Sichere Softwareentwicklung von Cloud-Anwendungen

Identifikation und Validierung von Erfolgsfaktoren aus strategischer und operativer Perspektive unter besonderer Berücksichtigung von Systemen, Prozessen und Werkzeugen

Marc Aurel Schubert, M.Sc.

Erstbetreuer:
Prof. Dr. Harald F. O. von Korflesch
Universität Koblenz-Landau

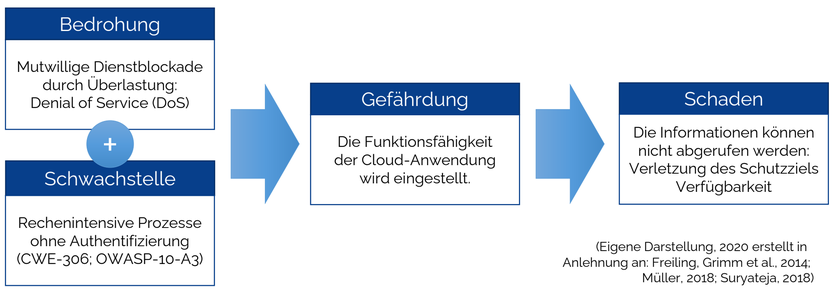
Zweitbetreuer:
Prof. Dr. Sven Pagel
Hochschule Mainz

UNIVERSITÄT KOBLENZ · LANDAU
Institute for Management
WIRTSCHAFT HOCHSCHULE MAINZ UNIVERSITY OF APPLIED SCIENCES
innoprom-security.eu

Motivation und Relevanz

Informationssicherheitsbedenken beinträchtigen den Einsatz von Cloud Computing (Cloud Monitor 21, Bitkom e.V.)

- Cloud Computing ermöglicht es Organisationen bedarfsorientiert Ressourcen (Netze, Server, Speicher, Anwendungen und Dienste) über Hochgeschwindigkeitsnetze mit minimalem Managementaufwand zu beziehen (Bedner, 2013; Mell und Grance, 2011).
- Der Studie Cloud-Monitor 2021 des Bundesverbandes Informationswirtschaft, Telekommunikation und neue Medien (bitkom e.V.) zufolge setzen 82% der Unternehmen in Deutschland (n =556) mit einer Größe ab 20 Mitarbeitern auf Cloud Computing (Heidkamp, 21).
- Informationssicherheitsbedenken beinträchtigen den Einsatz von Cloud Computing (Heidkamp, 21):
  - 81% der Unternehmen speichern unkritische Geschäftsdaten
  - 75% der Unternehmen befürchten unberechtigten Zugriff
  - 60% der Unternehmen befürchten Verlust von Daten
- Von der Bedrohung zum Schaden:



(Eigene Darstellung, 2020 erstellt in Anlehnung an: Freiling, Grimm et al., 2014; Müller, 2018; Suryateja, 2018)

Problemstellung

Problem

Die Informationssicherheitsbedenken im Cloud Computing beeinträchtigten den Einsatz von Cloud Computing in der Verarbeitung von kritischen Geschäftsdaten von Unternehmen in Deutschland.

Beitrag zur Problemlösung

Durch sichere Softwareentwicklung von Cloud-Anwendungen mit weniger Schwachstellen sollen Cloud-Anwendungen mit erhöhtem Sicherheitsniveau hervorgebracht werden und dadurch die Informationssicherheitsbedenken reduziert werden.

Identifikation und Validierung von Erfolgsfaktoren aus strategischer und operativer Perspektive

Ziel und Ergebnis

Das **Ziel** der Arbeit ist es, **Erfolgsfaktoren** in sicherer Softwareentwicklung von Cloud-Anwendungen aus strategischer und operativer Perspektive unter besonderer Berücksichtigung von Systemen, Prozessen und Werkzeugen zu identifizieren und zu validieren, um das **Sicherheitsniveau von Cloud-Anwendungen** zu erhöhen.

Das **Ergebnis** der angestrebten Arbeit ist die Erstellung eines **Erfolgsfaktorenmodell** in sicherer Softwareentwicklung von Cloud-Anwendungen aus strategischer und operativer Perspektive unter besonderer Berücksichtigung von Systemen, Prozessen und Werkzeugen.

Forschungsfragen

1. Welche **Merkmale** zeichnen eine Cloud-Anwendung aus?
2. Wie kann **Erfolg** von sicherer Softwareentwicklung von Cloud-Anwendungen gemessen werden?
3. Welche **Rolle spielen strategische und operative Aspekte** in sicherer Softwareentwicklung von Cloud-Anwendungen?
4. Welche **Rolle spielen Systeme, Prozesse und Werkzeuge** in sicherer Softwareentwicklung von Cloud-Anwendungen?
5. Welche **theoretischen Grundlagen** erklären Erfolg in sicherer Softwareentwicklung von Cloud-Anwendungen aus strategischer und operativer Perspektive?
6. Welche **subjektiv-validierten Erfolgsfaktoren** bestehen in sicherer Softwareentwicklung von Cloud-Anwendungen aus strategischer und operativer Perspektive?
7. Welche **objektiv-validierten Erfolgsfaktoren** bestehen in sicherer Softwareentwicklung von Cloud-Anwendungen aus operativer Perspektive und gegebenenfalls strategischer Perspektive?

Bezugsrahmen der Studien I-III

Arbeitsschritt	Vorläufiges konzeptionelles Modell potenzieller Erfolgsfaktoren	Studie I: Qualitativ-explorative Vorstudie	Studie II: Quantitative Validierungsstudie	Studie III: Feldexperiment zur Validierung ausgewählter Erfolgsfaktoren
Fokus	Identifikation von potenziellen Erfolgsfaktoren aus dem Stand der Forschung	Identifikation von weiteren potenziellen Erfolgsfaktoren	Validierung der identifizierten potenziellen Erfolgsfaktoren anhand des wahrgenommenen Erfolgs	Weitere Validierung der Erfolgsfaktoren, welche durch ein Feldexperiment überprüft werden können (beispielsweise den Einfluss von Sicherheitswerkzeugen auf den Erfolg)
Funktion	**Selektion von potenziellen Erfolgsfaktoren**	**Selektion von potenziellen Erfolgsfaktoren**	**Explikation der Erfolgsfaktoren durch wahrgenommenen Erfolg (subjektive Erfolgsindikation)**	**Explikation der Erfolgsfaktoren durch gemessenen Erfolg (objektive Erfolgsindikation)**
Forschungsansatz	**Systematisches Literaturreview** Breite Literatursuche in einschlägigen Datenbanken (vom Brocke et al., 2009) Gezielte Literatursuche in Konferenz-Proceedings (Cooper and Hedge, 1993) Gezielte Literatursuche nach Research Agendas (vom Brocke et al., 2009)	**Empirische Forschungsmethode: Qualitative Forschung** *Datenerhebung:* Leitfadengestützte Experteninterviews (Bogner et al., 2014) *Datenauswertung:* Qualitative Inhaltsanalyse (Mayring, 2014) *Erkenntnislogik:* Induktiv (Sandberg, 2016) *Zielgruppen:* Oberste Verantwortliche für Informationssicherheit (strategisch), Entwickler & Tester (operativ)	**Empirische Forschungsmethode: Quantitative Forschung** *Datenerhebung:* Fragebogen (Wilde & Hess, 2007) *Datenauswertung:* Strukturgleichungsmodell (Wilde & Hess, 2007) *Erkenntnislogik:* Deduktiv (Sandberg, 2016) *Zielgruppen:* Oberste Verantwortliche für Informationssicherheit (strategisch), Entwickler & Tester (operativ)	**Konstruktives Paradigma** Feldexperiment (Wilde & Hess, 2007) I) Bereitstellung von Arbeitsumgebungen unter Einfluss der Erfolgsfaktoren (Gruppe A-n) und ohne Einfluss der Erfolgsfaktoren (Kontrollgruppe) II) Entwickler erstellen eine Cloud-Anwendung gemäß einer Aufgabenbeschreibung III) Die erstellte Software wird zur Erfolgsmessung nach Schwachstellen untersucht und nach CVSS 3.1 bewertet **Common Vulnerability Scoring System v3.1** Application Score
Theoretische Grundlagen	**Path Goal Theory** (Evans, 1970; House, 1971) **Social Exchange Theory** (Homans, 1958)	Voraussichtlich Path Goal Theory, Social Exchange Theory und/oder weitere/andere theoretische Grundlagen	Voraussichtlich Path Goal Theory, Social Exchange Theory und/oder weitere/andere theoretische Grundlagen	
Erwartetes Ergebnis	**Vorläufiges konzeptionelles Modell:** Bestehend aus potenziellen Erfolgsfaktoren aus der Literatur (beispielsweise Belohnung für sicheren Softwarecode, Richtlinien für Sicherheitswerkzeuge, Commitment zur Organisation, […])	**Erweitertes konzeptionelles Modell:** Bestehend aus potenziellen Erfolgsfaktoren aus der Literatur und aus der Vorstudie.	**Subjektiv-validiertes konzeptionelles Modell:** Bestehend aus Erfolgsfaktoren, die durch den wahrgenommen Erfolg bestätigt wurden.	**Objektiv-validiertes konzeptionelles Modell:** Bestehend aus Erfolgsfaktoren, die durch den wahrgenommen Erfolg und dem Erfolg im Feld bestätigt wurden.

Forschungsstand / Grundlagen

Erfolgsfaktorenforschung

Die **Erfolgsfaktorenforschung** legt die Annahme zugrunde, dass trotz einer Multidimensionalität des Erfolgs und einer Multikausalität von potenziellen Erfolgsfaktoren, einige wenige Erfolgsfaktoren herausgestellt werden können, die einen Erfolg maßgeblich beeinflussen (Baumgarth et al., 2009; Daschmann, 1994). Sie lässt sich nach den drei folgenden Funktionen gliedern (Daschmann, 1994): **Selektion**, **Explikation** und **Technologie**. Der Prozess basiert im Wesentlichen auf den Schritten der Selektion und Explikation. Bei der Selektion werden aus einer Vielzahl von Faktoren potenzielle Erfolgsfaktoren ausgewählt, welche einen Zusammenhang zum Erfolg haben können. Diese potenziellen Erfolgsfaktoren werden dann in der Explikation nach Erfolgsindikatoren bewertet. (Baumgarth et al., 2009; Daschmann, 1994).

Sichere Softwareentwicklung (Systeme, Prozesse, Werkzeuge)

Sichere Softwareentwicklung beschreibt eine Vorgehensweise, bestehend aus Phasen wie Design, Erstellung und Qualitätssicherung von Software, und integriert Sicherheitspraktiken in allen Phasen (**System**) (Dodson, 2019; Müller, 2018; Waidner, 2013). Ein **Prozess** (engl. Process) ist eine Menge von zusammenhängenden oder interagierenden Aktivitäten, die Eingaben in Ausgaben transformieren (ISO/IEC/IEEE, 2017). Er verfügt über Praktiken (engl. Practices), in denen Werkzeuge eingesetzt werden können. Eine Praktik ist eine bestimmte Aktivität, die bei Ausführung eines Prozesses einen Beitrag leistet (ISO/IEC/IEEE, 2017). Ein **Werkzeug** (engl. Tool) unterstützt Software- und Systemlebenszyklusprozesse (ISO/IEC/IEEE, 2017). Werkzeuge zur Entdeckung und Behebung von Schwachstellen im Programmcode werden in der Literatur als Sicherheitswerkzeuge (engl. Security Tools) bezeichnet (Witschey et al., 2015; Xiao et al., 2014).

Cloud-Anwendung vs. Mehrschicht-Anwendung

Eine **Cloud-Anwendung** ist eine Zusammenstellung von Cloud-Computing-Diensten (SaaS, PaaS oder IaaS) und unterscheidet sich von einer mehrschichtigen Architektur dadurch, dass sie nicht in Schichten, sondern in Dienste unterteilt ist, die den Strukturprinzipien einer Cloud-Anwendungsarchitektur folgen.

Nach dem **Cloud Application Maturity Model** kann eine Cloud-Anwendung in vier Reifegrade eingeteilt werden (ODCA, 2014): Eine Cloud-Anwendung im Reifegrad **Cloud ready** kann eine traditionelle Anwendung im Stil einer Mehrschichtarchitektur (Presentation Tier, Logic Tier und Data Tier) darstellen, die auf einer virtuellen Infrastruktur in einer Cloud betrieben wird. In der **Cloud friendly** Reifegradstufe werden lose gekoppelte Dienste auf Basis von Cloud-Design-Mustern verwendet. Cloud-Anwendungen in der nächsten Reifegradstufe **cloud resilient** zeichnen sich dadurch aus, dass ihr Zustand nur in wenigen Diensten isoliert ist und sie unabhängig von Ausfällen anderer Dienste sind. Die volle Reifegradstufe wird als **Cloud native** bezeichnet. Cloud-Anwendungen auf diesem Reifegrad sind in der Lage, das volle Potenzial des Cloud-Computing auszuschöpfen.

No.	Principle	Characteristics
1	Resilient to failure	Resiliency is designed into the application, rather than wrapped around it after the fact. Failures in cloud infrastructure are handled fluidly without interruption of service.
2	Resilient to latency	Applications adapt gracefully to latency rather than timing out/failing.
3	Secure	Applications are based on secure lifecycle standards and include built-in security. Data at rest and in transit is encrypted. APIs are protected by authentication and authorization.
4	Location independent	Applications discover services dynamically rather than relying on hard-coded dependencies.
5	Elastically scalable	Applications respond to demand levels, growing and shrinking as required, in/among clouds.
6	SOA/Compose-ability	Applications consume and expose web services with APIs discoverable at runtime. The structure incorporates small, stateless components designed to scale out.
7	Designed for manag.	Applications are instrumented and expose metrics and management interfaces.
8	Infrastructure independent	Applications make no assumptions about the underlying infrastructure, using abstractions in relation to the operating system, file system, database, and so on.
9	Defined tenancy	Each application should have a deliberate, defined single tenancy or multitenancy model.
10	Available end-user self-service	Users should be able to register themselves to use the app through a self-service registration interface, without entering an IT service request.
11	Bandwidth aware	APIs and application protocols are designed to minimize bandwidth consumption.
12	Cost/resource consumption aware	Application architecture is designed to minimize costs due to bandwidth, CPU, storage consumption, and I/O requests.

Literatur:
Baumgarth, Carsten, Martin Eisend, and Heiner Evanschitzky. 2009. 'Empirische Ma stertechniken'. Pp. 3–26 in Empirische Mastertechniken, edited by C. Baumgarth, M. Eisend, and H. Evanschitzky. Wiesbaden: Gabler Verlag.
Bedner, M. (2013): Cloud Computing. Technik, Sicherheit und rechtliche Gestaltung. Kassel: University Press.
Bogner, A., B. Littig, and W. Menz. 2014. Interviews Mit Experten: Eine Praxisorientierte Einführung. Springer Fachmedien Wiesbaden.
vom Brocke, Jan, Alexander Simons, Bjoern Niehaves, Kai Riemer, Ralf Plattfaut, Anne Cleven, and others. 2009. 'Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process.' Pp. 2206–17 in ECIS. Vol. 9.
Daschmann, H. A. 1994. Erfolgsfaktoren/-Erfolgsfaktoren Mittelständischer Unternehmen: Ein Beitrag Zur Erfolgsfaktorenforschung. Stuttgart.
Dhir, Saru, Deepak Kumar, and V. B. Singh. 2019. 'Success and Failure Factors That Impact on Project Implementation Using Agile Software Development Methodology'. Pp. 647–54 in Software Engineering. Vol. 731. Advances in Intelligent Systems and Computing, edited by M. N. Hoda, N. Chauhan, S. M. K. Quadri, and P. R. Srivastava.
Dodson, Donna, Murugiah Souppaya, and Karen Scarfone. 2020. Mitigating the Risk of Software Vulnerabilities by Adopting a Secure Software Development Framework (SSDF). National Institute of Standards and Technology. doi: 10.6028/NIST.CSWP.04232020.
Evans, Martin G. 1970. 'The Effects of Supervisory Behavior on the Path-Goal Relationship'. Organizational Behavior and Human Performance 5(3):277–98. doi: 10.1016/0030-5073(70)90021-8.
Freiling, Felix, Rüdiger Grimm, Kari Enni-Großpietsch, Hubert B. Keller, Jürgen Mottok, Isabel Münch, Kai Rannenberg, and Francesca Saglietti. 2014. 'Technische Sicherheit und Informationssicherheit: Unterschiede und Gemeinsamkeiten'. Informatik-Spektrum 37(1):14–24. doi: 10.1007/s00287-013-0748-2.
Heidkamp, Peter. Marko Vogel, and Lukas Gentemann. 2021. 'Bitkom Cloud Monitor 2021'.
Homans, George C. 1958. 'Social Behavior as Exchange'. American Journal of Sociology 63(6):597–606.
House, Robert J. 1971. 'A Path Goal Theory of Leader Effectiveness'. Administrative Science Quarterly 321–39.
ISO/IEC/IEEE. 2017. ISO/IEC/IEEE Systems and Software Engineering Vocabulary. ISO/IEC/IEEE 24765:2017(E) 1–541. doi: 10.1109/IEEESTD.2017.8016712.
Mayring, Philipp. 2014. 'Qualitative Content Analysis: Theoretical Foundation, Basic Procedures and Software Solution'.
Mell, P. & Grance, T. (2011): NIST Definition of Cloud Computing.
Müller, Klaus-Rainer. 2018. IT-Sicherheit mit System: Integratives IT-Sicherheits-, Kontinuitäts- und Risikomanagement – Sichere Anwendungen – Standards und Practices. Wiesbaden: Springer Fachmedien Wiesbaden.
Open Data Center Alliance (ODCA) 2014. 'Open Data Center Alliance: Best Practices: Architecting Cloud-Aware Applications Rev. 1.0'.
Sandberg, B. 2016. Wissenschaftliches Arbeiten von Abbildung Bis Zitat. Lehr- Und Übungsbuch Für Bachelor. Master Und Promotion. De Gruyter.
Schmalen, Caroline, Matthias Kunert, and Hannes Weindlmaier. 2006. Erfolgsfaktorenforschung: Theoretische Grundlagen, methodische Vorgehensweise und Anwendungserfahrungen in Projekten für die Ernährungsindustrie'. 14.
Suryateja, P. S. 2018. 'Threats and Vulnerabilities of Cloud Computing A Review'. International Journal of Computer Sciences and Engineering 6(3):297–302. doi: 10.26438/ijcse/v6i3.297302.
Tejay, Gurvinder P., and Markus Winkfield. n.d. 'How CISOs Can Become Effective Leaders? A Path-Goal Approach'. 11.
Waidner, Michael. 2013. Entwicklung sicherer Software durch Security by Design'. 76.
Wilde, Thomas, and Thomas Hess. 2007. 'Forschungsmethoden der Wirtschaftsinformatik'. 8.
Witschey, Jim, Olga Zielinska, Allaire Welk, Emerson Murphy-Hill, Chris Mayhorn, and Thomas Zimmermann. 2015. 'Quantifying Developers' Adoption of Security Tools'. Pp. 260–71 in Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2015. Bergamo, Italy: ACM Press.
Xiao, Shundan, Jim Witschey, and Emerson Murphy-Hill. 2014. 'Social Influences on Secure Development Tool Adoption: Why Security Tools Spread'. Pp. 1095–1106 in Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing - CSCW '14. Baltimore, Maryland, USA: ACM Press.

Marc Aurel Schubert, M.Sc.
InnoProm | IT Security und Datenschutz in der Cloud
Doktorand | Wissenschaftlicher Mitarbeiter

Forschungsgruppe Wirtschaftsinformatik und Medienmanagement
Prof. Dr. Sven Pagel - Fachbereich Wirtschaft
Hochschule Mainz - University of Applied Sciences
Lucy-Hillebrand-Straße 2 | 55128 Mainz | Raum A0.22
E marc.schubert@hs-mainz.de | innoprom-security.hs-mainz.de

Projektförderung: Die kooperative Promotion ist gefördert durch den Europäischen Fonds für regionale Entwicklung (EFRE), kofinanziert vom Ministerium für Wissenschaft und Gesundheit Rheinland-Pfalz (MWG RLP) und der sapite GmbH.

Ergebnisse der Studie I (qualitativ-explorative Studie)

Im Rahmen der qualitativ-explorativen Studie I wurden insgesamt **15 leitfaden-gestützte Interviews** mit Experten der strategischen (S) und operativen (O) Ebene durchgeführt. Zusätzlich wurde eine Mischform (M) berücksichtigt, die sowohl der strategischen als auch der operativen Ebene zugerechnet werden kann. Im Anschluss wurden die Interviews transkribiert und nach der qualitativen Inhaltsanalyse nach Mayring (2014) ausgewertet. Im Ergebnis konnten 38 potenzielle Erfolgsfaktoren identifiziert werden. Die Einordnung erfolgte nach den Dimensionen nach Dhir et al. (2019):

Organisation (13)

(1) (Advanced) security training for software developers SMO
(2) Alignment with standards/best practices O
(3) Analysis and measures for protection against foreign activities/sovereign acts S
(4) Establishing a security culture SO
(5) Expectation management on the topic of technical debt M
(6) Free spaces/time SO
(7) Knowledge exchange/lessons learned SMO
(8) Management attention S
(9) Management commitment S
(10) Platform strategy for cloud computing SM
(11) Security as a duty/license to operate SMO
(12) Security champions/experts MO
(13) Transparency towards the employees M

Process (9)

(1) Automated code, platform and container scanning S
(2) Development procedure with rules SMO
(3) Internal inspection process/quality gate/security audits / code reviews SMO
(4) Measurement by means of penetration testing SMO
(5) Secure construction kit/development platform SMO
(6) Security awareness SO
(7) Security guidelines for the development S
(8) Threat modelling (with focus on cloud app. architecture) SO
(9) Vulnerability identification and remediation SMO

Technology (7)

(1) Secure architecture and design of a cloud application S
(2) Use of an automated CI/CD pipeline SMO
(3) Use of cloud native technology SO
(4) Use of modern technology SO
(5) Use of secure libraries SM
(6) Use of serverless functions MO
(7) Use of tools SMO

Human (5)

(1) Self-responsibility SM
(2) Identification with the project MO
(3) Motivation (extrinsic) MO
(4) Motivation (intrinsic) SMO
(5) Organizational commitment SO

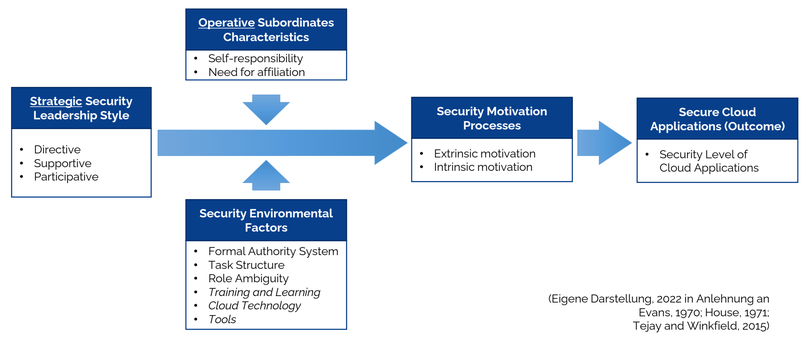
Project (4)

(1) Collection of error statistics in projects M
(2) Scope of the final project O
(3) Sufficient resources SO
(4) Sufficient time SO

Nr.	Rolle	Perspektive	Cloud-Erfahrung	Rolle-Erfahrung
1	CISO	Strategisch	10 Jahre	30 Jahre
2	CISO	Strategisch	7 Jahre	> 5 Jahre
3	CISO	Strategisch	10 Jahre	5 Jahre
4	CEO	Strategisch	10 Jahre	> 5 Jahre
5	Cloud Administrator	Operativ	7 Jahre	> 7 Jahre
6	Senior Cloud Engineer	Operativ	> 10 Jahre	> 20 Jahre
7	Senior Software Engineer	Operativ	9 Jahre	16 Jahre
8	Microsoft 365 Architect	Operativ	8 Jahre	5 Jahre
9	Senior Software Engineer	Operativ	5 Jahre	> 12 Jahre
10	Senior Software Engineer	Operativ	5 Jahre	15 Jahre
11	Leiter Softwareentwickler	Mischform	3 Jahre	14 Jahre
12	Cloud Sales Engineer und Security Architekt	Mischform	10 Jahre	8 Jahre
13	Managing Consultant	Mischform	7-8 Jahre	5 Jahre
14	Lead Auditor Informationssicherheit (BSI IT-Grundschutz / ISO 27001)	Mischform	13 Jahre	15 Jahre
15	Leiter Solution Arhictect	Mischform	7 Jahre	> 10 Jahre

Vorbereitung der Studie II (quantitativ-validierende Studie)

Im Rahmen der quantitativ-validierenden Studie II sollen die potenziellen Erfolgsfaktoren aus der Studie I weiter selektiert werden. Die weitere Selektion erfolgt nach dem Prozess von Schmalen et al. (2006), welcher aus drei Verfahren besteht: (1) Einbeziehung bereits vollzogener Erfolgsfaktorenstudien, (2) Setzen eines theoretischen Rahmens als Suchraum, (3) theoriegeleitete Ableitung von Hypothesen. Als theoretisches Modell wurde die Path Goal Theory (PGT) avisiert, da sie sowohl die strategische und operative Perspektive als auch die organisatorischen Konstrukte abbildet. Zur Passung wurde eine Zuordnung der potenziellen Erfolgsfaktoren zu den Variablen vorgenommen. Auf Basis bisheriger Studien wurde ein Online-Fragebogen konstruiert, der im April 2022 geschaltet wird.



(Eigene Darstellung, 2022 in Anlehnung an Evans, 1970; House, 1971; Tejay and Winkfield, 2015)

Stand 11.04.2022, Version 3-0-0