



OTTO-FRIEDRICH-UNIVERSITÄT
BAMBERG

**Qualitative Analyse von Stakeholdern
in einem Unternehmen zu Sichtweisen
auf Privatsphäre bei
Softwareentwicklern bzw.
-administratoren**

VON

Samet Murat Akcabay

Lehrstuhl für Privatsphäre und Sicherheit in Informationssystemen

2. August 2020

Zusammenfassung

Diese qualitative Analyse beschäftigt sich mit den Sichtweisen auf Privatsphäre und Sicherheit von Stakeholdern in einem Softwareentwicklungs-Unternehmen. Mit der stetig wachsenden Digitalisierung spielt der Datenschutz, und somit die Privatsphäre und Sicherheit, eine immer größere Rolle bei Nutzern des Internets. In Hinblick auf diese Frage wurde sieben Stakeholder in einem Softwareentwicklungsunternehmen befragt und ihre Aussagen in Kategorien, welche einer Präferenz der Einfachheit und Benutzerfreundlichkeit oder der Privatsphäre zugeordnet werden können, unterteilt.

Dabei ist deutlich geworden, dass der überwiegende Anteil der Kategorien dem ersten Aspekt entsprach, was dadurch begründet ist, dass sonst wichtige Faktoren, wie z.B. der Nachvollziehbarkeit, der Möglichkeit der Fehlerbehebung o.Ä. verloren gingen. Zusätzlich können durch diese Aussagen aktuelle Kenntnisse der Forschung, die Gründe hierbei bei der bloßen Existenz von Tools (welche diese Praktiken unterstützen), beim Vorhandensein der Bereitschaft zum Zurückgreifen auf Methoden, welche eher persönliche Daten in Verwendung nehmen und beim allgemeinen Interesse des Unternehmens an der Verarbeitung und Auswertung dieser, zu sehen, unterstützt werden.

Inwieweit sich die Tendenz zwischen den beiden Polen der Datenverarbeitung und -auswertung durch die Entwicklung neuer Technologien und neuen Methoden bzw. Praktiken am Arbeitsplatz in Softwareentwicklungsunternehmen, ändert, bleibt weiterhin abzuwarten.

Inhaltsverzeichnis

KAPITEL 1	Die Relevanz der Privatsphäre im 21. Jahrhundert - eine Einleitung	1
KAPITEL 2	Theoretischer Hintergrund	3
2.1	Die Privatsphäre - eine Eingrenzung	3
2.2	Persönliche Daten am Arbeitsplatz	4
2.3	Stakeholder	5
2.4	DevOps	5
KAPITEL 3	Forschungsmethode	9
3.1	Vorbereitung	9
3.2	Durchführung und Transkription der Experteninterviews	10
3.3	Kodierung der Ergebnisse	11
KAPITEL 4	Ergebnisse	13
4.1	Kategorien mit einstimmigen Ergebnissen	13
4.2	Kategorien mit abweichenden Ergebnissen	18
4.3	Vorgeschlagene Alternativen	19
KAPITEL 5	Diskussion der Ergebnisse	21
5.1	Implikationen für bestehende Theorie	21
5.2	Implikationen für die Praxis	22
5.3	Limitation der Arbeit	23
KAPITEL 6	Fazit und Ausblick	25
APPENDIX A	Verbundene Arbeiten	27
A.1	Stellenausschreibung zur Befragung von Softwareentwicklern bzw. -administratoren	27
A.2	Datenschutzerklärung	27
A.3	Online-Formular zum Ausfüllen über Microsoft Forms	28
APPENDIX B	Interviewleitfaden	31
	Literaturverzeichnis	33
	Eidesstattliche Erklärung	35

1 Die Relevanz der Privatsphäre im 21. Jahrhundert - eine Einleitung

„We believe privacy is a fundamental human right“ [App20] - mit dieser Aussage betritt Craig Federighi, der Vizepräsident der Softwareentwicklung von Apple, einem erfolgreichen Multimilliarden-Unternehmen der Software- und Hardwareentwicklung aus den Vereinigten Staaten, zur Vorstellung der neuen Privatsphäre-Richtlinien des Unternehmens zur eigenen „Worldwide Developer Conference“ die Bühne.

Und dieser Glaube ist nicht unbegründet: Mit der zunehmenden Digitalisierung in den letzten Jahrzehnten werden Menschen, und Internetnutzer im Spezifischen, häufiger vor dieses Dilemma gestellt. Entwicklungen wie dem Internet der Dinge (IoT), Industrie 4.0, „Smart Home“, Wearables (diverse Sensoren an der Kleidung selbst, Smartwatches, Fitnesstracker) uvm. ermöglichen neue (und zum Teil unerforschte) Angriffsmöglichkeiten für Cyberkriminalität [Bun19].

Dies zeichnet sich vor allem durch den Anstieg der Höhe der Fallzahlen und der gleichzeitigen Abnahme der Entdeckungsquote der Malware aus: Im ersten Halbjahr 2018 wurden im Durchschnitt 13.000 Malware-Samples am Tag neu entdeckt [G D18b], während diese Zahl im zweiten Halbjahr 2017 noch bei knapp 23.000 lag [G D18a].

Ähnlich sieht es hier bei dem Anstieg neuer Schadprogramme aus: Der Blog von G Data, einem deutschen Softwareunternehmen, welcher mehrfach für seine Sicherheitslösungen ausgezeichnet wurde [G D20], beschreibt dabei, dass die Zahl der neuen Schadprogrammtypen seit 2007 einen Anstieg um das knapp 63-fache vermerken konnte [G D18a]. Es ist anzunehmen, dass dies sicherlich auch damit zusammenhängt, dass das Internet in den letzten beiden Jahrzehnten einen regelrechten Ausbruch in der Nutzung erlebt hat: Während die Computer- und Internetnutzung 2012, einem Jahr, in welchem im Durchschnitt bereits knapp 46% der Generation über 50 Jahre Internetnutzer waren [GfK16], noch für private Haushalte und Personen ab 10 Jahren bei 78% lag [Sta19], ist diese Internetnutzung 2019 bis auf 88% angestiegen [Sta19].

Nun könnte man anhand der steigenden Tendenz davon ausgehen, dass das Internet sich etabliert hat und Nutzer durch den langjährigen Umgang und dem gleichzeitigen Sammeln von Erfahrung ein allgemeines Wissen über die Risiken und Gefahren im Bezug auf die eigene Sicherheit und Privatsphäre aufgebaut haben. Schließlich haben bei einer Umfrage durch Bitkom Research 2018 89% in der Altersgruppe ab 14 Jahren angegeben, sich in der Regel mit den Privatsphäre-Einstellungen auseinandergesetzt zu haben [Bit18].

Auf der Kehrseite sieht man jedoch, dass beispielsweise diverse Bonusprogramme der DeutschlandCard und Payback GmbH, welche häufig in der Kritik von Verbraucherschützern stehen, da Nutzerdaten zu den Einkäufen und dem Einkaufsverhalten gesammelt werden [HD], immer größer werden und sich verbreiten: Alleine zum Vorjahr konnte die Payback GmbH im Jahre 2018 einen Anstieg des Jahresumsatzes auf 330,49 Millionen Euro in Deutschland und Österreich verzeichnen, während 2012 der Umsatz noch bei 148,57 Millionen Euro lag [Pay19] - ein Anstieg von außergewöhnlichen 122%.

Dieses Phänomen, auf der einen Seite seine Privatsphäre zu wahren und schützen wollen, auf der anderen Seite jedoch seine persönlichen Daten freiwillig auf Sozialen Netzwerken, beim Einkaufen oder durch die eigene Art und Weise, ein internetfähiges Gerät zu bedienen, zu teilen, nennt man das „Privacy Paradox“ (dt. *Privatsphäre-Paradoxon*) [Baro6]. Dieses Phänomen wird im weiteren Verlauf dieser Analyse zur Erläuterung diverser Motive und Sichtweisen auftreten.

2 | Theoretischer Hintergrund

Um in den folgenden Kapiteln ein grundlegendes Verständnis der Begrifflichkeiten, Zusammenhänge und Konzepte gewährleisten zu können, ist es notwendig, diese vorab zu definieren und einzugrenzen. Da sich diese Arbeit mit der Privatsphäre von Stakeholdern in einem Softwareentwicklungsunternehmen beschäftigt, ist es zunächst erforderlich, die Grenzen der Privatsphäre klar zu definieren und anhand dieser, eine Messung zu schaffen. Da diese Messung durch die persönlichen Daten der einzelnen Stakeholder erfolgt, werden Daten im Allgemeinen ebenfalls auf dieses Feld begrenzt und der Umgang mit diesen erklärt. Was Stakeholder sind und welche in dieser Analyse relevant sind, ist ebenfalls ein erforderlicher Schritt zum grundlegenden Verständnis dieser Arbeit. Im Anschluss werden DevOps definiert und mithilfe gewählter Beispiele in der Praxis erläutert.

2.1 Die Privatsphäre - eine Eingrenzung

Der Begriff Privatsphäre ist im Wortschatz der deutschen Sprache tief verankert und findet im Alltag in vielerlei Hinsicht Gebrauch: Menschen decken Notebook- und Smartphonekameras ab, protestieren gegen Überwachungskameras im öffentlichen Raum [Sta13] oder möchten nicht, dass ihre persönlichen und privaten Daten auf sozialen Netzwerken ohne ihre Zustimmung bzw. ohne ihre Kenntnis verbreitet werden [Pic18] - eine grundlegende, allgemeine Kenntnis der Privatsphäre ist also präsent, aber wie lässt sich diese definieren?

Die Privatsphäre lässt sich erklären als einen eingegrenzten, nicht-öffentlichen Raum, in welchem ein Individuum bzw. Individuen nach eigenem Belieben, ohne äußere Einflüsse oder der Beobachtung durch Unbeteiligte, zur freien Entfaltung der eigenen Person, handeln kann bzw. können [Pet20]. Möchte man diese Definition nun auf die Softwareentwicklung übertragen, so müssen die möglichen, vorherrschenden Komponenten von persönlichen Daten näher betrachtet werden: In Unternehmen, welche sich mit der Softwareentwicklung befassen, müssen Daten vorhanden sein, welche Individuen einzigartig bestimmen können - dies kann in Form von IDs und individuell gewählten Nutzernamen vorzufinden sein, aber auf der Kehrseite auch mit Klarnamen, E-Mail Adressen sowie eindeutig identifizierbaren IDs. Schließlich möchte man einzelne, erledigte

Arbeiten innerhalb der Angestellten unterscheiden können.

Nichtsdestotrotz wird es dadurch nicht leichter, Privatsphäre im Allgemeinen greifbar zu machen - diese ist nämlich „vollständig determiniert durch die Bereitschaft zur persönlichen Offenlegung von bestimmten Menschen in einer bestimmten Gemeinschaft und zu einer bestimmten Zeit“ [MWoo]. In einer vereinfachten Ausdrucksweise bedeutet dies, dass jedes Individuum oder jede Gemeinschaft, eigene, selbst deklarierte Grenzen zum Veröffentlichen der eigenen persönlichen Daten besitzt und diese untereinander auch variieren können.

Auf diese Definition und Eingrenzung der Privatsphäre wird in den folgenden Kapiteln dieser Analyse Bezug genommen.

2.2 Persönliche Daten am Arbeitsplatz

In Hinblick auf aktuelle Arbeitsumgebungen in Softwareentwicklungsunternehmen ist bereits seit mehreren Jahren bekannt, dass eine Auswertung der persönlichen Daten (beispielsweise in DevOps-Tools) in den alltäglichen Arbeitsablauf herangewachsen ist. Grund hierfür ist schlichtweg die Natur dieser Arbeit in einem Softwareentwicklungsunternehmen:

Vorgesetzte haben schließlich das Recht sicherzustellen, dass ihre Angestellten ihrer Arbeit bis zu einem bestimmten Grad zufriedenstellend nachgehen [MWoo] - aber auch in Bezug auf die Allgemeinheit am Arbeitsplatz ist dieser Aspekt nicht irrelevant, da Angestellte nicht härter arbeiten möchten als erforderlich, um die Arbeit von inkompetenten Kollegen ausgleichen zu müssen [MWoo]. Die Arbeit in einem Softwareentwicklungsunternehmen kann schließlich nur erschwert nachvollzogen werden, da die bloße Anwesenheit am Arbeitsplatz hierfür meistens nicht ausreicht. Der Angestellte könnte beispielsweise an seinem Arbeitsrechner sitzen, aber seine gesamte Arbeitszeit durch arbeitsirrelevante Unterhaltungen mit Kollegen im internen Kommunikationstool verbringen.

Trotzdem muss das Recht auf Privatsphäre jedes Individuums berücksichtigt bzw. respektiert werden und kann nicht durch die bloße autoritäre Natur eines Vorgesetzter-Angestellten-Verhältnisses relativiert werden. Diese Gründe verdeutlichen das Dilemma, welches sich in Bezug auf die Privatsphäre, und damit verbunden die persönlichen Daten, am Arbeitsplatz herauskristallisiert: Wie weit kann bzw. muss das Unternehmen (also die Führungsebene, aber auch die Kollegen), um einen grundlegenden und annehmbaren Arbeitsfluss zu gewährleisten, und wie weit können bzw. müssen die einzelnen Individuen, welche zu diesem Unternehmen beitragen, von ihren eigenen persönlichen Daten für dieses Vorhaben preisgeben?

Aktuelle Arbeiten zu dem Aspekt der Privatsphäre und der Überwachung persönlicher Daten sind zu dem Entschluss gekommen, dass „Mitarbeiter [insbesondere aus dem DevOps-Bereich] einem besonderen Risiko der datengetriebenen Überwachung ausgesetzt [sind]“ [Fed20], da schlichtweg die Unterstützung von diversen Tools zu diesem Thema heutzutage existiert und eine empirisch nachgewiesene Bereitschaft zur Erfassung und

Auswertung von (persönlichen) Daten gewährleistet ist [Fed20]: Durch einer von Bitkom Research durchgeführten Studie wurde 2015 erwiesen, dass beispielsweise in den Personalabteilungen von jedem zweiten Unternehmen (hier: in der Softwareentwicklung) Bewerber (ohne eigene Kenntnisnahme) über soziale Netzwerke überprüft werden oder „Big Data“-Analysen zunehmend für Kernaufgaben zum Einsatz kommen [Bit15].

2.3 Stakeholder

Der Begriff „Stakeholder“ lässt sich neben der Informatik auch in der Betriebswirtschaftslehre vorfinden - das Bindeglied der beiden Bereiche in diesem Aspekt repräsentieren Unternehmen im Allgemeinen. Da diverse Institutionen, Personen und Personengruppen Erwartungen an Unternehmen haben und eigene Interessen vertreten, stellen Stakeholder jene „[...] [dar], die von den Aktivitäten eines Unternehmens direkt oder indirekt betroffen sind oder [...] ein Interesse an diesen [...] haben“ [Fle16]. Als Beispiele können diese in Form von Kunden und Lieferanten bis zu eigenen Mitarbeitern und Eigentümern (von Unternehmensanteilen oder vom Unternehmen selbst) auftreten [Fle16].

Neben den Stakeholdern selbst existiert der sogenannte „Stakeholder-Ansatz“: Dieser besagt die Betrachtung einer Unternehmung als Organisation unter Zusammenschluss verschiedener Stakeholder [Sey18]. Dabei stellt die Unternehmensleitung die Vermittlung zwischen den verschiedenen Gruppen dar [Sey18].

Betrachtet man diesen Begriff nun aus der informatischen Sicht, so kann man aus der Definition herleiten, dass Stakeholder hier in der Regel Erwartungen und Interessen an einem einem Programm bzw. einer Applikation oder einem Computersystem, statt einem Unternehmen verfolgen. In dieser Analyse ist der Bezug fortlaufend auf diese Stakeholder gelegt, da diese die befragten Experten darstellen.

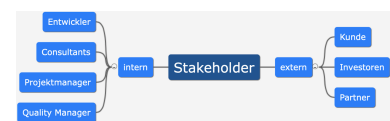


Abbildung 2.1: Beispielhafte Stakeholder-Mindmap.

2.4 DevOps

„Die Zufriedenheit unserer Kunden liegt uns am Herzen“ - durch dieses Motto strahlen Unternehmen ihre Kundenorientierung aus und möchten in der Regel zusätzlich durch Feedback, Kulanz, einem überzeugenden Produkt oder einem allgemein positiv zurückgebliebenem Eindruck beim Kunden überzeugen. In Softwareentwicklungsunternehmen vereint DevOps die benötigten Technologien, die Prozesse und die Menschen miteinander, um für diese Kunden andauernd hochwertige Produkte anbieten zu können [Mic20].

Der Begriff setzt sich zusammen aus „Development“ (dt. *Entwicklung*) von Software sowie „Operations“ (dt. *Vorgänge*), welche damit zusammenhängen [Mic20] und repräsentiert dabei die „Zusammenarbeit von Entwicklung und Betrieb“ [Has15]. Dies erfolgt durch die Einführung von DevOps-Methoden und -Tools (welche nachfolgend erläutert werden),

sodass den zuvor getrennten Bereichen (z.B. der Qualitätssicherung, Sicherheit etc.) die Möglichkeit zur Koordination und Zusammenarbeit geboten wird [Mic20].

Im Folgenden werden beispielhaft repräsentative DevOps-Methoden und -Tools definiert und aufgelistet.



Abbildung 2.2: Die CI-/CD-Pipeline [Red20].

2.4.1 Continuous Integration

In der Softwareprogrammierung ist es mit zunehmender Komplexität und Umfang des Programms üblich, mehrere einzelne Module in dieses einzubinden, weswegen sich die gängige Vorgehensweise in der Softwareentwicklung insofern gestaltet, dass das Gesamtprojekt, welches aus diesen einzelnen Modulen besteht, durch mehrere Teams parallel vorangetrieben wird [PA18].

Statt auf eine täglich erfolgende Kompilierung auf einem extra angelegten „Build“-Server warten zu müssen, kommen heutzutage in modernen Unternehmen die sogenannte „Continuous Integration“ (dt. *Kontinuierliche Integration*, kurz: CI) zum Einsatz [PA18]: Diese bezeichnet „eine Technik der agilen Softwareentwicklung [...], [welche die] in kleinen Schritten vorgenommenen Änderungen [...] regelmäßig zusammenführt“ [Dir18], also eine kontinuierliche Integration von Softwarekomponenten wie z.B. eines Guided User Interfaces (dt. *grafische Benutzeroberfläche*, kurz: GUI), eines Application Programming Interfaces (dt. *Programmierschnittstelle*) etc., in das gemeinsame Projekt bzw. den gemeinsamen Code [PA18].

Dies erfolgt in der Regel unter Einsatz von einem sogenannten „Trunk“, also der „Stamm“, welches das Hauptverzeichnis der Entwicklung des Projekts darstellt und „Branches“, die als „Äste“ bzw. Verzweigungen angesehen werden können und die untergeordneten Verzeichnisse der Hauptentwicklung repräsentieren [Pet15].

In der Arbeit im Team kann der Code dann für dasselbe Projekt also nicht erst wie zuvor nach dem Fertigstellen der anderen Teilbereiche mit in den Gesamtcode des Projekts aufgenommen werden, sondern nach eigenem Bedarf, sogar mehrmals am Tag [Dir18]. Dadurch kann der Code für die beteiligten Programmierer schon viel früher offengelegt werden, sodass Tests und damit verbundene Fehlerbehebungen früher und effizienter, durch Einsparung von Zeit und Ressourcen, durchgeführt werden können.

Trotzdem ist zunächst eine Umorientierung erforderlich ist, da ein bereits gängiger Prozess neu aufgelegt und zusätzliche Prozesse sowie Server neu eingebunden werden müssen [Dir18; PA18]. Dieser gesamte Prozess funktioniert durch das Einspeisen der zusammengeführten Änderungen des Codes in ein Repository, womit im Anschluss eine Versionskontrolle durch die Übernahme von einer aktuellen Version des Codes durch einen automatisierten Build-Prozess [PA18] erfolgt.

2.4.2 Continuous Delivery

Der Begriff „Continuous Delivery“ (dt. *Kontinuierliche Auslieferung*, kurz: CD) führt die CI einen Schritt weiter aus: Eine Sammlung, welche „kurze Entwicklungszyklen und [eine] schnelle Auslieferung von Software-Updates [durch Techniken, Prozesse und Werkzeuge]“ [IA17] ermöglicht, beschreibt man als Continuous Delivery. Anstatt konventionell darauf zu setzen, Software erst zu programmieren und im Anschluss einem Test und einer Evaluation zu unterziehen, bevor diese an Kunden ausgeliefert wird, kann durch Continuous Delivery eine vollfunktionsfähige Version der Software in jedem Entwicklungsstand nach Bedarf vom Kunden bezogen werden - in der Regel ohne Fehler zu enthalten [IA17].

2.4.3 Continuous Deployment

Das Continuous Deployment (kurz: CD) stellt den Abschluss der CI-/CD-Pipeline (s. Abbildung 2.2) dar und erweitert die Continuous Delivery. Hierbei werden die zur Auslieferung fertigen Builds einer Applikation in der Produktionsphase automatisch freigegeben [Red20]. Dies bedeutet in der Praxis, dass nach Fertigstellen der Codeänderungen und nach erfolgreichem Bestehen der definierten Tests, die Applikation automatisiert (bspw. an die Kunden zum Einholen von Feedback) ausgeliefert werden kann [Red20].

2.4.4 Versionsverwaltung

Durch die Zusammenarbeit im Team an einem gemeinsamen Projekt werden Softwareentwickler häufig vor das Problem der Versionsverwaltung gestellt: Wie können mehrere Personen an einem gemeinsamen Projekt und z.T. sogar am selben Programm gleichzeitig arbeiten? Wie können verschiedene Versionen eines Programms einheitlich und übersichtlich verwaltet werden? Und wie verfährt man möglichst effizient mit Code, welcher von unterschiedlichen Autoren geschrieben wurde und fügt diese zusammen?

Zur Lösung dieser Problemstellungen ist die Versionsverwaltung heutzutage in Softwareentwicklungsunternehmen kaum wegzudenken. Diese „ist ein System, welches die Änderungen an einer oder einer Reihe von Dateien über die Zeit hinweg, protokolliert, sodass man später auf eine bestimmte Version zurückgreifen kann“ [CS20].

Laut der Webseite Open Hub, welche zur Katalogisierung von Open-Source-Software verwendet wird und wie hier, den Anteil von Tools zur Versionsverwaltung anhand der registrierten Daten indexiert, stellen Git (71%) und Subversion (24%) die populärsten, welche heutzutage eingesetzt werden, dar (s. Abbildung 2.3) [Syn20], weswegen diese in den folgenden Unterkapiteln kurz angesprochen werden.

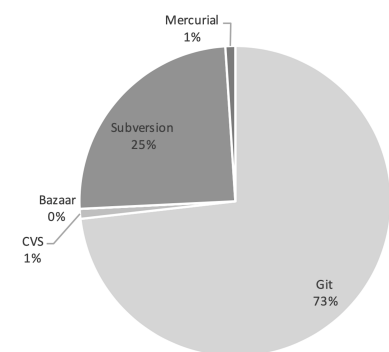


Abbildung 2.3: Versionsverwaltungs-Tools im Ranking (nach Anzahl der Repositories) - Stand: 2020 [Ope20].

2.4.4.1 Git

1: <https://git-scm.com>

Git¹ ist eine freie Software zur Versionsverwaltung [Git20]. Die grundlegende Funktion unterscheidet hierbei zwischen der lokalen, zentralen und verteilten Versionsverwaltung: Die lokale Versionsverwaltung befasst sich dabei mit der „[Verwaltung] aller Änderungen relevanter Dateien in einer Datenbank“ [CS20], wohingegen die zentrale mit dem Problem der Kooperation mit anderen Entwicklern an gemeinsamer Software beschäftigt ist [CS20]. Der letzte Punkt der verteilten Verwaltung behandelt das vollständige Kopieren eines sogenannten „Repositories“ (dt. *Quelle bzw. Lager von Software*) von einem Server anstatt lediglich den letzten Stand zu beziehen [CS20]. Unter Git wird das Hauptverzeichnis als „master“ bezeichnet, wohingegen die Versionierungen hiervon als „Branches“ (vgl. Abbildung 2.4), betitelt werden.

2.4.4.2 Subversion

2: <https://subversion.apache.org>

Apache Subversion² (kurz: SVN) ist ebenfalls eine freie Software zur Versionsverwaltung von Dateien sowie Verzeichnissen, wodurch alte Versionen von Daten und eine Historie der verschiedenen Änderungen und Versionen an diesen betrachtet werden können [CFPo4]. Unter Subversion wird das Verzeichnis der „Hauptlinie“ als „Trunk“ definiert und seine Subverzeichnisse, wie auch schon unter Git, als „Branches“.

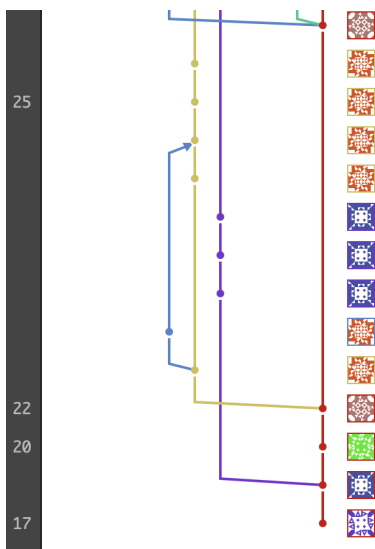


Abbildung 2.4: Beispielhafte Historie einer Versionsverwaltung (hier am Beispiel Git: Verschiedene Profilbilder repräsentieren einzelne Commit-Autoren, Abzweigungen repräsentieren Branches)

3 | Forschungsmethode

Aufgrund der Natur der offenen Befragung wurde hierbei auf Leitfadeninterviews zurückgegriffen. Dadurch wurde gewährleistet, dass das Forschungsthema im Fokus der Befragung bleibt und Thematiken, welche von den Interviewpartnern angesprochen wurden, näher spezifiziert und erläutert werden konnten. Hier wurden insgesamt sieben Interviews mit Personen durchgeführt, welche aus zum Teil verschiedenen Bereichen und Positionen in Softwareentwicklungsunternehmen stammen, aber dennoch mit DevOps und der Softwareentwicklung in Berührung stehen.

Besonders wichtig hierbei war es, soweit möglich, unterschiedliche Experten aus unterschiedlichen Bereichen und Positionen eines Unternehmens zu finden. Dabei waren alle Teilnehmer zum Zeitpunkt der Befragung über 23 Jahre alt. Fünf der sieben Interviews wurden bereits vor Beginn dieser Analyse vom Lehrstuhl für Privatsphäre und Sicherheit in Informationssystemen (kurz: PSI) durchgeführt und transkribiert, weswegen diese weiterverwendet und codiert werden konnten. Zwei der insgesamt sieben Interviews wurden während der Analyse durchgeführt, wovon eine persönlich und durch Audio-Aufzeichnung unterstützt und die andere durch ein zu ausfüllendes Online-Formular stattgefunden hat. Ein persönliches Gespräch war aufgrund der COVID-19 Pandemie nicht möglich. Einen Einfluss auf die Ergebnisse sollten diese Unterschiede in der Demographie und die Art der Durchführung jedoch nicht haben.

Der Leitfaden wurde vorab vom PSI-Lehrstuhl zur Verfügung gestellt und für diese Analyse weiterverwendet. Dieser ist im Anhang einsehbar (s. Anhang B).

3.1 Vorbereitung

Bevor die Experteninterviews durchgeführt wurden, mussten Vorarbeiten geleistet werden. Zunächst war es wichtig, eine Datenschutzerklärung zu erstellen, welche die zu interviewenden Personen über den Umgang mit ihren persönlichen Daten aufgeklärt hat (s. Anhang A.2). Dadurch konnte gewährleistet werden, dass es sich bei dieser Forschung um ein seriöses Vorgehen handelt und die Interessenten deswegen keine Bedenken bezüglich ihrer persönlichen Daten haben. Des weiteren konnte so den Interviewpartnern eine rechtliche Grundlage über ihren Datenschutz und der persönlichen Daten zugesichert werden. Außerdem wurde angegeben, mit der Teilnahme an der Befragung der Datenschutzerklärung

automatisch zuzustimmen.

Parallel dazu wurde eine Ausschreibung erstellt (s. Anhang A.1), welche das Interview beworben hat und in diversen sozialen Netzwerken (Twitter, LinkedIn, XING etc.) geteilt werden konnte. Hier wurde angegeben, Personen im Rahmen dieser Analyse zu suchen, welche bereits in einem Softwareentwicklungsunternehmen tätig waren oder tätig sind, Erfahrungen mit der Arbeit im Team und mit DevOps besitzen und grobe Kenntnisse über interne Abläufe (in Bezug auf Verwaltung des Unternehmens, Abläufe im Team, aber auch mit DevOps) besitzen.

Zu Personen, welche auf die Ausschreibung reagiert haben, wurde über E-Mail Kontakt aufgenommen und ein Termin zum gemeinsamen Treffen vereinbart. Die Befragung wurde in einem Seminarsaal durchgeführt und währenddessen mit einem Notebook aufgezeichnet. Da eine persönliche Befragung nach Ausbruch der COVID-19 Pandemie nicht mehr möglich war, wurde mithilfe von „Microsoft Forms“ ein Online-Formular mit den Leitfragen erstellt und als Direktlink per E-Mail an die Interessenten weitergeleitet (s. Anhang A.3). Hierbei war es besonders wichtig, die Leitfragen so zu wählen, dass ein grundlegendes Verständnis, der Sinn und die Zusammenhänge dieser gewährleistet waren, da eine nachträgliche Erläuterung und Spezifizierung aufgrund der gegebenen Umstände nicht mehr möglich war.

3.2 Durchführung und Transkription der Experteninterviews

Zu Beginn des Interviews wurde der zu Interviewende begrüßt und das Einverständnis zur Aufnahme des Gesprächs eingeholt. Danach wurde die Aufnahme gestartet und die Bestätigung zur Kenntnisnahme der Datenschutzerklärung sowie der Vermerk, diesem jederzeit widersprechen zu können, erläutert. Da in der Stellenausschreibung und im darauffolgenden in Kontakt treten kommuniziert wurde, DevOps und diverse andere Arbeitsabläufe zu kennen, wurde darauf verzichtet, im Vorfeld Erläuterungen und Definitionen zu Begriffen wie DevOps und den Tools, Stakeholder oder Vergleichbares zu klären. Dieses Vorgehen wurde durch den durchweg flüssigen Dialog zusätzlich bestätigt, da keinerlei Unverständnisse aufgetreten sind. Im Interview selbst wurde sich strikt an den Leitfaden gehalten: Sofern eine Abweichung der Antworten und somit eine Richtungsänderung stattgefunden hat, wurde, passend zum vorgegeben Leitfaden, eine „Alternativroute“ der Fragestellungen eingeschlagen.

Zur Abgrenzung der verschiedenen Stakeholder war es zu Anfang wichtig, welche Position im Unternehmen repräsentiert wurde und ob diese bereits oder gegenwärtig mit der Administration von Servern beauftragt waren bzw. sind, da auf diese Art gezielt Fragen in Bezug auf diese Merkmale gestellt werden konnten (s. Tabelle 3.1). In den Befragungen stand im Vordergrund, die Sichtweise auf die Verarbeitung von eigenen und fremden persönlichen Daten einzuholen. Falls verweigert wurde, eigene persönliche Daten (vor allem in DevOps) preiszugeben, wurden

Tabelle 3.1: Position und Erfahrung der Stakeholder im Unternehmen.

Rolle	Beobachtete Ergebnisse	
	Administration von Servern	Nutzung von Log-Dateien
Softwareentwickler	25% Ja, 75% Nein	Ja (Server- und Applikationslogs)
Softwareentwickler und Berater	Nein	Keine Angabe
Lead-Developer	Ja	Ja (Fehler- und Zugangslogs)
Systemadministrator und DevOps	Ja	Ja (Alle Logs)

Alternativen angeregt. Zudem sollten die Experten dazu angehalten werden, ihre eigenen Meinungen und Denkweisen zu präsentieren, wie der Umgang mit persönlichen Daten in anderen Unternehmen bzw. unabhängig von der eigenen Person stattfinden könnte. Bestärkt wurde dieser Aspekt durch die spezifische Frage nach Nutzung von Log-Dateien: Die Intention hierbei war, die Experten indirekt zu fragen, ob diese zunächst Einsicht in persönliche Daten von Kollegen durch Logs haben und ob sie diese Möglichkeit auch tatsächlich aktiv nutzen (möchten).

Die Befragungen fanden im Zeitraum zwischen März und Juli 2020 statt, wobei als spätmöglicher Tag zur Durchführung einer Befragung als der 15. Juli 2020 festgelegt wurde. Diese dauerten im Durchschnitt zehn Minuten. Im Anschluss erfolgte eine Transkription der Interviews. Dieses Prinzip wird auch die „fallbezogene Auswertung“ [DB14] genannt - hierbei wird das qualitative Datenmaterial sequenziell, also von Anfang bis Ende, nach Fällen ausgewertet und im Anschluss kodiert [DB14].

Hierzu wurden die Aufnahmen wiederholt und die Fragestellungen mit den entsprechenden Antworten mittels einer Markdown-Datei festgehalten. Dieses Vorgehen wurde bei der Befragung in Form einer Umfrage redundant, da dieses als PDF-Dokument exportiert und weiterverwendet werden konnte. Zur fristgerechten Fertigstellung der Analyse und der gleichzeitigen Behinderung durch die COVID-19 Pandemie wurde sich nach Absprache mit der Leitung des PSI-Lehrstuhls darauf geeinigt, die Anzahl der Befragungen auf sieben zu limitieren.

3.3 Kodierung der Ergebnisse

Zur Kodierung der Transkriptionen aus den Befragungen wurden die Methoden nach Döring und Bortz gewählt: Den Textstellen (entspricht einem Teil des qualitativen Datenmaterials) aus den Ergebnissen wurden hierbei Codes zugewiesen, welcher eine Zusammenfassung bzw. eine Erklärung dieser darstellt [DB14] (vgl. Tabelle ??) - dieses Vorgehen stellt die Vorgehensweise der deskriptiven Kodierung dar, mit dessen Hilfe ähnliche Aussagen zusammengefasst werden können. Zum Schluss werden diese Codes noch ihren entsprechen, zugehörigen Kategorien, welche im weiteren Verlauf dieser Analyse die jeweiligen Sichtweisen der einzelnen Stakeholder auf die Privatsphäre darstellen, zugeteilt, um eine allgemeine Abgrenzung zwischen diesen gewährleisten zu können.

4 | Ergebnisse

Anhand der Expertenbefragungen konnten insgesamt zwölf verschiedene Kategorien zu den jeweiligen Sichtweisen der einzelnen Stakeholder herausgearbeitet werden. Unterschieden wird hierbei zwischen Kategorien mit einstimmigen und abweichenden Ergebnissen, wobei die genaue Bedeutung dieser Eigenschaften in den entsprechenden Kapiteln näher erläutert werden. Die Alternativen zu den einzelnen Aspekten, welche von den einzelnen Stakeholdern vorgeschlagen wurden, werden darauffolgend aufgelistet. Als Letztes werden die vorgeschlagenen Alternativen in Bezug auf die Existenz, Verarbeitung und Auswertung der persönlichen Daten, welche die Stakeholder im Verlauf der Befragungen genannt haben, vorgestellt.

4.1 Kategorien mit einstimmigen Ergebnissen

In diesem Kapitel werden die einzelnen Kategorien anhand der Einstimmigkeit der einzelnen Kategorien aufgelistet. Einstimmig bedeutet in diesem Fall, dass bei den einzelnen Kategorien eine Abweichung von maximal drei von sieben Befragten (<50%) stattgefunden hat - also anhand der absoluten Häufigkeit eine eindeutige Aussage getroffen werden kann. Hierbei wird zwischen der Präferenz der Privatsphäre im Allgemeinen oder der Benutzerfreundlichkeit bzw. Einfachheit unterschieden: Kategorien, in welchen Experten überwiegend (>50% der Experten sind der selben Ansicht) angegeben haben, die Privatsphäre sei ihnen wichtiger, werden dem Kapitel 4.1.2 zugeordnet, wohingegen die Kategorien, in welchen die Privatsphäre aufgrund der Benutzerfreundlichkeit und Einfachheit vernachlässigt wurde, dem Kapitel 4.1.1 zugerechnet werden.

4.1.1 Kategorien mit einer Präferenz für die Benutzerfreundlichkeit und Einfachheit

Verarbeitete Daten zur Unterstützung der Softwareentwicklung in DevOps-Tools:

Zu Beginn der Interviews haben alle Befragten angegeben, mit persönlichen Daten in Berührung zu kommen und diese auch zu verarbeiten. Dabei haben alle vier Softwareentwickler angegeben, die Nutzeraktivitäten und den Programmcode von Kollegen einsehen und verarbeiten zu können. Diese genannten Daten beinhalten Benutzernamen, Zeitstempel (in Git Commits, Builds und Deployments, z.T.). Erfolgen tue dies zur

Nachvollziehbarkeit der Builds und Deployments, des Programmcodes selbst, z.B. durch Git Commits und zur allgemeinen Fehlerbehebung bei fehlgeschlagenen Builds bzw. fehlerhaftem Programmcode. Zwei Befragte haben zudem angegeben, Zugriff auf sensible Kundendaten in Form von Klarnamen, Anschriften, E-Mail Adressen o.Ä. zu besitzen: Dies ist bei jenen der Fall, welche häufig mit externen Ansprechpartnern und Unternehmen in Kontakt treten und zusammenarbeiten müssen.

Bevorzugter Detailgrad von Zeitstempeln in DevOps-Tools:

Dieser Aspekt nimmt genaueren Bezug auf Zeitstempel in DevOps-Tools und befasst sich mit der Auflösung (z.B. Detailgrad, Lebensdauer und relevante Faktoren) dieser. Hier wurden die Experten befragt, in welcher Auflösung diese konkret den Zeitstempel im Arbeitsalltag benötigen. Bis auf eine Ausnahme, haben alle Befragten angegeben, einen konkreten Zeitstempel mit der Anfangs- und Endzeit (bzw. alternativ mit der Anfangszeit und der Dauer) zu benötigen und auf diesen nicht verzichten zu können. Dies sei vor allem in der Fehlerbehebung (z.B. von fehlerhaften Commits) elementar, denn nur durch eine genaue Zeitangabe könne man den Fehler in Logs, im Code selbst („Wie lange hat es gedauert, bis der Fehler aufgetreten ist?“) oder bei potenziellen Verantwortlichen suchen.

Allgemeine Auswertungen von persönlichen Daten aus DevOps-Tools:

Bei dieser Frage sollten die Befragten angeben, ob diese sich Auswertungsmöglichkeiten von persönlichen Daten aus DevOps-Tools vorstellen könnten, welche sie selbst nicht zwangsläufig durchführen, aber welche bei anderen Kollegen bzw. Unternehmen zum Einsatz kommen könnten. Hier haben alle Stakeholder angegeben, dass unter großer Wahrscheinlichkeit eine Form der Leistungsbeurteilung bzw. -bewertung stattfinden könnte. Da in der Regel Benutzernamen und Zeitstempel in Builds, (Git) Commits und gelegentlich auch in Log-Dateien vorzufinden sind, könne man anhand dieser Daten Rückschlüsse auf die Quantität und Qualität der einzelnen Faktoren ziehen.

Ein Experte hat zudem angegeben, dass in seinem eigenen Unternehmen diese Daten häufig zum Einsatz zur Problembehebung kommen: Dabei wird anhand des Benutzernamens und Zeitstempels die Person hinter einem Build, einem Git Commit oder einer anderen Ausführung ermittelt und durch gemeinsame Absprache mit diesem, eine Behebung des Problems angesteuert.

Ein weiterer hat zusätzlich erläutert, Statistiken (vgl. 4.1) zum Vergleichen dieser Auswertungen mit persönlichen Daten erstellen zu können, da diese Daten unternehmensweit für nahezu jeden Mitarbeiter einsehbar sind.

Durch diese genannten Ansichten der verschiedenen Stakeholder zeigt sich eine unternehmensübergreifende Präferenz zur Benutzerfreundlichkeit bzw. Einfachheit in Bezug auf die Auswertungen der persönlichen Daten in DevOps-Tools.

Persönlichen Daten in Review-Tools:

Diese Frage hat sich damit befasst, ob die Befragten Wert auf die Existenz

Auswertung Tätigkeitsanalyse

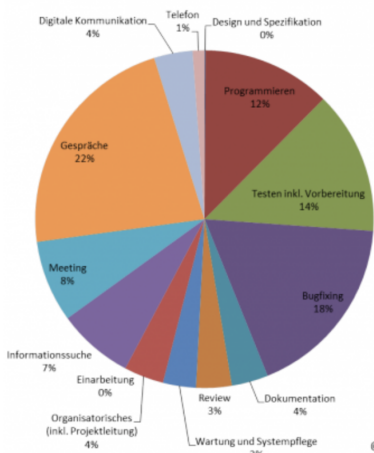


Abbildung 4.1: Beispielhafte Auswertung von Tätigkeiten in einem agilen Softwareprojekt [Voi15]

von persönlichen Daten in Review-Tools (vgl. GitHub oder Gerrit) legen und falls ja, in was für einem Umfang sie diese benötigen bzw. nutzen. Für fünf der insgesamt sieben Befragten war es wichtig, einen Benutzernamen in Review-Tools vorzufinden. Begründet wurde dies durch die allgemeine Möglichkeit der Zuordnung und der Identifikation eines Ansprechpartners im Falle von auftretenden Fragen und Wünschen und zum Einholen von Feedback in Bezug auf den verfassten Code. Drei Befragte, wovon sich bereits drei den Benutzernamen gewünscht haben, möchten zusätzlich Zeitstempel in Review-Tools vorfinden können und erklären diesen Wunsch durch die Möglichkeit der zeitlichen Zuordnung der einzelnen Reviews: „Wann wurde ein Review zu meinem Programmcode bzw. zu meinem Build durchgeführt und wie relevant ist dieses im Augenblick?“. Im Arbeitsalltag lässt sich also aus den Aussagen der Befragten ableiten, dass eine Einschätzung über die Dringlichkeit und Relevanz über die potenziell geforderten Änderungen und Anpassungen im Review seitens der Stakeholder erfolgt.

Die verbliebenen zwei Experten, welche keinen Wert auf persönliche Daten jeglicher Art in Review-Tools legen, begründen ihre Sichtweise dadurch, keine Reviews durchführen zu müssen bzw. keine Reviews zu erhalten oder benötigen in ihrem Arbeitsalltag lediglich Reviews zum Inhalt ihres Codes bzw. zum Fehlschlagen von Builds, ohne jegliche Zeitangaben.

Persönlichen Daten in Administrationstools:

Hier sollten die Befragten angeben, ob in den verwendeten Tools zur Administration persönliche Daten in Form von Zeitstempeln, Benutzernamen o.Ä. existieren. Die Befragten haben hierbei als Administrationstools angegeben, beispielsweise Jenkins als CI-Tool, SpringBoot als Framework oder einfache Kommunikationstools wie Slack und Microsoft-Teams (und insbesondere dessen Kalenderfunktion) einzusetzen.

Im Prinzip sind sich alle Befragten einig, sowohl Benutzernamen, als auch Zeitstempel in Administrationstools zu benötigen und im Arbeitsalltag auch regelmäßig miteinzubeziehen. Um eine allgemeine Funktion von internen Kommunikationstools (zur eindeutigen Identifikation, mit welcher Person kommuniziert wird) zu gewährleisten und bei Builds anhand ihrer Zeitstempel (Startzeit und Dauer) und dem Autor bei Fehlschlag nachvollziehen und einen Ansprechpartner identifizieren zu können, werden hier als Gründe angegeben. Lediglich ein Experte hat angegeben, in SpringBoot auf den Einsatz von persönlichen Daten jeglicher Form verzichten zu können, da diese bisher im eingesetzten SpringBoot ohnehin nicht existent waren.

Detailgrad von Tickets in Ticketverwaltungstools:

In Ticketverwaltungstools (vgl. Jira¹ oder Redmine²) kommen sogenannte „Issues“ zum Einsatz. Damit werden die Tickets bezeichnet, welche z.T. von Kunden, aber auch von Mitarbeitern eines Softwareentwicklungsunternehmens selbst erstellt werden. Mit diesen werden beispielsweise auftretende Fehler in jeglicher Form gemeldet, Verbesserung- bzw. Änderungswünsche angegeben oder eine andere Aufgabenstellung festgehalten.

1: Entwicklungstool zur agilen Softwareentwicklung, vgl. Atlassian <https://www.atlassian.com/de/software/jira>

2: Open-Source-Applikation für Projektmanagement, vgl. Redmine <https://www.redmine.org>

Dadurch ist es möglich, dass Mitarbeiter und Kunden einsehen, nachverfolgen und Kommentare zu den Tickets hinzufügen können, welche Probleme aufgetreten sind, welche Wünsche (z.B. von einer Applikation) gefordert sind oder welche Informationen zu einem bestimmten Thema relevant sind. Da ein grundlegendes Verständnis beim Anlegen eines solchen Issue-Tickets gewährleistet sein muss, haben alle Befragten auch angegeben, eine ausführliche Issue-Beschreibung zu benötigen. In dieser befindet sich i.d.R. der Inhalt des Tickets („Was genau muss getan werden?“, der Status („Wie weit fortgeschritten ist diese Aufgabe?“, ein Zeitstempel („Wann wurde diese Aufgabe erstellt?“ und die Angabe eines Ticket-Erstellers und -Beauftragten („Von wem wurde diese Aufgabe angelegt und wer soll diese bearbeiten?“, wobei diese bei sechs Befragten in Form eines Benutzernamen auftritt und beim verbliebenen als Vor- und Nachname.

Zusätzlich dazu wünschen alle sieben Experten, dass Benutzernamen und Zeitstempel in diesen vermerkt sind. Dies begründet ein Experte dadurch, dass die Lebensdauer eines Tickets stark variieren und z.T. nach mehreren Jahren und zweistelliger Anzahl von „Assignees“ (Derzeitiger Beauftragte für die Bearbeitung des Tickets) weiterhin existieren kann und die Übersicht ohne diese verloren ginge - schließlich könne man so nicht nachverfolgen, welche Person zu welchem Datum bzw. zu welcher Uhrzeit was bearbeitet habe und dies für die Bearbeitung dieser Tickets erforderlich sei.

Interesse an persönlichen Daten in editierten Issues:

Ticketverwaltungstools bieten häufig die Möglichkeit, eigenen, verfassten Text, also in Kommentaren oder in der Beschreibung der Tickets, hinterher zu editieren. Zu diesem Aspekt wurden die Experten befragt, ob ein Interesse, von welcher Person zu welcher Zeit etwas (im Nachgang) editiert wurde, vorhanden sei.

Hier haben sechs Befragte angegeben, Benutzernamen, Zeitstempel und die ursprüngliche Nachricht angezeigt bekommen zu wollen. Der Grund hierfür liegt erneut bei der Nachvollziehbarkeit von verfassten Kommentaren oder der Ticketbeschreibung, da es häufig vorkommt, dass Kommentare aufeinander oder auf die Beschreibung aufbauen und eine nachträgliche Änderung den Kontext dieser erschweren könnte. Vor allem bei „schwierigen Themen“ sei dies häufig der Fall. Dabei reicht es drei der fünf Experten nicht aus, lediglich einen Vermerk auf einen editierten Beitrag zu erhalten - diese wünschen sich zusätzlich, den ursprünglichen Beitrag, die Uhrzeit und den Benutzernamen, vom Verantwortlichen der Änderungen, angezeigt zu bekommen, da auf diese Art und Weise relevante Informationen aus den Änderungen gewonnen werden können („Warum hat diese Person zu diesem Zeitpunkt diesen Beitrag editiert?“). Einer der sechs Experten, welche sich für editierte Beiträge interessieren, hat angegeben, keine Historie einsehen zu können und sich aufgrund dessen nicht für diese zu interessieren, weswegen eine einfache Anzeige, dass ein Beitrag editiert wurde (bspw. in Form eines einfachen Stempels oder Symbols), ausreichen würde.

Tatsächliche Aufbewahrungsdauer von Log-Dateien:

Da ein überwiegender Anteil der Befragten von über 85% (vgl. Tabelle 3.1) angegeben haben, Log-Dateien aktiv zu nutzen, ist es für den weiteren Verlauf der Analyse erforderlich, die tatsächliche Aufbewahrungsdauer der genutzten Log-Dateien im Unternehmen zu spezifizieren. Hier haben alle Befragten angegeben, dass alle Log-Dateien in der Regel für eine unbegrenzte bzw. unbestimmte Dauer aufbewahrt werden. Die Stakeholder haben zudem meist keine Kenntnis über die genaue Aufbewahrungsdauer und beziehen sich bei ihrer Aussage auf die Existenz von alten Log-Dateien (über 5 Jahre).

Die Ausnahme dieser Regel betrifft manuell durchgeführte Bereinigungen von Speicher und eingeleitete Löschungen spezifischer Log-Dateien: Bei einem Befragten kam es vor, dass das System einen Absturz erlitten hat, welche die Log-Dateien unbenutzbar werden ließen, wohingegen bei einem anderen der Serverspeicher geleert werden musste und dieses Vorgehen sämtliche Log-Dateien gelöscht hat.

Aus diesen Aussagen lässt sich ableiten, dass ein Vorgehen zur automatisierten Löschung von Log-Dateien bei den Befragten bisher keine Anwendung gefunden hat.

Mögliche Verarbeitungsinteressen von persönlichen Daten in anderen Unternehmen:

Im Rahmen dieser Frage ist es relevant gewesen, die Ansichten der Experten zu möglichen Verarbeitungsinteressen von persönlichen Daten in anderen Unternehmen einzuholen. Dadurch ist es möglich, ein allgemeines Bild von Softwareentwicklungsunternehmen im Ganzen zu erhalten, falls die Ansichten der Stakeholder sich als deckungsgleich zeigen sollten. In der Tat haben etwa 57% (vier der sieben Befragten) angegeben, dass sie sich vorstellen können, wie Unternehmen die Arbeitszeiten der einzelnen Angestellten nachverfolgen können. Schließlich könne man einsehen, welche Person zu welcher Uhrzeit einen Commit bereitgestellt hat (durch Präsenz des Zeitstempels und Benutzernamens), wer im Kommunikationstool anwesend ist (Online-Status, z.B. in Microsoft Teams) oder wer zu welcher Uhrzeit auf E-Mails reagiert bzw. geantwortet hat, behaupten die Stakeholder. Aus diesen gesammelten Daten eine Arbeitszeiten-Statistik anzulegen, sei nur einen kleinen Schritt von diesem Aspekt fortgeführt. Durch diesen Punkt könne man gebuchte Arbeitszeiten nachprüfen, korrigieren oder ggf. weglassen, da dies durch die Kontrolle redundant werden würde.

Zusätzlich hat ein Befragter angegeben, sich vorstellen zu können, dass Ressourcenmanagement betrieben werden könne: Durch die Einsicht in die Arbeits- bzw. Anwesenheitszeiten der einzelnen Angestellten, hat dieser Befragte behauptet, könne man vorhandene Ressourcen (wie z.B. eingeschaltete Server, Computer oder Strom im Allgemeinen) sparen, indem man diese, abhängig zu den genannten Zeiten, zur Verfügung stellt. Das jeweilige Unternehmen könne so laufende Betriebskosten sparen, was im eigenen Interesse liegen würde.

4.1.2 Kategorien mit einer Präferenz für die Privatsphäre

Persönlichen Daten in Log-Dateien:

Dieser Punkt befasst sich mit dem allgemeinen Interesse an persönlichen Daten, welche in Form von Benutzernamen, Zeitstempeln, IDs etc. in Log-Dateien sämtlicher Art auftreten können. Hier haben die Befragten einstimmig entschieden, kein Interesse an diesen zu haben und ein mögliches Entfernen in der Zukunft willkommen zu heißen. Es wurde lediglich angegeben, dass in bestimmten Fällen (Zusammenarbeit an einem Projekt, in Testfällen von Code, Servern oder Applikationen oder auf Anfrage) eine Nachverfolgung auf Wunsch eines Kunden oder eines Auftrags erfolgen muss.

Für die Befragten ist es nur wichtig, den Ablauf und auftretende Fehler von Servern, Applikationen oder Code im Allgemeinen nachverfolgen zu können - ein Interesse an bestimmten Personen oder Zeitstempeln ist nicht vorhanden, weswegen in dieser Kategorie die Privatsphäre über der Nutzerfreundlichkeit steht.

Bevorzugte Aufbewahrungsdauer von Log-Dateien:

Als Weiterleitung der Frage über die tatsächliche Aufbewahrungsdauer von Log-Dateien in Kapitel 4.1.1 wurden in dieser Kategorie die Experten dazu angehalten, ihre persönliche Präferenz zur Dauer anzugeben. Hier haben alle Befragten angegeben, eine temporäre Aufbewahrungsdauer zu bevorzugen. Begründet wurde dies dadurch, dass Log-Dateien nur für wenige bestimmte Fälle, welche i.d.R. relativ neu sind (<2 Monate), benötigt werden und über diesen Zeitraum hinaus redundant werden.

4.2 Kategorien mit abweichenden Ergebnissen

Dieses Kapitel befasst sich mit der verbliebenen Kategorie, in welcher Aussagen zu Sichtweisen seitens der einzelnen Stakeholder getroffen wurden, die sich zu keiner eindeutigen Aussage zusammenfassen lassen und dementsprechend separat behandelt werden müssen. Als ein abweichendes Ergebnis wird dabei bezeichnet, dass bei sieben Befragten weniger als 28% (entspricht: maximal zwei Befragte besitzen die selbe Ansicht) die selbe Sichtweise zu einer Kategorie haben und somit kein einstimmiges Ergebnis vorliegt.

Sonstige relevante Daten aus der Sicht der Stakeholder:

Zu diesem Aspekt wurden die Experten befragt, ob ihnen sonstige, für sie relevante, Daten im Arbeitsalltag in den Sinn kommen. Diese Frage ist insofern relevant, da hierdurch potenziell neue Aspekte offenbart werden können, welche im Arbeitsalltag eines Stakeholders Anwendung finden. Dementsprechend haben lediglich drei der sieben Befragten eine individuelle Vorstellung wiedergeben können.

Da diese sich selbst aber voneinander unterscheiden und keine gemeinsame Kategorie ausfindig gemacht werden konnte, werden diese im Kapitel mit den Kategorien mit abweichenden Ergebnissen behandelt: Für einen Befragten ist es beispielsweise wichtig, den Umfang eines Commits in

DevOps-Tools einsehen zu können. Das bedeutet, dass eine Unterscheidung, ob es sich bei einem Commit nun um eine „kleine Fehlerbehebung“ oder eine „große Implementierung eines neuen Features“ handelt, durchgeführt werden kann. Dies sei vor allem zur Einschätzung des Arbeitsaufwandes und der Forderungen zur Herangehensweise an die Aufgabe selbst notwendig.

Ein weiterer Experte hat zusätzlich angegeben, die Überprüfung der Anwesenheit der Angestellten vor Ort würde einen wichtigen Bestandteil seines Arbeitsalltages darstellen. Dadurch könne dieser einen Überblick haben, wer gerade in unmittelbarer Nähe anzusprechen ist und seiner Arbeit nachgeht.

Als letzten Aspekt wurde die Aussage getroffen, dass die Commit-Frequenz der eigenen Kollegen für diesen Befragten eine große Rolle spielt: Dadurch könne er, im Falle eines Merge-Requests (Anfrage zum Einbinden eines Commits in das Hauptverzeichnis), einschätzen, wieviel dieser im jeweiligen Commit an Änderungen bzw. Anpassungen zu erwarten hat. Laut eigener Angabe sei es im eigenen Softwareentwicklungsunternehmen üblich, dass unterschiedliche Entwickler unterschiedliche Commit-Frequenzen aufweisen. Manche stellen deutlich größere, aber seltener Commits bereit, als manche andere, welche viele, kleine Commits als angemessen empfinden. Anhand dieser Information könne der Befragte den Commit wohl angemessener beurteilen.

4.3 Vorgeschlagene Alternativen

Temporäre Zeitstempel in DevOps-Tools:

Zusätzlich zur gewünschten Auflösung von Zeitstempeln in DevOps-Tools haben vier der sieben Befragten angegeben, eine temporäre Lebensdauer der Zeitstempel zu bevorzugen, da diese nach einer bestimmten Zeit an Relevanz verlieren würden. Abhängig sei dies beispielsweise durch die Anzahl der zuletzt erfolgreich durchgeführten Builds, einem bestimmten Zeitraum x oder nach vollendeter Lebensdauer eines Tickets. Die Befragten haben dabei Bezug zum eigenen Arbeitsalltag genommen, in welchem die Zeitstempel ohne festgelegte Dauer bestehen bleiben würden und nur nach einer manuell durchgeführten Löschung des übergeordneten Speicherpunktes (z.B. eines abgelaufenen Builds, veralteten Commits etc.) verschwinden würden.

Temporäre Aufbewahrungsdauer von Log-Dateien in Abhängigkeit von Faktor xy :

Zu der Frage, welche Länge zur Aufbewahrung von Log-Dateien seitens der Stakeholder bevorzugt werde, haben sechs von sieben Experten angegeben, eine temporäre Aufbewahrung von Log-Dateien willkommen zu heißen. Diese könne anhand der Lebensdauer von Projekten, der Lebensdauer von Programmcode, den letzten x Builds, einer festgelegten Zeit (z.B. zwei Wochen/Monaten) oder der Existenz von gemeldeten Fehlern deklariert werden. In der Regel sind sich die Befragten einig, eine Dauer von maximal wenigen Monaten zu bevorzugen.

Diese Sichtweisen stellen einen Gegensatz zur tatsächlichen Aufbewahrungsdauer von Log-Dateien in Softwareentwicklungsunternehmen dar, welche in Kapitel 4.1.1 angesprochen werden. Begründet wird dieser Vorschlag durch die kaum vorhandene Nutzung von alten (z.B. älter als zwei Monate) Log-Dateien im Arbeitsalltag der Stakeholder.

5 | Diskussion der Ergebnisse

Der Hintergrund dieser Arbeit lässt sich damit begründen, dass die Relevanz der Privatsphäre im 21. Jahrhundert aufgezeigt und durch gewählte Kategorisierungen der Aussagen aus den Experteninterviews untermauert wird.

Die aufgezeigten Ergebnisse und die folgenden Diskussionen in dieser Analyse dienen dem Zweck, den aktuellen Kenntnisstand zu unterstützen und zukünftige, potenzielle Analysen zu diesem Thema zu motivieren. Hierzu werden die aufgearbeiteten Ergebnisse mit bestehender Literatur korreliert und für den Einsatz in der Praxis ausgewertet. Im Anschluss werden die Einschränkungen, welcher diese Arbeit unterlag, erläutert.

5.1 Implikationen für bestehende Theorie

Anhand der überwiegenden Anzahl der entsprechenden Kategorien aus Kapitel 4, welche der Benutzerfreundlichkeit und damit dem Vernachlässigen der Privatsphäre zugeordnet werden können, kann die Schlussfolgerung mithilfe der Aussagen der Stakeholder gefällt werden, dass aus Sicht der Stakeholder die Benutzerfreundlichkeit bzw. Einfachheit im Arbeitsalltag für den Großteil dieser wichtiger als die Privatsphäre ist. Begründet wird dies dadurch, dass sich diverse Art und Weisen seiner eigenen Arbeit nachzugehen, entweder etabliert haben, eine bestimmte Ungewissheit seitens der Stakeholder vorherrscht oder eine grundlegende Funktion durch die Anonymisierung von persönlichen Daten nicht mehr gewährleistet wäre (vgl. Sinnbeeinträchtigung durch editierte Kommentare in der Kommentarsektion von Tickets, Abschnitt 3).

Diese Behauptung wird durch das zu Anfang erwähnte Phänomen des „Privacy Paradox“ unterstützt: Auf der einen Seite wird die Vorstellung angegeben, dass beispielsweise eine Auswertung der eigenen Arbeitsqualität durch persönliche Daten (z.B. durch Benutzernamen und Zeitstempel in Programmcodes, Builds etc.) in anderen Softwareentwicklungsunternehmen stattfinden könnte und eine allgemeine Ablehnung gegen solche Methoden vorherrscht. Sichtbar ist dies vor allem an der Aussage eines Befragten, welcher behauptet, dass „man [...] von den Commitzeiten [nicht] darauf schließen [kann], wann jemand gearbeitet hat. Manche Menschen leiten das vielleicht so ab, aber das ist komplett falsch“¹. Verstärkt wird diese Aussage durch die Tatsache, dass vor allem Verbraucher

1: Aussage eines Befragten dieser qualitativen Analyse

in Deutschland im Vergleich zu anderen Ländern überdurchschnittlich auf ihre eigene Privatsphäre achten [Ins15] und mit einer steigenden Tendenz Bedenken in Bezug auf diese äußern [BZG16].

Auf der anderen Seite möchte man aber seiner eigenen Arbeit so gut wie möglich nachgehen. Dies wird insofern deutlich, wenn die Befragten beispielsweise angeben, bei einem Fehler die Möglichkeit zu haben, möglichst viele Details (und persönliche Daten) abrufen zu können (vgl. Bevorzugter Detailgrad von Zeitstempeln in DevOps Tools aus Kapitel 4).

Grundlegend kann also der Entschluss gefasst werden, dass der bisherige Forschungsstand begründet und damit, die Kenntnisse und Befürchtungen in Bezug auf die Erfassung und Auswertung von persönlichen Daten in Softwareentwicklungsunternehmen berechtigt ist: Wie beispielsweise aus den Kategorien „Verarbeitete Daten zur Unterstützung der Softwareentwicklung in DevOps-Tools“, „Bevorzugter Detailgrad von Zeitstempeln“ und „Allgemeine Auswertungen von persönlichen Daten aus DevOps-Tools“ deutlich wird, besteht aus der Sicht der Stakeholder ein großes Interesse an der Erfassung von persönlichen Daten (hier: in Form von Benutzernamen und Zeitstempel). Zusätzlich wurde der unmögliche Verzicht auf diese in der Kategorie „Interesse an persönlichen Daten in editierten Issues“ angegeben - einem Sektor, welcher keinen Einfluss auf die grundlegende Funktion des eigenen Arbeitsflusses bietet und lediglich zur Einfachheit und Nachvollziehbarkeit von Inhalten Dritter existiert.

Aus der letzten Kategorie „Mögliche Verarbeitungsinteressen von persönlichen Daten in anderen Unternehmen“ wird eine Skepsis und z.T. eine pessimistische Einstellung gegenüber anderen Unternehmen geäußert, welche mit ethisch fragwürdigen Methoden (Leistungsbeurteilung, Nachverfolgung, Überprüfung ohne Kenntnis, z.B. während einer Bewerbung) agieren sollen. Diese genannten Punkte lassen sich nämlich den zuvor aus Kapitel 2.2 genannten Punkten (Existenz von Tools, welche die Erfassung und Auswertung von persönlichen Daten unterstützen; Bereitschaft vorhanden; im Interesse des Unternehmens) zuordnen.

5.2 Implikationen für die Praxis

Auf Grundlage der bisher durchgeführten Analyse des Forschungsstandes und der Expertenbefragungen, können im nächsten Schritt Schlussfolgerungen in Hinblick auf die Praxis für die Zukunft getroffen werden. Auf der einen Seite wird aus den herausgearbeiteten Kategorien zu den jeweiligen Sichtweisen der Stakeholder deutlich, dass die Einfachheit und Benutzerfreundlichkeit im Arbeitsalltag stärker ins Gewicht fällt als die Privatsphäre. Dadurch lässt sich auf eine allgemeine Zufriedenheit und Akzeptanz mit dem aktuellen Umgang der eigenen, aber auch fremden (von Kollegen, Vorgesetzten, Externen) persönlichen Daten schließen. Zu lediglich zwei Kategorien (Detailgrad des Zeitstempels in DevOps-Tools und der tatsächlichen Aufbewahrungsdauer von Log-Dateien) Vorschläge zu alternativen Methoden erläutert. Trotz zunehmender Relevanz der Privatsphäre im Allgemeinen, scheint ein Großteil, weiterhin den Fokus

auf die Einfachheit und Benutzerfreundlichkeit zu legen - zumindest am Arbeitsplatz.

Diese vorgeschlagenen Alternativen können aber erste Schritte darstellen, in welchen in DevOps-Tools persönliche Daten anonymisiert bzw. komplett entfernt werden können. Weiterhin kann aber auch das Miteinbeziehen der jeweiligen Sichtweisen der Stakeholder in einzelnen Softwareentwicklungsunternehmen ein grundlegendes Konzept zum Umgang mit persönlichen Daten darstellen: Wie aus den Ergebnissen dieser Analyse und dem aktuellen Forschungsstand herausgelesen werden kann, können die einzelnen Ansichtsweisen variieren und eine gute Möglichkeit, den bestmöglichen Umgang mit persönlichen Daten zu gewährleisten, kann durch die Kommunikation untereinander erzielt werden kann.

Dabei können Leitfragen wie „Wie weit sind die einzelnen Stakeholder bereit, ihre eigenen (persönlichen) Daten im Sinne der kooperativen Arbeit mit Kollegen offenzulegen?“ und „Welche Funktionen sind für die einzelnen Stakeholder weniger relevant als die eigene Privatsphäre und können daher durch eine selbstbestimmte Alternative ersetzt werden?“ beim ersten Aufbau einer Arbeitsumgebung oder dem nächsten Team-Meeting Einsatz finden.

5.3 Limitation der Arbeit

Im Laufe der Aufarbeitung dieser qualitativen Analyse unterlag die Arbeit diversen Limitationen. Aufgrund der subjektiven Natur der Sichtweisen der einzelnen Stakeholder und der Anzahl von insgesamt sieben Befragten stellt diese Analyse eine Unterstützung zur bestehenden Literatur dar und kann durch mehrmalige oder veränderte Durchführungen von Expertenbefragungen unterschiedliche Ergebnisse erzielen. Zudem befanden sich im gewählten Zeitraum zur Durchführung der einzelnen Interviews potenzielle Interessenten in der Situation, sich aufgrund der COVID-19-Pandemie im Home-Office zurechtfinden zu müssen und in der Regel keine Zeit bzw. Motivation für externe Anfragen (wie z.B. für Interviews) aufbringen zu können. Durch einen längeren und geeigneteren Zeitraum zur Durchführung und der Steigerung der Anzahl der Befragungen können genauere und repräsentativere Ergebnisse in der Zukunft erzielt werden.

6 | Fazit und Ausblick

Im Umfang dieser Analyse wurde die Privatsphäre in Verbindung mit Stakeholdern und DevOps in Softwareentwicklungsunternehmen präsentiert und anhand Belege aktueller Forschungen ein erster Versuch einer praktischen Feldanalyse gewagt. Anhand der Expertenbefragungen konnten so einzelne Kategorisierungen herausgearbeitet werden, welche entweder einer Präferenz zur Einfachheit und Benutzerfreundlichkeit oder der Privatsphäre im Arbeitsalltag und -ablauf zugeordnet wurden. Die einzelnen Kategorien deckten sich dabei mit den Annahmen aus der Forschung.

Da eine so geringe Anzahl an gewählten Experten für eine Analyse jedoch nicht als vollständig repräsentativ angesehen werden kann, ist diese Arbeit als eine Unterstützung der aktuellen Forschung in Bezug auf die Privatsphäre der Stakeholder und deren Sichtweisen am Arbeitsplatz in Softwareentwicklungsunternehmen zu betrachten.

Nun bleibt zu hoffen, dass zusätzlich zu den aktuellen Entwicklungen, welche mit der zunehmenden Digitalisierung Eintritt in den Alltag aller finden, neue Entdeckungen auch in Hinsicht auf Privatsphäre in der Zukunft kritisch begutachtet werden. Hilfreich ist es allemal, wenn „Big Player“ der Software- bzw. Hardwareentwicklung ein Vorbild repräsentieren und die Relevanz der Privatsphäre immer wieder betonen.

Vielleicht überlegt es sich manch einer auf diese Weise zweimal, eine Kundenkarte wie Payback, der DeutschlandCard o.Ä. zu beantragen und beim Einkaufen zu verwenden.

A | Verbundene Arbeiten

A.1 Stellenausschreibung zur Befragung von Softwareentwicklern bzw. -administratoren

Im Rahmen einer Abschlussarbeit, welche sich mit der qualitativen Analyse von Stakeholdern in einem IT-Unternehmen zu Sichtweisen auf Privatsphäre befasst, sind wir auf der Suche nach potenziellen Befragungen von Softwareentwicklern/-administratoren. Sie sollten bereits

- ▶ in einem Softwareentwicklungsunternehmen gearbeitet haben
- ▶ Erfahrungen mit der Arbeit im Team haben
- ▶ mit einem Versionsverwaltungs-Tool (z.B. Git, SVN etc.) gearbeitet haben
- ▶ grobe Kenntnisse über interne Abläufe besitzen

Melden Sie sich bei Interesse bitte bei mir, **Samet Akcabay**, unter der E-Mail-Adresse **samet-murat.akcabay@stud.uni-bamberg.de**!

A.2 Datenschutzerklärung

Datenschutzerklärung zur Befragung im Rahmen der Abschlussarbeit „Qualitative Analyse von Stakeholdern in einem Softwareentwicklungsunternehmen zu Sichtweisen auf Privatsphäre bei Softwareentwicklern bzw. -administratoren“

Wir, der Lehrstuhl für Privatsphäre und Sicherheit in Informationssystemen, legen großen Wert auf Datenschutz und das Recht auf informationelle Selbstbestimmung. Aufgrund dessen möchten wir Sie mit dieser Datenschutzerklärung über die Erhebung, Speicherung und Verarbeitung der Daten, die im Zusammenhang mit der Abschlussarbeit „Qualitative Analyse von Stakeholdern in einem Softwareentwicklungsunternehmen zu Sichtweisen auf Privatsphäre bei Softwareentwicklern bzw. -administratoren“ erhoben werden, informieren.

Alle personenbezogenen Daten werden in keinsten Weise betrachtet, verarbeitet oder an eine andere Instanz weitergeleitet. Bei der Befragung geht es lediglich darum, die Interessen in verschiedenen Unternehmen

sowie von verschiedenen Interessensgruppen zu analysieren. Jede einzelne Angabe in jeder Befragung wird im Anschluss zu allgemeinen Punkten kategorisiert und zu Mustern zusammengefasst. Nach erfolgreicher Verarbeitung der nicht-personenbezogenen Daten werden sämtliche Aufnahmen und Transkriptionen unwiderruflich gelöscht.

Die befragten Personen willigen bei Teilnahme an der Befragung im Rahmen der Abschlussarbeit dieser Datenschutzerklärung zu. Sie haben jederzeit das Recht, Ihre Angaben zu widerrufen. In diesem Fall werden alle von Ihnen gespeicherten Daten unwiderruflich gelöscht und eine Verarbeitung dieser findet nicht statt.

Sie können jederzeit den Lehrstuhl für Privatsphäre und Sicherheit in Informationssystemen oder den Verfasser der Abschlussarbeit bei Fragen oder Anregungen kontaktieren:

Universität Bamberg
Lehrstuhl für Privatsphäre und Sicherheit in Informationssystemen
An der Weberei 5
96047 Bamberg

Raum: WE5/05.064
Sekretariat: WE5/05.063
Telefon: +49 (0)951-863 2661

Samet Akcabay
An der Spinnerei 11 Zimmer-Nr. D29
96047 Bamberg

Telefon: +49 (0)172-315 5296
E-Mail: samet-murat.akcabay@stud.uni-bamberg.de

A.3 Online-Formular zum Ausfüllen über Microsoft Forms

1. Welche Rolle übernehmen Sie im Unternehmen?
2. Administrieren Sie auch Server?
3. Welche Daten verarbeiten Sie allgemein bei der Softwareentwicklung?
Welche Daten schauen Sie sich explizit mithilfe von DevOps-Tools an, die Sie bei der Softwareentwicklung unterstützen?
4. Wo liegen diese Daten, die von Ihnen verarbeitet werden? Finden sich unter diesen auch personenbezogene Daten?
5. Was bedeutet für Sie allgemein "Daten verarbeiten"? Welche Abläufe bringen Sie damit in Verbindung?
6. Wie genau können Sie diese von Ihnen genannten Daten einsehen?
7. Schauen Sie sich bestimmte Daten explizit an bzw. legen Sie Wert darauf, dass bestimmte Daten gegeben sind?

8. Schauen Sie sich auch explizit an, wer an welchen Issues/Builds (falls vorhanden) in welchem Projekt gearbeitet hat?
9. Reviewen Sie auch Code von anderen Entwicklern? Falls ja, was genau schauen Sie sich beim verwendeten Review-Tool an (z.B. Name des Reviewers, Zeitstempel, ID etc.)? Worauf legen Sie besonders wert?
10. Allgemein gefragt: Was wäre für Sie eine angemessene Zeit zur Aufbewahrung von personenbezogenen Daten? Bitte begründen Sie Ihre Antwort.
11. Wie lange werden die von Ihnen genannten Daten bei Ihnen im Unternehmen tatsächlich aufbewahrt?
12. Zurück zu Issues/Builds: Schauen Sie sich auch an, wann/ob/von wem der Issue erstellt/bearbeitet/kommentiert wurde?
13. Welche Interessen verfolgen Sie generell, wenn Sie sich Zeitstempel in DevOps-Tools ansehen? Worauf legen Sie besonders wert?
14. Von der anderen Seite aus betrachtet: Worauf könnten Sie verzichten?
15. Was glauben Sie, zu welchem Zweck man diese Tools verwendet? (z.B. Nachvollziehbarkeit, Leistungsbeurteilung, Qualitätssicherung etc.)
16. Könnten Sie sich andere Auswertungen aus den Daten vorstellen? (Auch, was Sie persönlich noch nicht selbst erlebt/gemacht haben, sondern welche im Interesse des Unternehmens liegen)
17. Verwenden/arbeiten Sie mit Log-Dateien?
18. Falls ja: Finden Sie es wichtig, dass Benutzername und Zeitstempel in den Log-Dateien vorhanden sind? Falls nicht, worauf würden Sie stattdessen setzen?
19. Wie lange werden Log-Dateien bei Ihnen aufbewahrt? (z.B. von Tickets, Builds, Issues etc.)
20. Fällt Ihnen noch etwas bisher nicht genanntes ein, worauf Sie wert legen und was Sie sich ansehen?
21. Fallen Ihnen andere Verarbeitungsinteressen bzw. Auswertungsansätze ein, die andere Firmen haben könnten?

B | Interviewleitfaden

Literaturverzeichnis

- [PA18] H (Pseudonym) und S Augsten. *Was ist Continuous Integration?* 2018. <https://www.dev-insider.de/was-ist-continuous-integration-a-690914/> (besucht am 16. 03. 2018) (siehe S. 6).
- [App20] Apple. *WWDC 2020 Special Event Keynote — Apple*. 2020. <https://www.youtube.com/watch?v=GEZhD3J89ZE> (besucht am 22. 06. 2020) (siehe S. 1).
- [BZG16] G Bansal, FM Zahedi und D Gefen. Do context and personality matter? Trust and privacy concerns in disclosing private information online. In: *Information and Management* 53 1:(2016), 1–21 (siehe S. 22).
- [Baro6] SB Barnes. A privacy paradox: Social networking in the United States. In: *First Monday* 11(9):(2006) (siehe S. 2).
- [Bit18] Bitkom. Social-Media-Trends. In: *Bitkom Research*:(Feb. 2018), 12. <https://www.bitkom.org/Presse/Anhaenge-an-PIs/2018/180227-Bitkom-PK-Charts-Social-Media-Trends.pdf> (siehe S. 2).
- [Bit15] Bitkom Research. 2015. <https://www.bitkom-research.de/de/presse?page=73> (siehe S. 5).
- [Bun19] Bundeskriminalamt. *Cybercrime. Bundeslagebild 2018*. Bundeskriminalamt, 2019 (siehe S. 1).
- [CS20] S Chacon und B Straub. Pro Git. In:(2020), 9–11 (siehe S. 7, 8).
- [CFPo4] B Collins-Sussman, BW Fitzpatrick und CM Pilato. *Version Control with Subversion*. 2004 (siehe S. 8).
- [Dir18] J Dirk. *Was ist Continuous Integration (Kontinuierliche Integration)?* 2018. <https://www.it-talents.de/blog/it-talents/was-ist-continuous-delivery> (besucht am 26. 10. 2018) (siehe S. 6).
- [DB14] N Döring und J Bortz. *Forschungsmethoden und Evaluation*. Springer, 2014 (siehe S. 11).
- [Fed20] H Federrath. EMPRI-DEVOPS. Employee Privacy in Software Development and Operations. In:(2020) (siehe S. 4, 5).
- [Fle16] J Fleig. *Stakeholder erkennen und analysieren. Was sind Stakeholder und was bedeutet der Stakeholder-Ansatz?* 2016. <https://www.business-wissen.de/hb/was-sind-stakeholder-und-was-bedeutet-der-stakeholder-ansatz/> (besucht am 15. 03. 2016) (siehe S. 5).
- [G D18a] G Data. *Malware-Zahlen 2017*. 2018. <https://www.gdata.de/blog/2018/03/30607-malware-zahlen-2017> (besucht am 26. 03. 2018) (siehe S. 1).
- [G D18b] G Data. *Malwarezahlen erstes Halbjahr 2018: Die Gefahr lauert im Web*. 2018. <https://www.gdata.de/blog/2018/08/31027-malwarezahlen-erste-halbjahr-2018-die-gefahr-lauert-im-web> (besucht am 30. 08. 2018) (siehe S. 1).
- [G D20] G Data. *Auszeichnungen & Prüfsiegel*. 2020. <https://www.gdata.de/ueber-g-data/auszeichnungen> (siehe S. 1).
- [GfK16] GfK. *Consumer Study 2012, 2014, 2016*. 2016. <https://www.nim.org/compact/fokusthemen/nicht-ohne-mein-handy-auch-jenseits-der-50/> (besucht am 06/2016) (siehe S. 1).
- [Git20] Git. *About*. 2020. <https://git-scm.com> (siehe S. 8).
- [Has15] W Hasselbring. DevOps: Softwarearchitektur an der Schnittstelle zwischen Entwicklung und Betrieb. In:(2015), 1–20 (siehe S. 5).
- [HD] N Hatke und DPA. *Kundenkarten lohnen sich nicht immer*. <https://www.augsburger-allgemeine.de/themenwelten/wirtschaft/Kundenkarten-lohnen-sich-nicht-immer-id7725336.html> (siehe S. 2).
- [IA17] Ilan_r_r und S Augsten. *Definition "Kontinuierliche Auslieferung". Was ist Continuous Delivery?* 2017. <https://www.dev-insider.de/was-ist-continuous-delivery-a-664429/> (besucht am 08. 12. 2017) (siehe S. 7).

- [Ins15] Institut für Handelsforschung. Perspektiven für den Datenschutz in Europa aus der Sicht der Verbraucher und des (elektronischen) Handels. In: (2015) (siehe S. 22).
- [Mic20] Microsoft Azure. *Was ist DevOps?* 2020. <https://azure.microsoft.com/de-de/overview/what-is-devops/#culture> (siehe S. 5, 6).
- [MWoo] S Miller und J Weckert. Privacy, the Workplace and the Internet. In: *Journal of Business Ethics* 28:(2000), 255–275 (siehe S. 4).
- [Ope20] Open Hub. 2020. <https://www.openhub.net/repositories/compare> (besucht am 24. 07. 2020) (siehe S. 7).
- [Pay19] Payback. *Umsatz der Payback GmbH in den Jahren von 2012 bis 2018*. 2019. <https://de.statista.com/statistik/daten/studie/1088022/umfrage/umsatz-der-payback-gmbh/> (besucht am 09/2020) (siehe S. 2).
- [Pet15] M Peters. *SVN, was sind Trunk, Branch und Co?* 2015. <https://www.kussin.de/development/svn-was-sind-trunk-branch-und-co/> (besucht am 30. 09. 2015) (siehe S. 6).
- [Pet20] B Pettinger. *Recht auf Privatsphäre – ein kleiner Hoffnungsschimmer bleibt*. 2020. <https://www.dr-datenschutz.de/recht-auf-privatsphaere-ein-kleiner-hoffnungsschimmer-bleibt/> (besucht am 08. 04. 2020) (siehe S. 3).
- [Pic18] A Picchi. *#LogOutFacebook: The social media giant faces protests and canceled accounts*. 2018. <https://www.cbsnews.com/news/logoutfacebook-the-social-media-giant-faces-naacp-protests-and-canceled-accounts/> (besucht am 18. 12. 2018) (siehe S. 3).
- [Red20] RedHat. *Was versteht man unter CI/CD?* 2020. <https://www.redhat.com/de/topics/devops/what-is-ci-cd> (siehe S. 6, 7).
- [Sey18] T Seyfriedt. *Stakeholder-Ansatz*. 2018. <https://wirtschaftslexikon.gabler.de/definition/stakeholder-ansatz-46282/version-269567> (besucht am 15. 02. 2018) (siehe S. 5).
- [Sta13] O Stallwood. Game to destroy CCTV cameras: vandalism or valid protest? In: *The Guardian*:(2013) (siehe S. 3).
- [Sta19] Statistisches Bundesamt. *Computer- und Internetnutzung im ersten Quartal des jeweiligen Jahres von Personen ab 10 Jahren*. 2019. [https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Einkommen-Konsum - Lebensbedingungen/ IT - Nutzung/Tabellen/zeitvergleich-computernutzung-ikt.html](https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Einkommen-Konsum-Lebensbedingungen/IT-Nutzung/Tabellen/zeitvergleich-computernutzung-ikt.html) (besucht am 05. 09. 2019) (siehe S. 1).
- [Syn20] Synopsis Inc. *Compare Repositories*. 2020. <https://www.openhub.net/repositories/compare> (siehe S. 7).
- [Voit15] S Voigt. Agile Softwareprojekte: Informationssuche & Dokumentation. In: *SprintDoc*:(2015) (siehe S. 14).

Eidesstattliche Erklärung

Ich erkläre hiermit gemäß § 17 Abs. 2 APO, dass ich die vorstehende Bachelorarbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Bamberg, den _____

Samet Murat Akcabay