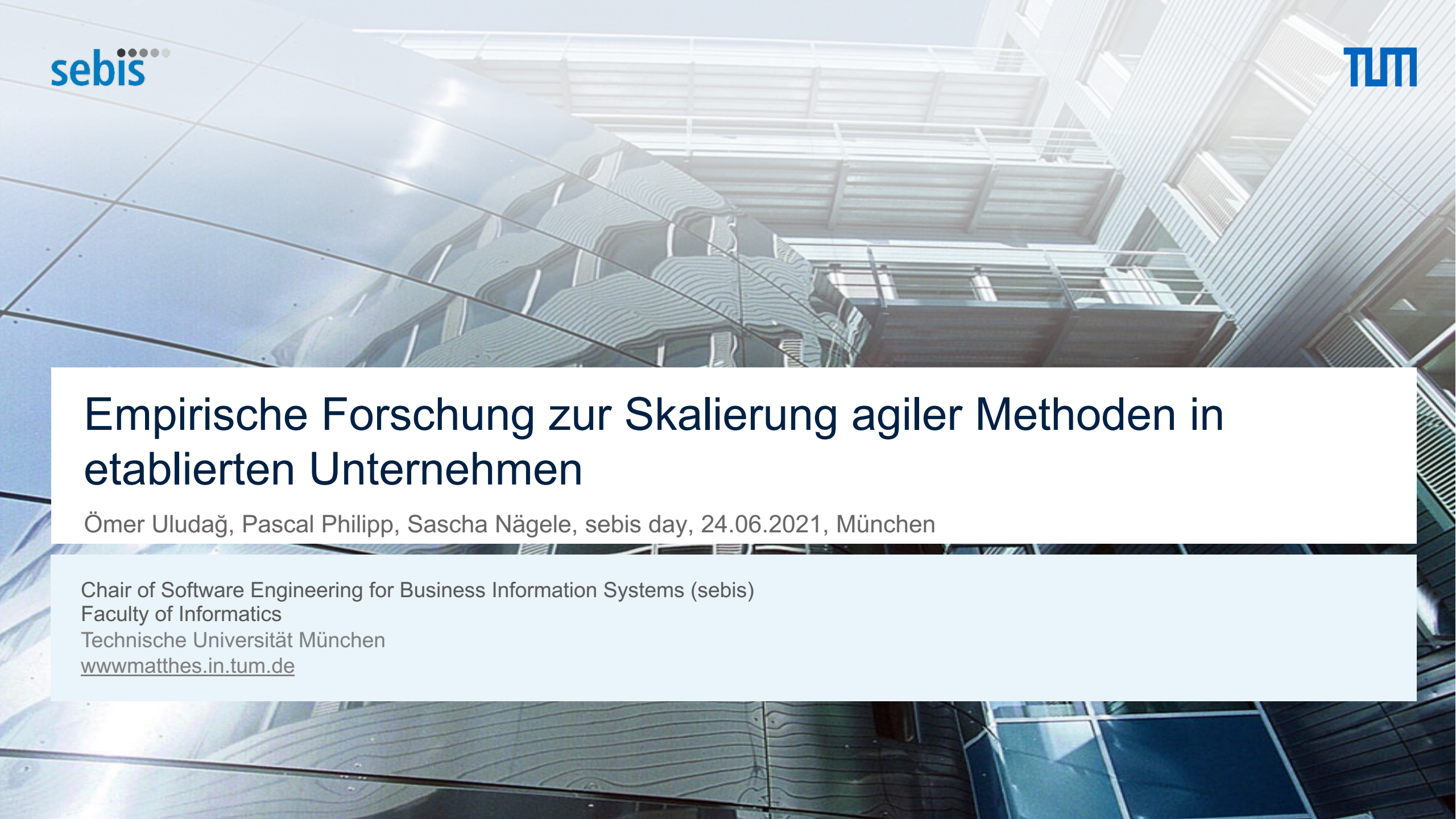


Empirische Forschung zur Skalierung agiler Methoden in etablierten Unternehmen

Ömer Uludağ, Pascal Philipp, Sascha Nägele, sebis day, 24.06.2021, München

Chair of Software Engineering for Business Information Systems (sebis)
Faculty of Informatics
Technische Universität München
wwwmatthes.in.tum.de

Compliance in Large-Scale Agile Development

Motivation

Agile software development		IT-Governance and compliance
Focus on functionality and working software	⟷	Ensuring non-functional requirements (e.g. security, architecture, ..)
Iterative, incremental development	⟷	Sequential control and formal review activities
Early and frequent delivery	⟷	Traceability and documentation for regulatory compliance
Emergent architecture design by self-organizing teams	⟷	Technical consistency, quality and standardization across teams
Self-organized, autonomous teams	⟷	Often complex hierarchical structures, quality assurance and audit processes

Importance of Governance and Compliance

An “agilist” might ask:

> *“Why is (centralized) IT governance and compliance still important in agile development, can‘t we just get rid of it?”*

In theory, purely self-organizing teams that govern themselves might be possible (with limitations where legislators require an external auditing entity).

But multiple areas with requirements independent from development methodology, e.g.:

- **Information security** and **data protection** (attackers do not care about the development method, and neither do legislators that e.g. impose fines if certain standards were not met)
- **(Enterprise) architecture**, e.g. for cross-team standardization to achieve more enterprise wide agility[1]
- **IT operations** and **business continuity management**, e.g. to ensure stability, performance etc.
- **UI/UX/Accessibility** (in certain cases, accessibility is even legally mandatory)
- **Risk management**
- Further **legal aspects** (e.g. open-source license compliance)

Even if the individual, self-organizing teams possess the skills and resources necessary to handle all these complex areas, it would **not be efficient** to solve these cross-cutting concerns independently.

[1] *Also see the „agility paradox“ by Ross, Weill, and Robertson (2006) in “Enterprise Architecture as Strategy”*

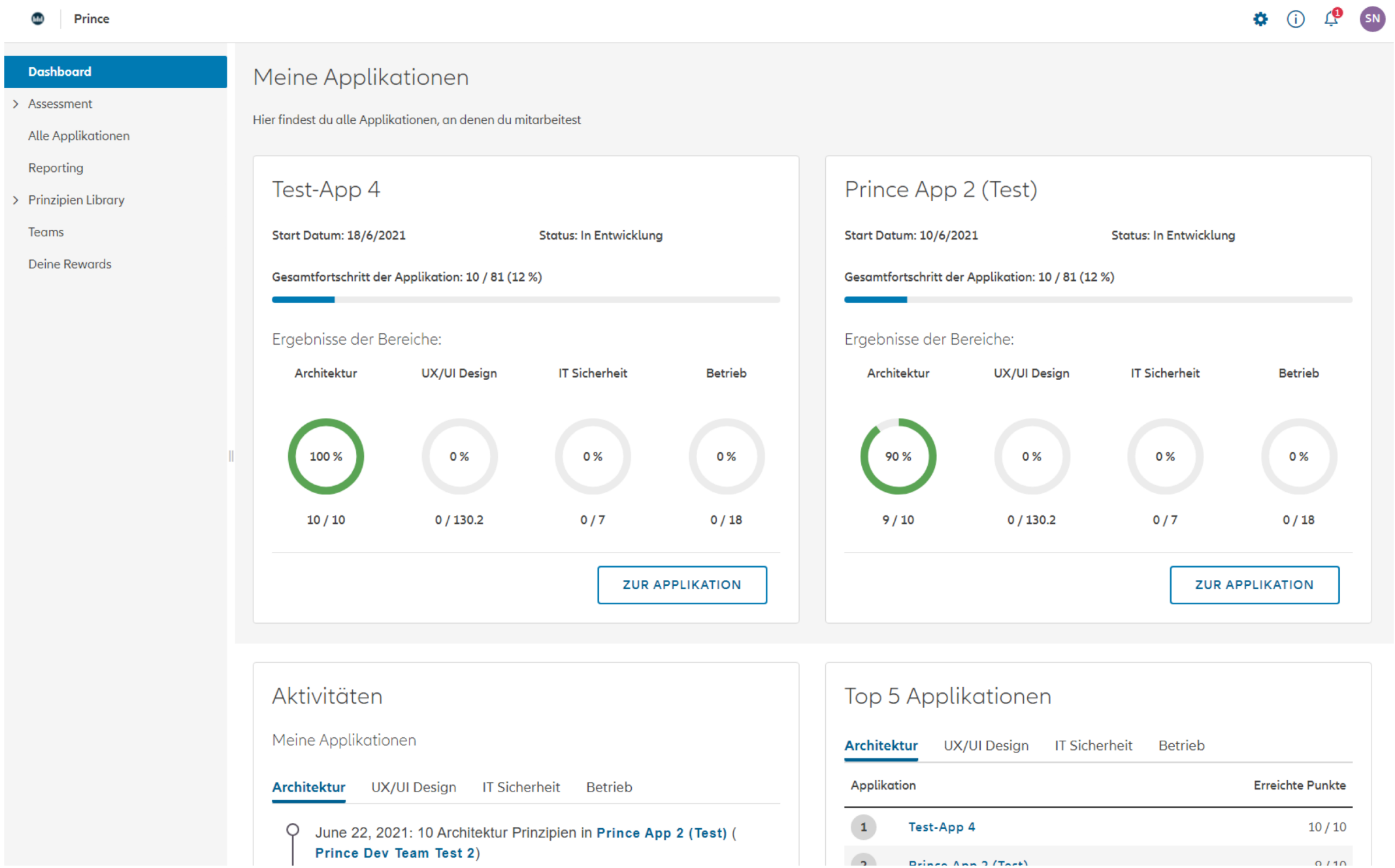
Planned research results

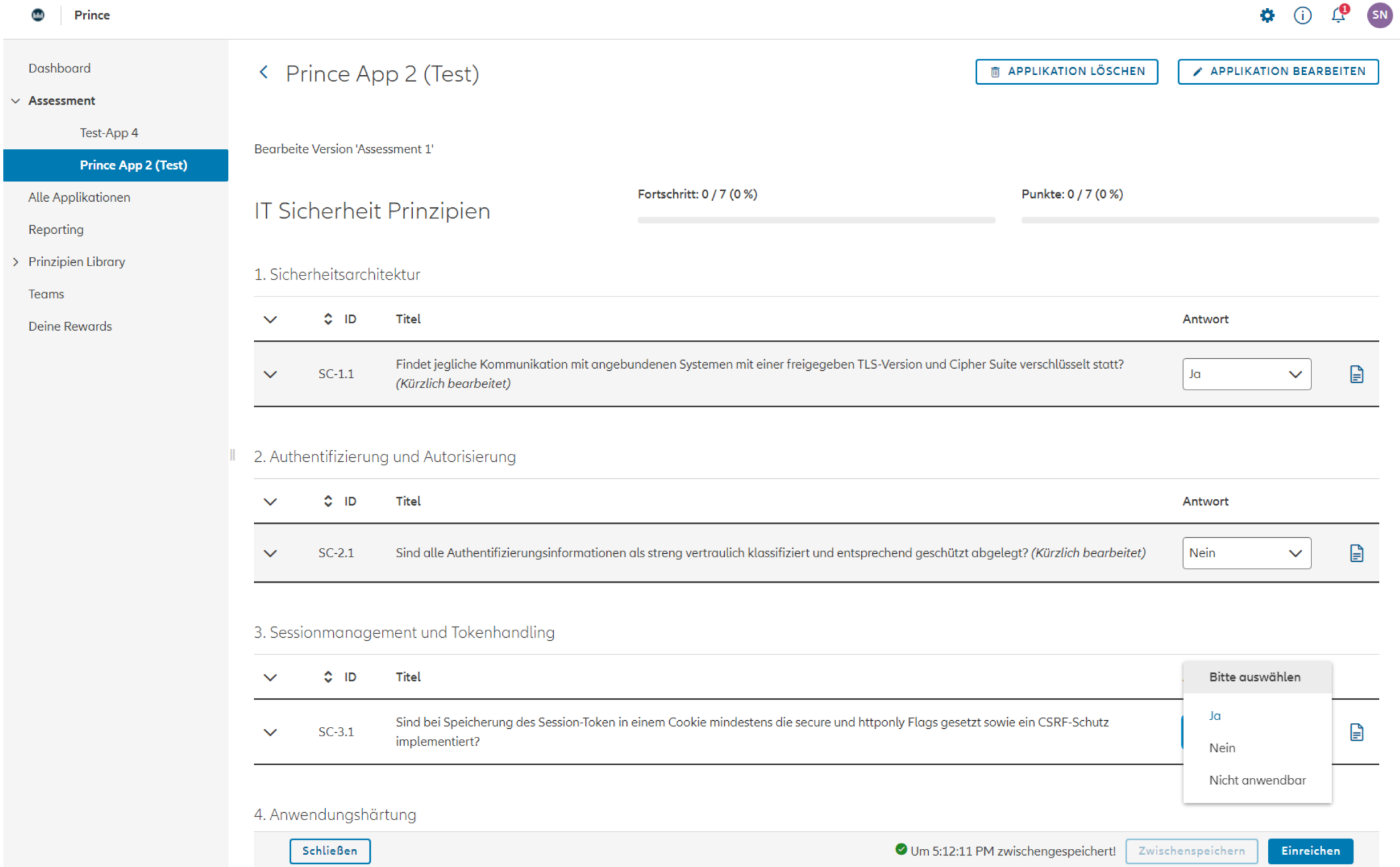
1. **Interaction and collaboration model** for defining, maintaining, governing and evaluating (security) standards in agile development at scale
   - Governance has to be a participatory or even collaborative process which is based on subject matter expertise instead of hierarchy
   - Standards are not driven top-down anymore by central departments, but expertise of agile teams is used in a structured way, focus is on self-commitment to standards that really matter

2. **Tool-support for collaborative, semi-automated self-assessments of standards**
   - Web application prototype for collaborative self-assessments of standards, with DevSecOps pipeline integration and automated testing
   - Allows the usage of community, collaboration, gamification and reporting features, which enables applying empiricism, systematic inspection and adaptation → fits well to “evidence-based management” approach of agile methods (e.g. Scrum)

3. **Security standards catalogue and maturity self-assessment model** to enable agile teams to take over more responsibility and self-govern their application security maturity level
   - Concept on how to categorize standards, integrate them into the iterative development and use them to self-assess the maturity level of a software application or development team
   - Concrete use case: Secure design, development and operations of web-based software applications, with examples in Angular and Spring Boot

Tool-Support Prototype


Prince
SN
Dashboard
Assessment
Alle Applikationen
Reporting
Prinzipien Library
Teams
Deine Rewards
Meine Applikationen
Hier findest du alle Applikationen, an denen du mitarbeitest
Test-App 4
Start Datum: 18/6/2021
Status: In Entwicklung
Gesamtfortschritt der Applikation: 10 / 81 (12 %)
Ergebnisse der Bereiche:
Architektur
UX/UI Design
IT Sicherheit
Betrieb
100 %
0 %
0 %
0 %
10 / 10
0 / 130.2
0 / 7
0 / 18
ZUR APPLIKATION
Prince App 2 (Test)
Start Datum: 10/6/2021
Status: In Entwicklung
Gesamtfortschritt der Applikation: 10 / 81 (12 %)
Ergebnisse der Bereiche:
Architektur
UX/UI Design
IT Sicherheit
Betrieb
90 %
0 %
0 %
0 %
9 / 10
0 / 130.2
0 / 7
0 / 18
ZUR APPLIKATION
Aktivitäten
Meine Applikationen
Architektur
UX/UI Design
IT Sicherheit
Betrieb
June 22, 2021: 10 Architektur Prinzipien in Prince App 2 (Test) (Prince Dev Team Test 2)
Top 5 Applikationen
Architektur
UX/UI Design
IT Sicherheit
Betrieb
Applikation
Erreichte Punkte
1
Test-App 4
10 / 10


Prince
SN
Dashboard
Assessment
Test-App 4
Prince App 2 (Test)
Alle Applikationen
Reporting
Prinzipien Library
Teams
Deine Rewards
Prince App 2 (Test)
APPLIKATION LÖSCHEN
APPLIKATION BEARBEITEN
Bearbeite Version 'Assessment 1'
IT Sicherheit Prinzipien
Fortschritt: 0 / 7 (0 %)
Punkte: 0 / 7 (0 %)
1. Sicherheitsarchitektur
ID
Titel
Antwort
SC-1.1
Findet jegliche Kommunikation mit angebundenen Systemen mit einer freigegeben TLS-Version und Cipher Suite verschlüsselt statt? (Kürzlich bearbeitet)
Ja
2. Authentifizierung und Autorisierung
ID
Titel
Antwort
SC-2.1
Sind alle Authentifizierungsinformationen als streng vertraulich klassifiziert und entsprechend geschützt abgelegt? (Kürzlich bearbeitet)
Nein
3. Sessionmanagement und Tokenhandling
ID
Titel
SC-3.1
Sind bei Speicherung des Session-Token in einem Cookie mindestens die secure und httponly Flags gesetzt sowie ein CSRF-Schutz implementiert?
Bitte auswählen
Ja
Nein
Nicht anwendbar
4. Anwendungshärtung
Schließen
Um 5:12:11 PM zwischengespeichert!
Zwischenspeichern
Einreichen

Further intermediary results

Security-focused:

1. Multi-case study on security governance in large-scale agile development with (so far) eight companies
2. Systematic analysis and empirical evaluation of solution approaches for agile security (at scale)
   - More than 100 publications identified, ~30 promising solution approaches left after first quality criteria filtering
   - Further filtering planned based on expert interviews and experiments
3. Systematic multi-vocal literature review on security activity automation („DevSecOps“) in software development
   - 1800 relevant publications identified, filtering resulted in ~300 DevSecOps tools
   - ➢ DevSecOps tool taxonomy and mapping with secure software development activities
4. Analysis of current (web) application vulnerabilities and funnel approach to reduce manual effort of agile development teams, case study in a software development consultancy

Broader perspective:

5. Analysis of popular large-scale agile frameworks and their maturity on governance and compliance
6. Case study in the automotive sector on IT compliance documentation requirements and agile development at scale

Contact

If you are interested in the area of (security) governance and compliance in large-scale agile software development, please feel free to get in touch with me:

sascha.naegele@tum.de

Backup

Übersicht der agilen Informationssicherheitsansätzen

Rollen & Wissen

- InfoSec Training
- InfoSec Wissensaustausch
- Beratende InfoSec Rolle
- InfoSec Experte im Team
- InfoSec Verifizierungsrolle
- Security Architekt
- Security Master

Prozess & Methodik

<u>Sicherheitsanforderungen</u>
- Security (User) Stories
- Security Backlog
- Misuse/Abuse Cases
- Kriterien in Definition of Done
- Rollen-Matrix
- Explizite InfoSec Anforderungen

<u>Sicheres Design & Implementieren</u>
- Agile Analyse/Modellierung von Sicherheitsbedrohungen
- Agile Risikoanalyse
- Secure Coding Guidelines
- Sicherheitsarchitektur
- Pair Programming
- InfoSec Meetings & Sprints
- Genehmigte InfoSec Tools
- Dokumentationsartefakte

<u>Sicherheitsverifizierung</u>
- Quality Gates
- Sicherheitstests
- Security Code Review
- Automatisierung von Prozesselementen

Ganzheitlicher Ansatz

- Neuer oder erweiterter agiler Entwicklungsprozess
- Integrations- oder Bewertungsansatz