



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. Aus praktischer Sicht sind die Sichtweisen auf diese unterschiedlich, weswegen bei der Recherche darauf Wert gelegt wurde, möglichst viele Experten aus unterschiedlichen Berufsgruppen (z.B. Werkstudenten, Festangestellte, Team-Leiter o.Ä.) einzubeziehen und deren Aussagen auszuwerten. Nach erfolgreicher Auswertung der Expertenbefragungen hat sich ergeben, dass die Berufsgruppe, damit verbunden die Arbeitsweise und der -umfang, [...] keinen Einfluss auf die Einstellung gegenüber personenbezogenen Daten und die Verarbeitung dieser in DevOps-Tools haben.

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	Stakeholder	4
2.3	DevOps	4
KAPITEL 3	Forschungsmethode	7
3.1	Vorbereitung	7
3.2	Durchführung und Transkription der Experteninterviews	8
3.3	Codierung der Ergebnisse	9
KAPITEL 4	Ergebnisse	11
KAPITEL 5	Diskussion der Ergebnisse	13
5.1	Implikationen in der Theorie	13
5.2	Implikationen in der Praxis	13
5.3	Limitation der Arbeit	13
KAPITEL 6	Fazit und Ausblick	15
	Literaturverzeichnis	17
	Eidesstattliche Erklärung	19

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

„We believe privacy is a fundamental human right“ [**Apple:2020aa**] - mit dieser Aussage betritt Craig Federighi, der Vizepräsident der Softwareentwicklung von Apple, einem Multimilliarden-Unternehmen der Software- und Hardwareentwicklung aus den Vereinigten Staaten, zur Vorstellung der neuen Privatsphäre-Richtlinien des Unternehmens zur sogenannten, firmeneigenen „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 [**Bundeskriminalamt:2018aa**]. 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 [**GDB:18**], während diese Zahl im zweiten Halbjahr 2017 noch bei knapp 23.000 lag [**GDB:17**].

Ähnlich sieht es hier bei dem Anstieg neuer Schadprogramme aus: Der G Data Blog, ein deutsches Softwareunternehmen, welches mehrfach für seine Sicherheitslösungen ausgezeichnet wurde [**GD:2020aa**], beschreibt dabei, dass die Zahl der neuen Schadprogrammtypen seit 2007 einen Anstieg um das knapp 63-fache vermerken konnte [**GDB:17**]. 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 Plus50 Internetnutzer waren [**GfK:2016aa**], noch für private Haushalte und Personen ab 10 Jahren bei 78% lag [**Bundesamt:2019aa**], ist die Internetnutzung 2019 bis auf 88% angestiegen [**Bundesamt:2019aa**].

Nun könnte man anhand der steigenden Tendenz davon ausgehen, dass das Internet sich etabliert und die Nutzer durch den langjährigen Umgang

und das gleichzeitige Sammeln von Erfahrung ein allgemeines Wissen über die Risiken und Gefahren im Bezug auf die eigene Sicherheit und Privatsphäre aufgebaut haben. Bei einer Umfrage durch Bitkom Research 2018 haben 89% in der Altersgruppe ab 14 Jahren angegeben, sich in der Regel mit den Privatsphäre-Einstellungen auseinandergesetzt zu haben [Bitkom:2018aa]. Auf der Kehrseite sieht man jedoch, wie 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 [Hatke:aa], immer größer und verbreiteter werden: 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 [Payback:2019aa].

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*) [Barnes:2006aa] und wird im weiteren Verlauf eine Rolle spielen.

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. Des weiteren müssen die Stakeholder, welche in dieser Analyse betrachtet werden, definiert, identifiziert und im weiteren Verlauf in Gruppen gegliedert werden. Im Anschluss werden DevOps definiert und anhand gewählter Beispiele, eine Analyse dieser aufgebaut.

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 [Stallwood:2013aa] oder möchten nicht, dass ihre persönlichen und privaten Daten auf sozialen Netzwerken ohne ihre Zustimmung bzw. ohne ihre Kenntnis verbreitet werden [Picchi:2018aa] - doch was genau bedeutet Privatsphäre?

Die Privatsphäre lässt sich definieren als einen eingegrenzten, nicht-öffentlichen Raum, in welchem ein Individuum bzw. Individuen nach eigenem Belieben, ohne äußere Einflüsse oder die Beobachtung durch Unbeteiligte, zur freien Entfaltung der eigenen Person, handeln kann [Pettinger:2020aa]. Möchte man diese Definition nun auf die Softwareentwicklung widerspiegeln, 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 individuell und bestimmen können (Grund liefern!) - 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.

2.2 Stakeholder

Der Begriff „Stakeholder“ lässt sich neben der Informatik auch in der Betriebswirtschaftslehre vorfinden - das Bindeglied der beiden Bereiche in diesem Aspekt stellen Unternehmen im Allgemeinen selbst dar. 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“ [Fleig:2016aa]. Als Beispiele können diese in Form von Kunden und Lieferanten bis zu eigenen Mitarbeitern und Eigentümern (von Unternehmensanteilen oder vom Unternehmen selbst) auftreten [Fleig:2016aa]. Neben den Stakeholdern selbst existiert der sogenannte „Stakeholder-Ansatz“: Dieser besagt die Betrachtung einer Unternehmung als Organisation unter Zusammenschluss verschiedener Stakeholder [Seyfried:2018aa]. Dabei stellt die Unternehmensleitung die Vermittlung zwischen den verschiedenen Gruppen dar [Seyfried:2018aa]. 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 Programm bzw. einer Applikation oder einem Computersystem, statt einem Unternehmen verfolgen.

In dieser Analyse ist der Bezug fortlaufend auf diese Stakeholder gelegt.

2.3 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 [MSAzure:2020aa]. Der Begriff setzt sich zusammen aus „Development“ (dt. *Entwicklung*) von Software sowie „Operations“ (dt. *Vorgänge*), welche damit zusammenhängen [MSAzure:2020aa] und repräsentiert dabei die „Zusammenarbeit von Entwicklung und Betrieb“ [Hasselbring:2015aa] dar. Dies erfolgt durch die Einführung von DevOps-Methoden und -Tools, sodass den zuvor getrennten Bereichen, und zusätzlichen wie beispielsweise der Qualitätssicherung, der Sicherheit etc., die Möglichkeit zur Koordination und Zusammenarbeit geboten wird [MSAzure:2020aa].

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

2.3.1 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 [HJL:2018aa]. Statt auf eine täglich erfolgende Kompilierung auf einem extra angelegten „Build“-Server warten zu müssen, kommt heutzutage in modernen Unternehmen die sogenannte „Continuous Integration“ (dt. *Kontinuierliche Integration*, kurz: CI) zum Einsatz [HJL:2018aa]: Diese bezeichnet „eine Technik der agilen Softwareentwicklung [...], [welche die] in kleinen Schritten vorgenommenen Änderungen [...] regelmäßig zusammenführt“ [Dirk:2018aa], 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 [HJL:2018aa]. 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 [Peters:2015aa].

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 [Dirk:2018aa]. Dadurch kann der Code für die beteiligten Programmierer schon viel früher offengelegt werden, sodass Tests und damit verbunden, Fehlerbehebungen früher und effizienter durch Einsparung von Zeit und Ressourcen, durchgeführt werden können - wobei trotzdem zunächst eine Umorientierung erforderlich ist, da ein bereits gängiger Prozess neu aufgelegt und zusätzliche Prozesse sowie Server neu eingebunden werden müssen [HJL:2018aa; Dirk:2018aa]. 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 [HJL:2018aa].

2.3.2 Continuous Delivery

Der Begriff „Continuous Delivery“ (dt. *Kontinuierliche Auslieferung*, kurz: CD) geht einen Schritt weiter: Eine Sammlung, welche „kurze Entwicklungszyklen und [eine] schnelle Auslieferung von Software-Updates [durch Techniken, Prozesse und Werkzeuge]“ [Ilanrr:2017aa] 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 [Ilanrr:2017aa].



Abbildung 2.1: Die CI-/CD-Pipeline.

2.3.3 Continuous Deployment

Das Continuous Deployment (kurz: CD) stellt den Abschluss der CI-/CD-Pipeline (s. Abbildung 2.1) dar und erweitert den Continuous Delivery, welche die zur Auslieferung fertigen Builds automatisch freigibt, um die automatische Freigabe einer Applikation in der Phase der Produktion [RedHat:2020aa]. Dies bedeutet in der Praxis betrachtet, dass nach Fertigstellen der Codeänderungen und nach erfolgreichem Bestehen der definierten Tests, die Applikation automatisch (bspw. an die Kunden zum Einholen von Feedback) ausgeliefert werden kann [RedHat:2020aa].

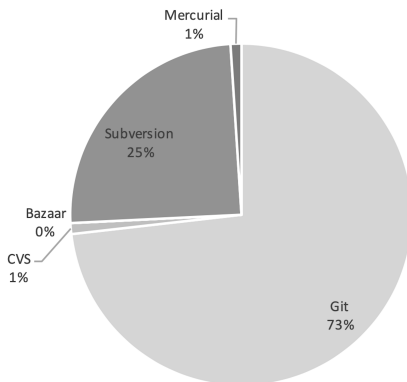


Abbildung 2.2: Versionsverwaltungs-Tools im Ranking (nach Anzahl der Repositories) - Stand: 2020.

2.3.4 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 Konflikte ist die Versionsverwaltung heutzutage in Softwareentwicklungsunternehmen essentiell. 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.“ [Scott-Chacon:2020aa]. Laut der Webseite Open Hub, welche zur Katalogisierung von Open-Source-Software verwendet wird und wie hier, den Anteil von Versionsverwaltungs-Tools anhand der registrierten Daten indexiert, stellen Git (71%) und Subversion (24%) die Spitze dar (s. Abbildung 2.2) [Inc.:2020aa], weswegen diese in den folgenden Unterkapiteln näher betrachtet werden.

2.3.4.1 Git

Git 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“ [Scott-Chacon:2020aa], wohingegen die zentrale mit dem Problem der Kooperation mit anderen Entwicklern an gemeinsamer Software [Scott-Chacon:2020aa]. 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 [Scott-Chacon:2020aa].

2.3.4.2 Subversion

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 hierbei 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 Letzte durch ein zu ausfüllendes Online-Formular stattgefunden hat, da ein persönliches Gespräch aufgrund der COVID-19 Pandemie nicht möglich war. Einen Einfluss auf die Ergebnisse sollten diese Unterschiede in der Demographie und in 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. Anlage [...]).

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. Dadurch konnte zunächst gewährleistet werden, dass es sich bei dieser Forschung um ein seriöses Vorgehen handelt. 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, 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. 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 waren.

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. 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 Alternativen angeregt. Zudem sollten die Experten dazu angeregt werden, ihre eigene Meinung und Denkweise zu präsentieren, wie der Umgang mit persönlichen Daten in anderen Unternehmen bzw. unabhängig von

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)

der eigenen Person stattfindet. Bestärkt wurde dieser Aspekt durch die spezifische Frage nach Nutzung von Log-Dateien: Die Intention hierbei war dabei, 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.

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. 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 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 Myers gewählt.

4 | Ergebnisse

Dies ist das Kapitel, in welchem ich meine Ergebnisse präsentiere und evaluiere.

5 | Diskussion der Ergebnisse

5.1 Implikationen in der Theorie

5.2 Implikationen in der Praxis

5.3 Limitation der Arbeit

6 | **Fazit und Ausblick**

Dies ist das Kapitel, in welchem ich meine Bachelorarbeit abschlieÙe und ein Fazit angebe.

Literaturverzeichnis

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