unternehmen mit Hochschulen und Bildungseinrichtungen mit dem Ziel der Anwendung von Green Coding in Pilotprojekten werden finanziell gefördert.

Maßnahmen:

- Die Bundesregierung sollte finanzielle Mittel zur Förderung von Pilotprojekten bereitstellen, die Green-Coding-Praktiken in der Praxis einsetzen und evaluieren.
- Die Bundesregierung sollte die Geförderten im Rahmen der Förderung verpflichten, Erkenntnisse dieser Pilotprojekte zu veröffentlichen, um Best Practices und erfolgreiche Ansätze öffentlich zu teilen.

Erwartetes Ergebnis:

Die Pilotprojekte demonstrieren messbare Ergebnisse durch Green Coding. Die veröffentlichten Ergebnisse bieten Best Practices und konkrete Anwendungsszenarien, die anderen Unternehmen umsetzbare Ansätze liefern. Diese erfolgreichen Beispiele ermutigen weitere Unternehmen zur Einführung von Green Coding, was langfristig zur Standardisierung stromeffizienter Softwareentwicklungspraktiken führt und die Nachhaltigkeit in der IT-Branche fördert.

Die Bundesregierung sollte Maßnahmen zur Optimierung von Software für die Zertifizierung mit dem Blauen Engel finanziell fördern.

Ziel:

Unternehmen werden finanziell bei Maßnahmen der Softwareoptimierung unterstützt, mit dem Ziel, die Software mit dem Blauen Engel zu zertifizieren.

Maßnahmen:

- Die Bundesregierung sollte ein spezielles Förderprogramm einführen, das moderate finanzielle Zuschüsse für Unternehmen bereitstellt, um ihre Software an die Anforderungen des Blauen Engels anzupassen.
- Die Bundesregierung sollte moderate finanzielle Unterstützung für Unternehmen bereitstellen, um die Kosten für externe Audits und Zertifizierungsverfahren zu senken, die notwendig sind, um die Konformität ihrer Software mit den Kriterien des Blauen Engels nachzuweisen.

Erwartetes Ergebnis:

Das Ziel dieser Maßnahme ist, Unternehmen durch finanzielle Unterstützung zu ermutigen, gezielte Maßnahmen zur Optimierung ihrer Software in Richtung Umweltfreundlichkeit und Energieeffizienz umzusetzen, um die Blaue-Engel-Zertifizierung zu erlangen.

Die Bundesregierung sollte den Erwerb von Software (bzw. Softwarelizenzen), die mit dem Blauen Engel zertifiziert ist, finanziell fördern.

Ziel:

Unternehmen werden finanziell unterstützt beim Erwerb von Software, die mit dem Blauen Engel ausgezeichnet ist.

Maßnahmen:

- Die Bundesregierung sollte moderate finanzielle Zuschüsse oder Fördermittel für Unternehmen bereitstellen, die Software erwerben möchten, die den Blauen Engel trägt. Diese Programme könnten prozentuale Rabatte oder fixe Beträge anbieten, um die Kosten für die Anschaffung von zertifizierter, umweltfreundlicher Software zu senken.
- Die Bundesregierung sollte steuerliche Vergünstigungen einführen, wie zum Beispiel Abschreibungsmöglichkeiten oder Steuererleichterungen, für Unternehmen, die Software mit dem Blauen Engel erwerben.

Erwartetes Ergebnis:

Unternehmen werden durch finanzielle Anreize und steuerliche Vergünstigungen motiviert, vermehrt Software mit dem Blauen Engel zu erwerben, was zu einer gesteigerten Verbreitung umweltfreundlicher Softwarelösungen und einem stärkeren Fokus auf Nachhaltigkeit in der IT-Branche führt.

Die Bundesregierung sollte die Integration von Nachhaltigkeitskriterien für Software in der öffentlichen Beschaffung gesetzlich verpflichtend machen.

Ziel:

Mittelfristig soll ein Markt für nachhaltige Software entstehen, in dem Green Coding als Standard in der Softwareentwicklung gilt und in dem der öffentliche Sektor ausschließlich Software mit dem Blauen Engel beschafft. Da derzeit noch nicht ausreichend zertifizierte Software verfügbar ist, sollten zunächst Nachhaltigkeitskriterien in die öffentliche Beschaffung integriert werden, um Unternehmen zu nachhaltigen Praktiken zu motivieren und ihre Wettbewerbsfähigkeit zu stärken.

Maßnahmen:

- Die Bundesregierung sollte ein Expertengremium einsetzen, das verbindliche Nachhaltigkeitskriterien für Software definiert. Diese Vorgaben dienen als Grundlage für öffentliche Ausschreibungen, um Transparenz und Vergleichbarkeit zu gewährleisten.
- Die Bundesregierung sollte die öffentliche Hand dazu verpflichten, die identifizierten Nachhaltigkeitskriterien in

den Beschaffungsprozess zu integrieren, sodass Softwareanbieter in öffentlichen Ausschreibungen nachweisen müssen, dass ihre Produkte die vorgegebenen Kriterien erfüllen.

- Die Bundesregierung sollte eine verbindliche Frist einführen, bis zu der alle neuen Softwarebeschaffungen im öffentlichen Sektor nur noch Produkte betreffen dürfen, die mit dem Blauen Engel zertifiziert sind.

Erwartetes Ergebnis:

Durch die Einführung verbindlicher Nachhaltigkeitskriterien und die spätere Verpflichtung zur Blauen-Engel-Zertifizierung wird der öffentliche Sektor langfristig nur noch Software beschaffen, die den Nachhaltigkeitskriterien entspricht. Dies fördert die nachhaltige Softwareentwicklung und stärkt die Wettbewerbsfähigkeit von Anbietern mit zertifizierter Software.

4.3.2 Unternehmen

Die Privatwirtschaft, das heißt im Konkreten ihr Konsum, ihre Produktion und ihre Innovation in Bezug auf IT-Dienstleistungen, spielt eine zentrale Rolle bei der Umstellung auf Green-Coding-Praktiken. Die Privatwirtschaft ist der größte Konsument dieser Lösungen, für die Bereitstellung von IT-Dienstleistungen verantwortlich und auch in vielen Fällen die Quelle für Innovationen und technologische Fortschritte. Aus diesem Grund spielen Unternehmen eine essenzielle Rolle bei der Reduktion der CO₂-Emissionen des IT-Sektors. Dies betrifft sowohl IT-Produktionsfirmen, die sich mit Innovationen und Entwicklung beschäftigen und Lösungen bereitstellen, als auch solche Unternehmen, die diese Lösungen einkaufen, konsumieren und so die Marktnachfrage antreiben und entsprechende Angebote anreizen.

Stärkung des Bewusstseins

Datengrundlage:

Während politische Maßnahmen wie Förderungen, Pilotprogramme und Subventionen wesentliche Instrumente sind, bieten Green-Coding-Lösungen auch unternehmensinterne Vorteile, die erkannt und genutzt werden sollten. Es scheint dabei einfach, keine Verantwortung zu übernehmen und nur auf die Politik zu zeigen. Vorteile für Unternehmen liegen in Kosteneinsparungen durch einen effizienteren Ressourceneinsatz und in positiven Effekten für die Unternehmensreputation, was über eine gesteigerte Nachfrage zu ökonomischen Gewinnen führen kann.

Handlungsempfehlungen:

- Unternehmen, die Software entwickeln, sollten Kennzahlen auf Softwareprodukt- und Softwareprojektebene einführen, die den Stromverbrauch und die operativen und eingebetteten Emissionen von Software entlang ihres Lebenszyklus erfassen.

- Unternehmen, die Software einkaufen, sollten Transparenz in Bezug auf den Stromverbrauch und die operativen und eingebetteten Emissionen für Hardware, IT-Dienstleistungen, Cloud-Services und Software bei den Herstellern einfordern.

Akteure:

Unternehmen, die Software entwickeln, und Unternehmen, die Software einkaufen

Beschreibung:

Unternehmen sollten Strategien entwickeln, um Green Coding als Standardpraxis zu etablieren. Die Prüfung und Berichterstattung über nicht finanzielle Kennzahlen und die Integration nachhaltiger Praktiken in die Wertschöpfungskette können dazu beitragen, Green Coding als Standard zu verankern. Dies fördert die Unternehmensverantwortung und unterstützt eine nachhaltige digitale Transformation.

Konkretisierung der Handlungsempfehlungen:

Unternehmen, die Software entwickeln, sollten Kennzahlen auf Softwareprodukt- und Softwareprojektebene einführen, die den Stromverbrauch und die operativen und eingebetteten Emissionen von Software entlang ihres Lebenszyklus erfassen.

Ziel:

Unternehmen erfassen die Nachhaltigkeit ihrer Software über Kennzahlen, prüfen diese in regelmäßigen Abständen und nehmen die Zahlen in ihre Berichterstattung auf. Auf diese Weise können die Nachhaltigkeitsaspekte handlungsleitend wirken.

Maßnahmen:

- Unternehmen, die Software entwickeln, sollten Kennzahlen definieren, um den Stromverbrauch und die operativen und eingebetteten Emissionen der Software entlang ihres Lebenszyklus zu messen. Die Kennzahlen sollten vom Unternehmen in regelmäßigen Abständen bewertet werden.
- Unternehmen, die Software entwickeln, sollten die Kennzahlen pro Service bzw. pro Produkt in ihren Geschäftsberichten ausweisen.

Erwartetes Ergebnis:

Durch die Einführung und Auswertung von Kennzahlen zum Stromverbrauch und zu den operativen und eingebetteten Emissionen von Software wird die Nachhaltigkeit von Software zunehmend transparent. Über die Zeit können Unternehmen, die Software entwickeln, Verbesserungen oder Verschlechterungen in Bezug auf den Stromverbrauch und die operativen und eingebetteten Emissionen sehen und bewerten. Darüber hinaus kann durch die Berichterstattung ein gewisser Druck entstehen, die Kennzahlen über die Zeit zu verbessern.

Unternehmen, die Software einkaufen, sollten Transparenz in Bezug auf den Stromverbrauch und die operativen und eingebetteten Emissionen für Hardware, IT-Dienstleistungen, Cloud-Services und Software bei den Herstellern einfordern.

Ziel:

Durch zunehmenden Druck auf die Hersteller erhöht sich die Transparenz bezüglich der Emissionen von Produkten bzw. Services über den gesamten Lebenszyklus.

Maßnahmen:

- Unternehmen, die Software einkaufen, sollten ausdrücklich Transparenz hinsichtlich des Stromverbrauchs und der operativen und eingebetteten Emissionen von erworbener und zu beschaffender Hardware, IT-Dienstleistungen, Cloud-Services und Software von den Herstellern einfordern.

Erwartetes Ergebnis:

Durch die konkrete Nachfrage nach weniger Hardware, IT-Dienstleistungen, Cloud-Services und Software mit hoher Stromintensität werden Anreize für Hersteller und Anbieter gesetzt, ihre Produkte und Dienstleistungen so zu entwickeln, dass sie mit weniger Stromverbrauch produziert werden und im Betrieb weniger Strom verbrauchen.

Unternehmensinterne Ausbildungen und Schulungen

Datengrundlage:

Die Analyse der Umfrage und der Experten-Workshops zeigt, dass es Vorbehalte gegen Green Coding gibt und Methoden und Praktiken, die in wissenschaftlichen Arbeiten als erfolgversprechend dargestellt werden, oft nicht bekannt sind. Viele Befragte scheinen überzeugt, dass Green Coding nur geringe Effekte hat und wenig Wirkung zeigt. Hingegen zeigen Beispiele wie Carbon Aware Computing, dass durch die räumliche oder zeitliche Verschiebung des Softwareeinsatzes in Regionen mit weniger CO₂-intensiver Stromerzeugung deutliche Einsparungen bei den strombezogenen Emissionen erreicht werden können. Diese Wissenslücke sollte durch umfassende Aufklärungs- und Schulungsinitiativen geschlossen werden.

Handlungsempfehlungen:

- Unternehmen sollten interne und zielgruppengerechte Bildungsangebote zum Thema Green Coding bereitstellen, um unternehmensinternes Wissen im Bereich Green Coding aufzubauen.
- Unternehmen sollten Angebote erarbeiten, um ihre Führungskräfte für Green Coding zu sensibilisieren.

Akteure:

Unternehmen, die Software entwickeln, und Unternehmen, die Software einkaufen

Beschreibung:

Die Überwindung von Vorbehalten gegen Green Coding und die effektive Kommunikation der wissenschaftlich belegbaren Vorteile von Green Coding sind entscheidend, um die erheblichen Stromeinsparpotenziale zu realisieren. Durch gezielte Schulungsangebote in Unternehmen, die verstärkte Ausbildung von Beschäftigten und die Weiterbildung von IT-Entscheiderinnen und -Entscheidern können diese Ziele erreicht werden. Diese Maßnahmen tragen dazu bei, das Verständnis und die Anwendung von Green Coding zu fördern.

Konkretisierung der Handlungsempfehlungen:

Unternehmen sollten interne und zielgruppengerechte Bildungsangebote zum Thema Green Coding bereitstellen, um unternehmensinternes Wissen im Bereich Green Coding aufzubauen.

Ziel:

Durch zielgruppengerechte Bildungsangebote zum Thema Green Coding sollen das Wissen und die Kompetenz der Mitarbeiterinnen und Mitarbeiter im Bereich Green Coding ausgebaut werden.

Maßnahmen:

- Unternehmen, die Software entwickeln, sollten spezifische und auf die Zielgruppe zugeschnittene Schulungen anbieten, die sowohl technische als auch organisatorische Aspekte abdecken. Auf diese Weise kann Green Coding in Entwicklungsprogramme integriert werden, um eine nachhaltige Softwareentwicklung in den Entwicklungsprozessen zu verankern.
- Unternehmen, die Software einkaufen, sollten Schulungsprogramme entwickeln, die es den Beschäftigten ermöglichen, IT-Lieferanten anhand ihrer Green-Coding-Praktiken zu bewerten und so nachhaltigere Entscheidungen in der Beschaffung zu treffen.

Erwartetes Ergebnis:

Durch gezielte interne Bildungsangebote mit Fokus auf die wirtschaftlichen Vorteile, die in Ressourceneinsparungen liegen, sowie die Wahrnehmung von Fördermöglichkeiten durch gezielte Bildungsmaßnahmen innerhalb von Unternehmen können das Wissen über und die Anwendung von Green-Coding-Praktiken im beruflichen Umfeld gesteigert werden.

Unternehmen sollten Angebote erarbeiten, um ihre Führungskräfte für Green Coding zu sensibilisieren.

Ziel:

Durch die Steigerung des Bewusstseins und die Unterstützung für nachhaltige Software und somit für Green Coding insbesondere auf Führungsebene werden nachhaltige Praktiken unternehmensweit verankert.

Maßnahmen:

- Verbände und Beratungsunternehmen sollten Workshops und Seminare entwickeln, die gezielt Führungskräfte ansprechen und die Bedeutung von Green Coding für den langfristigen Geschäftserfolg sowie für die Erreichung der Umwelt- und Klimaziele verdeutlichen. Die Schulungen sollten auch darauf abzielen, Führungskräfte für die Integration von Green Coding in die unternehmensweite CSR-Strategie (Corporate Social Responsibility) zu gewinnen.

Erwartetes Ergebnis:

Durch diese Maßnahmen werden das Bewusstsein und das Engagement der Führungsebene für Green Coding gesteigert. Dies kann die Umsetzung nachhaltiger Praktiken im gesamten Unternehmen vorantreiben und die Erreichung von ökologischen und wirtschaftlichen Zielen unterstützen.

4.3.3 Bildungseinrichtungen

Bildungseinrichtungen, insbesondere Hochschulen und Universitäten, aber auch Verbände, Volkshochschulen und andere Lernstätten spielen eine entscheidende Rolle bei der Förderung nachhaltiger Praktiken in der Softwareentwicklung. Hier bietet sich die Chance, durch die Integration nachhaltiger Prinzipien in Lehrpläne und Forschungsprojekte eine zielführende Ausbildung zukünftiger Fachkräfte zu garantieren.

Integration von Green Coding in Curricula von Hochschulen

Datengrundlage:

Die Analyse der Antworten auf die Frage „Wie gut sind Sie mit dem Thema ‚Green Coding‘ vertraut?“ (OU2) hat ergeben, dass der Begriff „Green Coding“ den Teilnehmerinnen und Teilnehmern der öffentlichen Umfrage nicht durchgängig bekannt ist. Zudem werden fehlendes Wissen und fehlende Schulungen von 70 % der Expertinnen und Experten als Hindernis für die breitflächige Implementierung von Green Coding genannt (EU7).

Handlungsempfehlungen:

- Hochschulen/Universitäten sollten Green Coding in die akademischen Curricula von informatikorientierten Studiengängen integrieren.
- Hochschulen/Universitäten sollten Kurse und Seminare anbieten, um auch Studentinnen und Studenten in nicht informatikorientierten Studiengängen für nachhaltige IT und Software zu sensibilisieren.

Akteure:

Hochschulen/Universitäten

Beschreibung:

Die Integration von Green Coding in Schulungs- und Bildungsprogramme ist entscheidend, um eine breite Akzeptanz und Anwendung dieser nachhaltigen Entwicklungsmethoden zu gewährleisten. Durch die gezielte Ausbildung und Sensibilisierung von Entwicklerinnen und Entwicklern sowie von Forscherinnen und Forschern kann sichergestellt werden, dass Green Coding fest in der Praxis verankert wird und so einen Beitrag zur nachhaltigen digitalen Transformation leistet.

Konkretisierung der Handlungsempfehlungen:

Hochschulen/Universitäten sollten Green Coding in die akademischen Curricula von informatikorientierten Studiengängen integrieren.

Ziel:

Green-Coding-Praktiken werden zu einem elementaren Bestandteil in Curricula von informatikorientierten Studiengängen an Hochschulen und Universitäten.

Maßnahmen:

- Hochschulen und Universitäten sollten Green Coding als festen Bestandteil in ihre Curricula von informatikorientierten Studiengängen integrieren, indem sie spezielle Module oder modulübergreifende Konzepte entwickeln.

Erwartetes Ergebnis:

Studentinnen und Studenten informatikorientierter Studiengänge erwerben fundierte Kenntnisse in nachhaltiger Softwareentwicklung durch Green Coding, die sie direkt in der Praxis anwenden können, was langfristig zu einer breiten Umsetzung von Green-Coding-Praktiken im Beruf führen kann.

Hochschulen/Universitäten sollten Kurse und Seminare anbieten, um auch Studentinnen und Studenten in nicht informatikorientierten Studiengängen für nachhaltige IT und Software zu sensibilisieren.

Ziel:

Durch Angebote zur Weiterbildung zu nachhaltiger IT und Software in nicht informatikorientierten Studiengängen wie beispielsweise Wirtschaftswissenschaften, Umweltwissenschaften und Politikwissenschaften sowie anderen Fachrichtungen wird ein breites Spektrum der Studentinnen und Studenten für das Thema nachhaltige IT und Software sensibilisiert.

Maßnahmen:

- Hochschulen/Universitäten sollten interdisziplinäre Ansätze fördern, bei denen nachhaltige IT und Software auch in anderen Fachbereichen thematisiert wird.
- Hochschulen/Universitäten sollten Seminare, Workshops und Vorlesungen zu nachhaltiger IT und Software anbieten, um so zur Sensibilisierung eines breiteren Publikums angehender Führungskräfte und Expertinnen und Experten beizutragen.

Erwartetes Ergebnis:

Durch diese Maßnahmen soll Bewusstsein und Verständnis für nachhaltige IT und Software bei Studentinnen und Studenten verschiedener Disziplinen verstärkt werden, wodurch die Bedeutung nachhaltiger IT-Lösungen in unterschiedliche berufliche und gesellschaftliche Kontexte weitergetragen wird.

Mehr Bildungsangebote

Datengrundlage:

Die Umfrageergebnisse zeigen eine deutliche Bildungslücke im Bereich Green Coding. Viele Teilnehmerinnen und Teilnehmer der öffentlichen Umfrage gaben bei der Frage „Wie gut sind Sie mit dem Thema ‚Green Coding‘ vertraut?“ (OU2) an, mit dem Thema wenig oder nicht ausreichend vertraut zu sein. In der weiteren Analyse gaben Befragte fehlende Schulungs- und Weiterbildungsangebote als Hindernis für die breite Anwendung von Green Coding an (EU7). Es wurde klar, dass Green Coding nicht ausreichend vermittelt wird und stärker in die öffentliche Wahrnehmung gerückt werden muss, um breitere Akzeptanz zu finden und praxisnahe Lösungen zu etablieren.

Handlungsempfehlungen:

- Volkshochschulen, Verbände und weitere Bildungseinrichtungen sollten Schulungsangebote zu Green Coding entwickeln und organisieren.

Akteure:

Volkshochschulen, Verbände, Bildungseinrichtungen

Beschreibung:

Zusätzliche Bildungsangebote sollten sowohl das Bewusstsein für Green Coding erhöhen als auch fundierte, wissenschaftlich gestützte Kenntnisse vermitteln. Dies erfordert die Entwicklung leicht zugänglicher Schulungsressourcen und Schulungsangebote, um unterschiedliche Zielgruppen zu erreichen – von Entwicklerinnen und Entwicklern sowie IT-Fachleuten bis hin zu Entscheiderinnen und Entscheidern in Unternehmen und Politik. Ein breiteres Wissen in der Bevölkerung kann zudem generell dazu führen, dass Verbraucherinnen und Verbraucher mehr nachhaltige Software nachfragen.

Konkretisierung der Handlungsempfehlungen:

Volkshochschulen, Verbände und weitere Bildungseinrichtungen sollten Schulungsangebote zu Green Coding entwickeln und organisieren.

Ziel:

Durch Angebote von Volkshochschulen und Verbänden werden Schulungsressourcen und -angebote zu Green Coding für eine breite Zielgruppe zugänglich gemacht.

Maßnahmen:

- Fachverbände wie die Gesellschaft für Informatik (GI), Bitkom und der Bundesverband Green Software (BvGS) sollten Programme und Kurse zu Green Coding entwickeln und kostenlos zur Verfügung stellen.
- NGOs und Fachverbände sollten Online-Kurse, Webinare und Workshops zu Green Coding organisieren und sich dabei sowohl an Entwicklerinnen und Entwickler als auch an Führungskräfte und politische Entscheidungsträgerinnen und -träger richten.

Erwartetes Ergebnis:

Durch die Schulungsangebote werden der allgemeine Wissensstand und die praktische Anwendung von Green-Coding-Praktiken erhöht und sie finden eine breitere Akzeptanz in verschiedenen Sektoren.

4.3.4 Kommunikation und Zusammenarbeit

Das Zusammenspiel der verschiedenen Akteure sowie die allgemeine Bewusstseinssteigerung sind entscheidend für die erfolgreiche Verankerung von Green Coding in der Praxis. Aus diesem Grund wurden Kommunikation und Zusammenarbeit auch als eigener Punkt in diese Reihe der Handlungsempfehlungen aufgenommen. Um den Wissenstransfer zu fördern und den gesellschaftlichen Diskurs voranzutreiben, braucht es effektive Öffentlichkeitsarbeit, die Green Coding zugänglicher macht und das Bewusstsein stärkt. Dabei spielen sichtbare Erfolgsgeschichten eine wichtige Rolle, um den Mehrwert nachhaltiger Softwareentwicklung für Umwelt und Wirtschaft zu demonstrieren. Die gezielte Medienpräsenz und die Verbreitung konkreter Beispiele stärken nicht nur das Bewusstsein in der Öffentlichkeit, sondern schaffen auch die Grundlage für politischen und gesellschaftlichen Druck, der Green Coding in Organisationen und Unternehmen verankert.

Erfolgsberichte

Datengrundlage:

Ein wesentlicher Aspekt, der in den Umfragen und Workshops häufig genannt wurde, ist das Fehlen konkreter Erfolgsgeschichten und messbarer Ergebnisse, die als Beispiele und Vorbilder für erfolgreiche Green-Coding-Projekte dienen können. Diese Erfolgsberichte sind entscheidend, um den Mehrwert von Green Coding zu demonstrieren und als Motivation für andere Unternehmen und Organisationen zu dienen.

Handlungsempfehlungen:

- Verbände und Behörden sollten konkrete Beispiele für die Anwendung von Green Coding sammeln, aufbereiten und öffentlich verfügbar machen.
- Unternehmen sollten eigene Erfolgsgeschichten öffentlich teilen und über soziale Medien und Informationskanäle eine breite Öffentlichkeit über die Erfolgsgeschichten informieren.

- Ministerien und Verbände sollten Preise für herausragende Green-Coding-Praktiken verleihen und sowohl den Preis als auch die Preisverleihung öffentlichkeitswirksam bewerben.

Akteure:

Unternehmen, Ministerien, Verbände, NGOs

Beschreibung:

Die Kommunikation von Erfolgsgeschichten spielt eine zentrale Rolle bei der Sensibilisierung der Öffentlichkeit und von Entscheidungsträgerinnen und -trägern für Green Coding. Unternehmen sollten ihre positiven Erfahrungen öffentlich teilen, um das Vertrauen und die Nachfrage zu steigern. Staatliche Anerkennung in Form von Preisen hilft, das Thema stärker ins öffentliche Bewusstsein zu rücken. Diese Maßnahmen erhöhen das Bewusstsein und die Motivation, Green Coding auf breiter Ebene umzusetzen.

Konkretisierung der Handlungsempfehlungen:

Verbände, NGOs und Behörden sollten konkrete Beispiele für die Anwendung von Green Coding sammeln, aufbereiten und öffentlich verfügbar machen.

Ziel:

Die Bereitstellung einer umfassenden Liste konkreter Beispiele und deren messbarer Wirkung soll den Erfolg und die Vorteile von Green Coding aufzeigen.

Maßnahmen:

- Verbände wie zum Beispiel die Gesellschaft für Informatik oder der Bundesverband Green Software sollten Erfolgsgeschichten sammeln, aufbereiten und öffentlich zugänglich machen.

Erwartetes Ergebnis:

Die Erhöhung des Verständnisses und der Akzeptanz von Green Coding wird durch konkrete, nachweisbare Beispiele gefördert, die als Best Practices dienen und zur breiteren Anwendung dieser Praktiken anregen können.

Unternehmen sollten eigene Erfolgsgeschichten öffentlich teilen und über soziale Medien und Informationskanäle eine breite Öffentlichkeit über die Erfolgsgeschichte informieren.

Ziel:

Die Sichtbarkeit von und das Bewusstsein für Green-Coding-Praktiken sollen durch das Teilen von konkreten Erfolgsgeschichten innerhalb und außerhalb der Branche erhöht werden.

Maßnahmen:

- Unternehmen sollten ihre Erkenntnisse in Bezug auf die Anwendung von Green Coding veröffentlichen, beispielsweise durch eine Präsentation von Erfahrungen und Best Practices.
- Unternehmen sollten Fallstudien und Erfahrungsberichte veröffentlichen, die sowohl technische Details als auch messbare Ergebnisse darstellen.
- Unternehmen sollten soziale Medien, Pressemitteilungen und Branchenkonferenzen nutzen, um Erfolgsgeschichten öffentlichkeitswirksam zu teilen und eine breite Leserschaft zu erreichen.

Erwartetes Ergebnis:

Durch das gezielte Teilen von Erfolgsgeschichten können das Bewusstsein und das Vertrauen bei Stakeholdern, darunter Kunden und Investoren, in die positiven Effekte von Green Coding erhöht werden, was die Nachfrage nach nachhaltigen Softwarelösungen steigert.

Ministerien und Verbände sollten Preise für herausragende Green-Coding-Praktiken verleihen und sowohl den Preis als auch die Preisverleihung öffentlichkeitswirksam bewerben.

Ziel:

Die Anerkennung und Förderung herausragender Leistungen im Bereich nachhaltiger Softwareentwicklung motiviert mehr Unternehmen, sich mit Green Coding zu beschäftigen.

Maßnahmen:

- Ministerien sollten jährliche Preise für die besten Green-Coding-Initiativen und -Lösungen vergeben, um innovative Ansätze öffentlich zu honorieren.
- Ministerien sollten Informationen über die Preisträger in der Breite kommunizieren, um für Unternehmen mehr Anreize zu schaffen, mit dem Preis ausgezeichnet zu werden.
- Ministerien, die Preise verleihen, sollten die Preisverleihung öffentlichkeitswirksam gestalten, um die Reichweite und den Grad der Wahrnehmung des Themas zu maximieren.

Erwartetes Ergebnis:

Die Aussicht auf Preise und Anerkennung erhöht die Motivation bei Unternehmen, sich intensiver mit Green Coding zu beschäftigen. Eine stärkere öffentliche Wahrnehmung der Erfolge im Bereich nachhaltiger Softwareentwicklung kann wiederum andere Unternehmen anreizen, selbst mehr in Green Coding zu investieren.

4.3.5 Keine Handlungsempfehlungen für Messen und Standardisierung

Das Messen des Stromverbrauchs von Software ist essenziell, wenn der Stromverbrauch und die damit einhergehenden Emissionen reduziert werden sollen. Während für den Begriff „Green Coding“ im Rahmen der Handlungsempfehlungen eine einheitliche Definition gefordert wird, existieren im Bereich Messen bereits anwendbare Standards wie zum Beispiel die Software Carbon Intensity (SCI), die seit April 2024 ein ISO-Standard ist, oder auch die ISO 14001 mit dem Zusatz des GHG-Protokolls speziell für Softwareprodukte. Hiermit ist es bereits möglich, Umwelteinflüsse von digitalen Produkten auszuweisen. Auch ist die Messung des Stromverbrauchs und der Emissionen von Software bereits Bestandteil des Blauen Engels. Handlungsempfehlungen beziehen sich daher auf Maßnahmen, die den Blauen Engel explizit integrieren, wie etwa die Förderung des Erwerbs von Software, die mit dem Blauen Engel ausgezeichnet ist. Da die Messung schon in einem anerkannten und weit verbreiteten Standard integriert ist, war es nicht notwendig, dies erneut als separate Handlungsempfehlung aufzuführen.

Darüber hinaus ist es den Autorinnen und Autoren dieses Reports bewusst, dass weitere Umwelteinflüsse wie beispielsweise der Wasserverbrauch und die Landnutzung und damit verbunden der Verlust von Biodiversität existieren, die über die Betrachtung von Strom und CO₂ hinausgehen. Um die Einflüsse von Software auf die Umwelt im Allgemeinen zu erfassen, sollten diese in die Entwicklung von Standards mit einfließen. Diese Aspekte zusätzlich zu adressieren, hätte jedoch den Umfang dieses Reports überschritten.

Auch gibt es bereits bestehende Standardisierungs- und Normungsaktivitäten etwa im Kontext von Green ICT, die sich auf die Ressourcenminimierung von Elektronikprodukten für eine ressourcenbewusste IKT fokussieren (www.greenict.de). Eine weitere Aktivität ist die Open Consultation der International Organization for Standardization (ISO), die im Oktober 2024 angelaufen ist und Normungs- und Standardisierungsbedarfe im Kontext von ressourceneffizienter Software im internationalen Kontext erarbeitet. Diese Aktivitäten verdeutlichen, dass das Thema ressourceneffiziente IKT und Software zunehmend an Bedeutung gewinnt und auf internationaler Ebene Beachtung findet. Die existierenden Initiativen und der begonnene Austausch auf globaler Ebene unterstreichen das wachsende Interesse an der Etablierung von Standards, die zur nachhaltigen Entwicklung in der Software- und IKT-Branche beitragen sollen.

5.

Fazit

Die Ergebnisse des Reports zeigen, dass Green Coding eine zentrale Rolle bei der nachhaltigen Transformation der Informations- und Kommunikationstechnologie spielen kann. Die Anwendung von Green Coding bietet eine vielversprechende Lösung, um Software energieeffizienter zu gestalten, ohne dabei Leistung und Funktionalität zu beeinträchtigen. Die Implementierung von Green-Coding-Praktiken stellt eine bedeutende Möglichkeit dar, den Stromverbrauch und die Emissionen der IKT und der IT-Branche erheblich zu reduzieren. Green Coding bietet somit die Chance, einen wesentlichen Beitrag zum Klimaschutz zu leisten.

Der Report zeigt, dass die umfassende Einführung von Green Coding in der Softwareentwicklung mit verschiedenen Herausforderungen verbunden ist. Eine der zentralen Forschungsfragen war, welche Herausforderungen die Einführung von Green Coding erschweren. Die Ergebnisse des Reports verdeutlichen, dass der wichtigste Faktor das fehlende Wissen und die mangelnde Schulung sind.

Technische und organisatorische Herausforderungen wurden ebenfalls von vielen Befragten hervorgehoben. Dazu gehören der Mangel an technischer Expertise, das Fehlen technischer Lösungen im konkreten Unternehmenskontext sowie bestehende organisatorische Strukturen, die die Implementierung von Green Coding behindern. Zudem sind ökonomische Prioritäten und fehlende Anreize ein wesentliches Hindernis. Weitere Herausforderungen umfassen regulatorische und gesellschaftliche Aspekte, wie das Fehlen gesetzlicher Vorgaben und breiter gesellschaftlicher Unterstützung, eine fehlende einheitliche Definition von Green Coding und fehlende Anwendungsfälle, die die Vorteile von Green Coding verdeutlichen könnten.

In Bezug auf die zweite Forschungsfrage, welche Stakeholder relevant bei der Implementierung von Green Coding sind, identifizierte der Report Politik, Bildungseinrichtungen und Unternehmen als die zentralen Akteure, die eine Schlüsselrolle bei der Umsetzung spielen. Regierungen sind für die Schaffung von regulatorischen Rahmenbedingungen und Anreizen verantwortlich, die nachhaltige IT-Praktiken fördern. Bildungseinrichtungen tragen die Verantwortung dafür, zukünftige Fachkräfte auszubilden und ihnen von Beginn an ein Bewusstsein für Nachhaltigkeit zu vermitteln. Unternehmen sollten durch die Einführung transparenter Nachhaltigkeitsrichtlinien eine Vorreiterrolle einnehmen und aktiv zur Etablierung von Green Coding beitragen – als Produzenten oder als Nachfrager.

Die dritte Forschungsfrage behandelte, welche konkreten Maßnahmen relevante Stakeholder ergreifen sollten, um Green Coding in die breite Anwendung zu bringen. In diesem Report wurden daher auf Basis der durchgeführten empirischen Untersuchungen praxisorientierte Handlungsempfehlungen entwickelt. Diese Empfehlungen, die in Kapitel 5 detailliert und nach den jeweiligen Stakeholdern strukturiert vorgestellt werden, bieten eine fundierte Grundlage, um Green Coding in der Softwareentwicklung zu etablieren. Dadurch kann ein wesentlicher Beitrag zur Reduzierung des Stromverbrauchs in der IT-Branche geleistet werden.

Die Untersuchung der Forschungsfragen zeigt, dass die erfolgreiche Einführung von Green Coding ein Zusammenspiel von Bildung, unternehmerischen Aktivitäten und politischen Maßnahmen erfordert. Diese Koordination kann dazu beitragen, die bestehenden Herausforderungen zu überwinden und nachhaltige Praktiken in der Softwareentwicklung zu etablieren.

Ausblick

Der nächste Schritt besteht in der praktischen Umsetzung der erarbeiteten Handlungsempfehlungen. Im Verlauf dieses Prozesses wird sich herausstellen, inwieweit diese Maßnahmen ausreichen, um den steigenden Stromverbrauch in der IKT-Branche zu begrenzen und die angestrebten Nachhaltigkeitsziele zu erreichen. Eine fortlaufende Evaluierung der Maßnahmen sowie die kontinuierliche Anpassung der Strategien werden entscheidend sein, um den langfristigen Erfolg von Green Coding zu gewährleisten. Angesichts der dynamischen Entwicklungen in der IKT-Branche und der Einführung neuer Technologien wird es notwendig sein, die Maßnahmen fortlaufend zu überprüfen und gegebenenfalls weiterzuentwickeln. Eine kontinuierliche wissenschaftliche Begleitung und weiterführende Forschung sind essenziell, um sicherzustellen, dass Green Coding langfristig einen effektiven Beitrag zur Reduktion des Stromverbrauchs und der CO₂-Emissionen in der IT-Branche leisten kann.

Glossar

Begriff	Definition
Blauer Engel	Ein deutsches Umweltzeichen, das seit 1978 Produkte und Dienstleistungen auszeichnet, die besonders umweltfreundlich sind. Es berücksichtigt dabei den gesamten Lebenszyklus des Produkts, von der Herstellung über die Nutzung bis hin zur Entsorgung. Das Umweltzeichen wurde 2020 auf Software erweitert.
Bytecode	Ein Zwischenprodukt zwischen Quellcode und Maschinencode, das betriebssystemunabhängig von einer speziellen Laufzeitumgebung ausgeführt werden kann.
Bytecode Generator	Ein Programm, das Quellcode in Bytecode umwandelt, der von einer virtuellen Maschine ausgeführt werden kann.
Compiler	Ein Programm, das Quellcode in ausführbaren Maschinencode übersetzt.
Deployment	Der Prozess der Installation, Konfiguration und Aktivierung einer Softwareanwendung oder eines Systems in einer bestimmten Umgebung.
Developer	Eine Person, die Software entwickelt, programmiert und testet.
Demand Shifting	Eine Methode, den Stromverbrauch an die Variabilität der CO ₂ -Intensität der Stromerzeugung anzupassen, indem Arbeiten, die viel Strom verbrauchen, in Zeitspannen oder Regionen verlagert werden, in denen die Stromerzeugung weniger CO ₂ -intensiv ist.
Demand Shaping	Demand Shaping bezeichnet die proaktive Gestaltung und Steuerung von Rechenlasten, um die Energienachfrage gezielt an die Verfügbarkeit von CO ₂ -arme Energiequellen anzupassen, oft durch den Einsatz intelligenter Algorithmen und Echtzeitdaten.
Eingebettete Emissionen	Eingebettete Emissionen (englisch: Embodied Emissions) sind die Treibhausgasemissionen, die während des gesamten Lebenszyklus eines Produkts entstehen, einschließlich der Phasen von Rohstoffgewinnung, Produktion, Transport, Nutzung und Entsorgung.
FOSS	Free and Open Source Software
Hardwareressource	Physische Komponenten eines Computers wie CPU, Speicher oder Festplatten, die für die Ausführung von Software genutzt werden.
Linker	Ein Programm, das verschiedene Programmteile zu einer ausführbaren Datei zusammenführt.
Microservices-Architektur	Der Begriff „Microservices-Architektur“ bezieht sich auf einen Architekturstil, bei dem eine Anwendung aus einer Sammlung von kleinen, autonomen Diensten besteht, die unabhängig voneinander entwickelt, bereitgestellt und skaliert werden können.
Monitoring	Die kontinuierliche Überwachung und Überprüfung von Systemen, Netzwerken oder Anwendungen zur Sicherstellung ihrer ordnungsgemäßen Funktion und Leistung.
PUE	Power Usage Effectiveness: Ein Maß zur Bestimmung der Energieeffizienz eines Rechenzentrums, berechnet als Verhältnis der Gesamtenergie, die vom Rechenzentrum verbraucht wird, zur Energie, die direkt für die IT-Ausrüstung genutzt wird.
Sandbox	Eine isolierte Testumgebung, in der Software sicher ausgeführt werden kann, ohne Auswirkungen auf das umgebende System zu haben.

Literaturverzeichnis

Allianz (2023): More emissions than meet the eye: Decarbonizing the ICT sector. Verfügbar unter: https://www.allianz.com/en/economic_research/insights/publications/specials_fm0/decarbonizing-information-technologies.html

Balzert, H. (2011): Der Software-Lebenszyklus. In: H. Balzert (Hrsg.), Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb (S. 1–4). Heidelberg: Spektrum Akademischer Verlag. https://doi.org/10.1007/978-3-8274-2246-0_1

Beck, M. T., Feld, S., Linnhoff-Popien, C. & Pützschler, U. (2016): Mobile Edge Computing. In: Informatik-Spektrum, 39(2), 108–114. <https://doi.org/10.1007/s00287-016-0957-6>

Bitkom (2021): Leitfaden Ressourceneffiziente Programmierung. Verfügbar unter: <https://www.bitkom.org/Bitkom/Publikationen/Ressourceneffiziente-Programmierung>

Bordage, F. (greenIT.fr, Hrsg.) (2015): Was ist Software-Obsoleszenz? Verfügbar unter: <https://www.halteobsolescence.org/quest-ce-que-l-obsolescence-logicielle/>

Brundtland, G.H. (1987): Our Common Future: Report of the World Commission on Environment and Development. Geneva, UN-Dokument A/42/427. Verfügbar unter: <http://www.un-documents.net/ocf-ov.htm>

Buchanan, W., Foxon, J., Cooke, D., Iyer, S., Graham, E., DeRusha, B., Binder, C., Chiu, K., Corso, L., Richardson, H., Knight, V., Hussain, A., Allison, A. & Mathews, N. (Microsoft, UBS, WattTime, Green Software Foundation, Hrsg.) (2023): Carbon-aware computing. Measuring and reducing the carbon intensity associated with software in execution. Verfügbar unter: <https://news.microsoft.com/de-ch/2023/01/10/carbon-aware-computing-whitepaper/>

Bundesministerium für Umwelt, Naturschutz, nukleare Sicherheit und Verbraucherschutz (2024). Verfügbar unter: <https://produktinfo.blauer-engel.de/uploads/criteriafile/de/DE-UZ%20215-202001-de%20Kriterien-V4.pdf>

Calero, C., Moraga, M. & Piattini, M. (2021): Introduction to Software Sustainability. In: Introduction to Software Sustainability (S. 1–15). https://doi.org/10.1007/978-3-030-69970-3_1

Community Nachhaltige Digitalisierung (Bundesministerium für Umwelt, Naturschutz, nukleare Sicherheit und Verbraucherschutz (BMUV), Hrsg.) (2024): Green Coding – Booklet zur Workshopreihe. Verfügbar unter: https://www.bmu.de/fileadmin/Daten_BMU/Download_PDF/Digitalisierung/das_green_coding_booklet_bf.pdf

D21 (2022): DIGITAL INDEX 2021/2022: Jährliches Lagebild zur Digitalen Gesellschaft. Verfügbar unter: https://initiated21.de/uploads/03_Studien-Publikationen/D21-Digital-Index/2021-22/d21digitalindex-2021_2022.pdf

Dena (Hrsg.) (2023a): Neue Energiebedarfe digitaler Technologien – Untersuchung von Schlüsseltechnologien für die zukünftige Entwicklung des IKT-bedingten Energiebedarfs

Dena (Hrsg.) (2023b): Blockchains und ihren Stromverbrauch neu denken – Ein Leitfaden für das stromsparende Design dezentraler Dateninfrastrukturen – Kurzfassung

Dena (Hrsg.) (2024): Studie: Energieeffiziente künstliche Intelligenz für eine klimafreundliche Zukunft. Neue Erkenntnisse über Energieeinsparpotenziale bei KI-Anwendungen.

Dick, M., Drangmeister, J., Kern, E. & Naumann, S. (2013): "Green software engineering with agile methods," 2nd International Workshop on Green and Sustainable Software (GREENS), San Francisco, CA, USA, 2013, pp. 78-85, <https://doi.org/10.1109/GREENS.2013.6606425>.

Ericsson (2023): A quick guide to your digital carbon footprint. Ericsson. Verfügbar unter: <https://www.ericsson.com>

Erman, J., Gerber, A., Hajiaghayi, M., Pei, D., Sen, S. & Spatscheck, O. (2011): To Cache or Not to Cache: The 3G Case. In: IEEE Internet Computing, 15(2), 27–34. <https://doi.org/10.1109/MIC.2010.154>

Freitag, C., Berners-Lee, M., Widdicks, K., Knowles, B., Blair, G., Friday, A. (2021): The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations. *Patterns* 2(9):100340. <https://doi.org/10.1016/j.patter.2021.100340>.

Gamma, E., Helm, R., Johnson, R. & Vlissides, J. M. (1994): Design Patterns: Elements of Reusable Object-Oriented Software (1. Aufl.). Addison-Wesley Professional. Verfügbar unter: http://www.amazon.com/Design-Patterns-Elements-Reusable-Object-Oriented/dp/0201633612/ref=ntt_at_ep_dpi_1

Gelenbe, E. (2023): Electricity Consumption by ICT: Facts, trends, and measurements. In: *Ubiquity*, 2023 (August). <https://doi.org/10.1145/3613207>

Goldman Sachs (2024). Verfügbar unter: <https://www.goldmansachs.com/insights/articles/AI-poised-to-drive-160-increase-in-power-demand>

Google (2023): Google Rechenzentren: Effizienz-Datenzentren. Verfügbar unter: <https://www.google.com/intl/de/about/datacenters/efficiency/>

Green Software Foundation (2023a): State of Green Software. Verfügbar unter: <https://stateof.greensoftware.foundation/en/executive-summary/>

Green Software Foundation (2023b): Green Coding is a matter of code quality. Verfügbar unter: <https://greensoftware.foundation/articles/green-coding-is-a-matter-of-code-quality>

Green Software Foundation (2024): Case Study: CAST uses SCI to measure the improvement in CO₂ emissions when decarbonizing software. Verfügbar unter: <https://github.com/Green-Software-Foundation/sci-guide/issues/60>

Green Software Foundation (kAa): Green Software Patterns. Verfügbar unter: <https://patterns.greensoftware.foundation/catalog/cloud/>

Green Software Foundation (kAb): Carbon Awareness. Verfügbar unter: <https://learn.greensoftware.foundation/carbon-awareness>

Green Software Foundation (kAc): Green Software Practitioner – Introduction. Verfügbar unter: <https://learn.greensoftware.foundation/introduction/>

Heldal, R., Nguyen, N.-T., Moreira, A., Lago, P., Duboc, L., Betz, S., Coroamă, V. C., Penzenstadler, B., Porras, J., Capilla, R., Brooks, I., Oyedepi, S., & Venters, C. C. (2024): Sustainability competencies and skills in software engineering: An industry perspective. Verfügbar unter: <https://www.sciencedirect.com/science/article/pii/S0164121224000219>

Hilty, L., Coroama, V., Eicker, M., Ruddy, T. & Thiébaud, E. (2009): The Role of ICT in Energy Consumption and Energy Efficiency. Verfügbar unter: https://www.researchgate.net/publication/267411194_The_Role_of_ICT_in_Energy_Consumption_and_Energy_Efficiency

Hilty, L., Lohmann, W., Behrendt, S., Evers-Wölk, M., Fichter, K. & Hintemann, R. (2015): Grüne Software: Ermittlung und Erschließung von Umweltschutzpotenzialen der Informations- und Kommunikationstechnik (Green IT). Verfügbar unter: https://www.umweltbundesamt.de/sites/default/files/medien/378/publikationen/texte_22_2015_gruene_software.pdf

Hilty, L., Maksimov, Y., Naumann, S., Filler, A., Guldner, A., Kern, E., Gröger, J. & Köhler, A. (2017): Kriterienkatalog für nachhaltige Software. Verfügbar unter: <https://www.umwelt-campus.de/forschung/projekte/green-software-engineering/kriterienkatalog/einleitung>

Hollier, N., Lewis-Toakley, D. & Earle, G. (Thoughtworks, Hrsg.) (2020): Why green cloud optimization is profitable for you and the planet. Verfügbar unter: <https://www.thoughtworks.com/insights/articles/green-cloud>

IBM Cloud Education (2023): Why Green Coding is a Powerful Catalyst for Sustainability Initiatives. Verfügbar unter: <https://www.ibm.com/blog/green-coding/>

Institute for Sustainable IT (2021): Handbook of Sustainable Design of Digital Services. Verfügbar unter: <https://gr491.isit-europe.org/en/>

IONOS Redaktion (IONOS Digital Guide, Hrsg.) (2023): Green Coding im Überblick. Verfügbar unter: <https://www.ionos.de/digitalguide/websites/web-entwicklung/green-coding/#:~:text=Green%20Coding%20ist%20eine%20Programmierpraxis,m%C3%B6glichst%20umweltvertr%C3%A4gliche%20Softwaresysteme%20zu%20erstellen.>

Jones, N. (2018): How to stop data centres from gobbling up the world's electricity. In: Nature, 561, 163–166. <https://doi.org/10.1038/d41586-018-06610-y>

Junger, D., Westing, M., Freitag, C., Guldner, A., Mittelbach, K., Weber, S., et al. (2023): Potentials of Green Coding – Findings and Recommendations for Industry, Education and Science. In: INFORMATIK 2023 – Designing Futures: Zukünfte gestalten (S. 1289–1299). Bonn: Gesellschaft für Informatik e.V. https://doi.org/10.18420/inf2023_137

Kelly, C., Mangina, E. & Ruzelli, A. (2011): Putting a CO₂ figure on a piece of computation. In: 11th International Conference on Electrical Power Quality and Utilisation (S. 1–7)

Koningstein, R. (2021): We now do more computing where there's cleaner energy. Verfügbar unter: <https://blog.google/outreach-initiatives/sustainability/carbon-aware-computing-location/>

Kruchten, P., Obbink, H. & Stafford, J. (2006): The Past, Present, and Future for Software Architecture. In: IEEE Software, 23, 22–30. <https://doi.org/10.1109/MS.2006.59>

Lago, P. (KA): Awesome and Dark Tactics. Catalog. Verfügbar unter: <https://s2group.cs.vu.nl/AwesomeAndDarkTactics/catalog.html>

Lavi, H. (2022): Measuring greenhouse gas emissions in data centres: the environmental impact of cloud computing. Verfügbar unter: <https://www.climatiq.io/blog/measure-greenhouse-gas-emissions-carbon-data-centres-cloud-computing>

Le Goaer, O. (2023): Decarbonizing Software with Free and Open Source Software: The ecoCode Project. In: Research Projects Exhibition Papers Presented at the 35th International Conference on Advanced Information Systems Engineering (CAISE 2023) (Bd. 3413, S. 9–13). Zaragoza, Spain. Verfügbar unter: <https://hal.science/hal-04145890>

Lieder, Y. (adesso Blog, Hrsg.) (2024): Green Coding – Guidelines für die Entwicklung nachhaltiger Software. Verfügbar unter: <https://www.adesso.de/de/news/blog/green-coding-guidelines-fuer-die-entwicklung-nachhaltiger-software.jsp>

Mair, M. (2023): Green QA. Sustainable Software Testing: 5 Strategies for a Greener and Efficient Tomorrow. Verfügbar unter: <https://medium.com/engineered-publicis-sapient/5-steps-to-kick-off-green-qa-in-your-team-80fecc232581>

Malmodin, J., Lövehagen, N., Bergmark, P. & Lundén, D. (2024): ICT sector electricity consumption and greenhouse gas emissions – 2020 outcome. In: Telecommunications Policy, 48(3). <https://doi.org/10.1016/j.telpol.2023.102701>

Manotas, I., Pollock, L. & Clause, J. (2014): SEEDS: A software engineer's energy-optimization decision support framework. In: Proceedings of the 36th International Conference on Software Engineering (ICSE 2014, S. 503–514). New York, NY, USA: Association for Computing Machinery

Michanan, J., Dewri, R. & Rutherford, M. J. (2017): GreenC5: An adaptive, energy-aware collection for green software development. In: Sustainable Computing: Informatics and Systems, 13, 42–60. <https://doi.org/10.1016/j.suscom.2016.11.004>

MoldStud (2024): The Role of Algorithms and Data Structures in Software Development. Verfügbar unter: <https://moldstud.com/articles/p-the-role-of-algorithms-and-data-structures-in-software-development>

Naumann, S., Dick, M., Kern, E. & Johann, T. (2011): The GREENSOFT Model: A reference model for green and sustainable software and its engineering. In: Sustainable Computing: Informatics and Systems, 1(4), 294–304. <https://doi.org/10.1016/j.suscom.2011.06.004>

Penzenstadler, B., Raturi, A., Richardson, D., Calero, C., Femmer, H. & Franch, X. (2014a): Systematic mapping study on software engineering for sustainability (SE4S). In: Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering (EASE '14). New York, NY, USA: Association for Computing Machinery

Penzenstadler, B., Raturi, A., Richardson, D. & Tomlinson, B. (2014b): Safety, Security, Now Sustainability: The Nonfunctional Requirement for the 21st Century. In: IEEE Software, 31(3), 40–47. <https://doi.org/10.1109/MS.2014.22>

Pereira, R., Carção, T., Couto, M., Cunha, J., Fernandes, J. P. & Saraiva, J. (2017a): Helping Programmers Improve the Energy Efficiency of Source Code. In: 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C) (S. 238–240)

Pereira, R., Couto, M., Ribeiro, F., Rua, R., Cunha, J., Fernandes, J. P., et al. (2017b): Energy efficiency across programming languages: how do energy, time, and memory relate? In: Proceedings of the 10th ACM SIGPLAN International Conference on Software Language Engineering (SLE 2017, S. 256–267). New York, NY, USA: Association for Computing Machinery

Poth, A. & Rrjollí, O. (2023): Sustainable IT Products and Services Facilitated by “Whole Team Sustainability” – A Post-mortem Analysis. In: Yilmaz, M., Clarke, P., Riel, A. & Messnarz, R. (Hrsg.): Systems, Software and Services Process Improvement (S. 151–165). Cham: Springer Nature Switzerland

Radersma, R. (2022): Green Coding: Reduce Your Carbon Footprint. ERCIM News, 19–20. Verfügbar unter: <https://ercim-news.ercim.eu/images/stories/EN131/EN131-web.pdf>

Radovanovic, A., Koningstein, R., Schneider, I., Chen, B., Duarte, A., Roy, B., et al. (2022): Carbon-Aware Computing for Data-centers. In: IEEE Transactions on Power Systems, PP, 1. <https://doi.org/10.1109/TPWRS.2022.3173250>

Rajesh, R., Kanakadhurga, D., Prabakaran, N. (2022): Electronic waste: A critical assessment on the unimaginable growing pollutant, legislations and environmental impacts, Environmental Challenges, Volume 7, 100507. <https://doi.org/10.1016/j.envc.2022.100507>

Rattle, R. (2010): Computing Our Way to Paradise? The Role of Internet and Communication Technologies in Sustainable Consumption and Globalization. AltaMira Press

Rozite, V., Bertoli, E. & Reidenbach, B. (International Energy Agency (IEA), Hrsg.) (2023): Data Centres and Data Transmission Networks. Verfügbar unter: <https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks>

Russ, M. (exxeta.com, Hrsg.) (2023): Green Coding: Die Zukunft der Softwareentwicklung ist nachhaltig. Verfügbar unter: <https://exxeta.com/blog/green-coding-die-zukunft-der-softwareentwicklung-ist-nachhaltig>

Sahin, C., Cayci, F., Gutiérrez, I. L. M., Clause, J., Kiamilev, F., Pollock, L., et al. (2012): Initial explorations on design pattern energy usage. In: 2012 First International Workshop on Green and Sustainable Software (GREENS) (S. 55–61)

Surianarayanan, C., Ganapathy, G. & Pethuru, R. (2019): Essentials of microservices architecture: Paradigms, applications, and techniques. Taylor & Francis

Sustainable Web Design (2024): Web Sustainability Guidelines. Verfügbar unter: <https://sustainablewebdesign.org/guidelines/>

Tarara, A. (2022): Wordpress vs. HUGO + Cloudflare. Verfügbar unter: <https://www.green-coding.io/case-studies/wordpress-vs-hugo-cloudflare/>

Umweltbundesamt (2015): Verfügbar unter: https://www.umweltbundesamt.de/sites/default/files/medien/378/publikationen/texte_22_2015_gruene_software.pdf

Vaismoradi, M., Jones, J., Turunen, H. & Snelgrove, S. (2016): Theme development in qualitative content analysis and thematic analysis. Verfügbar unter: <https://nordopen.nord.no/nord-xmlui/bitstream/handle/11250/2386408/Vaismoradi.pdf?sequence>

Venters, C. C., Capilla, R., Betz, S., Penzenstadler, B., Crick, T. & Crouch, S. (2018): Software sustainability: Research and practice from a software architecture viewpoint. In: Journal of Systems and Software, 138, 174–188. <https://doi.org/10.1016/j.jss.2017.12.026>

Verdecchia, R., Lago, P., Ebert, C. & Vries, C. (2021): Green IT and Green Software. In: IEEE Software, 38, 7–15. <https://doi.org/10.1109/MS.2021.3102254>

Weatherbed, J. (2022): This Windows 11 update is trying to save the world. Verfügbar unter: <https://www.techradar.com/news/windows-11-is-getting-an-eco-friendly-update-but-could-microsoft-do-more>

