

# **Blatt 6**

## Software Engineering 2

Luis Staudt

## Aufgabe 1

### Fragen der KI

1. Was sind Ihre Gründe dafür, dass Sie “einfachere Verwaltung“ als Vorteil des zentralisierten Modells ansehen? Könnte dies unter bestimmten Umständen auch ein Nachteil sein? (Beispiel 1: Nennen Sie uns Ihre Gründe für diese Behauptung)
2. Welche Schlussfolgerungen ziehen Sie aus der Tatsache, dass beide Modelle gegensätzliche Vor- und Nachteile aufweisen? Gibt es einen idealen Mittelweg? (Beispiel 2: Wie lautet Ihre Schlussfolgerung?)
3. Warum halten Sie die “größere Angriffsfläche“ im verteilten Modell für relevant, obwohl gleichzeitig das Risiko eines vollständigen Datenverlusts geringer ist? (Beispiel 3: Warum glauben Sie das?)
4. Welche Argumente sprechen dafür und dagegen, dass ein verteiltes Modell in Zeiten zunehmender Cloud-Nutzung und Remote-Arbeit die bessere Wahl ist? (Beispiel 4: Was sind die Argumente dafür und dagegen?)
5. Welche Annahmen über die IT-Infrastruktur und das Know-how eines Unternehmens müssen wir machen, um ein verteiltes Sicherheitsmodell zu empfehlen? (Beispiel 5: Welche Annahmen müssen wir machen?)
6. Worauf stützen Sie Ihre Behauptung, dass das zentralisierte Modell “konsistente Durchsetzung von Sicherheitsrichtlinien“ bietet? Gibt es empirische Belege dafür? (Beispiel 6: Worauf stützen Sie sich bei dieser Aussage?)
7. Welche zusätzlichen Dimensionen müssten wir betrachten, um zu entscheiden, welches Modell für ein konkretes Unternehmen besser geeignet ist? (Beispiel 11: Welche zusätzlichen Informationen benötigen wir?)
8. Welche Konsequenzen hätte es für ein Unternehmen, wenn es sich für ein hybrides Sicherheitsmodell entscheiden würde, das Elemente beider Ansätze kombiniert? (Beispiel 13: Was sind die Konsequenzen?)
9. Welche alternativen Sicherheitsarchitekturen jenseits des rein zentralisierten oder verteilten Modells wurden bisher nicht berücksichtigt? (Beispiel 14: Welche Alternativen haben wir noch nicht erforscht?)

10. Gibt es potenzielle unerwünschte Folgen bei der Implementierung eines verteilten Modells, die nicht sofort erkennbar sind, aber langfristig problematisch werden könnten? (Beispiel 16: Gibt es mögliche unerwünschte Folgen?)

### **Antworten der KI Fragen**

Gründe für einfachere Verwaltung als Vorteil des zentralisierten Modells:

Bei einem zentralisierten Modell muss ich nur an einem Ort Änderungen vornehmen und habe alles im Blick. Allerdings kann das bei wachsenden Unternehmen zum Problem werden, weil dann die zentrale Verwaltung zum Flaschenhals wird.

Schlussfolgerungen aus gegensätzlichen Vor- und Nachteilen:

Ich denke, dass ein Mix aus beiden Modellen am sinnvollsten ist, da man so die Vorteile kombinieren kann. Zentrale Richtlinien mit lokaler Flexibilität wären mein Ansatz.

Relevanz der "größeren Angriffsfläche" im verteilten Modell:

Mehr Systeme bedeuten mehr potenzielle Einstiegspunkte für Angreifer, auch wenn nicht alles auf einmal ausfallen kann. Ich muss bei einem verteilten System einfach viel mehr absichern und überwachen.

Argumente für/gegen verteiltes Modell bei Cloud-Nutzung:

Für Remote-Arbeit passt ein verteiltes Modell besser, weil es flexibler ist und nicht alles zusammenbricht, wenn ein Teil ausfällt. Andererseits macht es die Verwaltung komplizierter und es ist schwerer, überall die gleichen Sicherheitsstandards durchzusetzen.

Annahmen für verteiltes Sicherheitsmodell:

Ich setze voraus, dass das Unternehmen genug IT-Security-Experten und Geld für mehrere Sicherheitssysteme hat. Außerdem müssten gute Monitoring-Tools und eine Kultur der Zusammenarbeit zwischen den Teams existieren.

Grundlage für "konsistente Durchsetzung von Sicherheitsrichtlinien":

Wenn eine zentrale Stelle die Regeln festlegt, ist es wahrscheinlicher, dass überall das Gleiche gilt. In der Praxis zeigen die meisten Sicherheitsvorfälle, dass unterschiedliche Implementierungen von Richtlinien oft das Problem sind.

Zusätzliche Dimensionen für die Modellentscheidung:

Ich müsste noch die Unternehmenskultur, gesetzliche Anforderungen und die geografische Verteilung des Unternehmens berücksichtigen. Auch das Budget und die Art der Daten spielen eine wichtige Rolle bei der Entscheidung.

Konsequenzen eines hybriden Sicherheitsmodells:

Ein hybrides Modell könnte mehr Flexibilität bieten, macht aber alles komplexer zu verwalten. Die größte Herausforderung wäre, klare Zuständigkeiten zu definieren und die Kommunikation zwischen zentralen und dezentralen Teams sicherzustellen.

Alternative Sicherheitsarchitekturen:

Ich finde, dass Mikrosegmentierung mit granularer Zugriffskontrolle oder KI-gestützte adaptive Sicherheitsmodelle interessante Alternativen sind. Man könnte auch über software-definierte Sicherheit nachdenken, bei der Sicherheitskontrollen programmierbar sind.

Potenzielle unerwünschte Folgen eines verteilten Modells:

Mit der Zeit könnten verschiedene Teile des Systems unterschiedliche Standards entwickeln, was die Sicherheit insgesamt schwächt. Außerdem besteht die Gefahr einer Kostenexplosion durch die Duplikation von Sicherheitssystemen an verschiedenen Standorten.

## **Feedback der KI zu den Lösungen**

Vielen Dank für Ihre durchdachten Antworten. Sie zeigen eine gute Auseinandersetzung mit den verschiedenen Aspekten von Sicherheitsarchitekturen.

Stärken Ihrer Analyse:

- Sie erkennen gut die Kontextabhängigkeit der Vor- und Nachteile (z.B. dass "einfachere Verwaltung" bei Wachstum zum Flaschenhals werden kann)
- Ihre Überlegung zu einem hybriden Modell zeigt pragmatisches Denken
- Sie beziehen wichtige organisatorische Faktoren wie Unternehmenskultur und Ressourcenverfügbarkeit ein
- Ihre Antwort zu alternativen Architekturen (Mikrosegmentierung, KI-gestützte Modelle) demonstriert einen Blick über den Tellerrand

Potenzial für vertieftes kritisches Denken:

- Bei den “Annahmen für verteiltes Sicherheitsmodell” könnten Sie technologische Voraussetzungen noch detaillierter betrachten
- Ihre Antwort zur “konsistenten Durchsetzung von Sicherheitsrichtlinien” könnte durch konkrete Beispiele oder Studien gestützt werden
- Die Diskussion zu unerwünschten Folgen könnte noch Aspekte wie Datenhoheit oder komplexe Compliance-Anforderungen berücksichtigen

### **Ausarbeitung - Vergleich der alten und neuen Sicherheitsmodell-Lösung**

Hier ist mein Vergleich zwischen meiner ursprünglichen Lösung zu Sicherheitsmodellen und den später erarbeiteten Erkenntnissen.

Meine ursprüngliche Lösung

In meiner ersten Lösung stellte ich lediglich die Vor- und Nachteile von zentralisierten und verteilten Sicherheitsmodellen gegenüber. Ich betrachtete grundlegende Faktoren wie Verwaltungsaufwand, Kosten, Single Point of Failure und Skalierbarkeit. Diese Analyse war für den Anfang ausreichend, aber im Rückblick recht oberflächlich.

Neue Erkenntnisse

#### **Hybride Ansätze**

In der neuen Lösung erkannte ich, dass die Wahl nicht auf komplett zentralisiert oder verteilt beschränkt sein muss. Hybride Modelle mit zentralen Richtlinien und lokaler Flexibilität bieten eine Alternative, die ich vorher nicht bedacht hatte.

#### **Praxisbezug**

Während meine erste Analyse theoretisch blieb, berücksichtigt die neue Lösung den Anwendungskontext. Remote-Arbeit und Cloud-Nutzung begünstigen beispielsweise verteilte Modelle.

#### **Implementierungsvoraussetzungen**

Für verteilte Systeme werden ausreichende finanzielle Mittel, Security-Experten und geeignete Monitoring-Werkzeuge benötigt - Faktoren, die in meiner ersten Lösung fehlten.

#### **Alternative Architekturen**

Es existieren weitere Ansätze wie Mikrosegmentierung und KI-gestützte Sicherheitsmodelle, die ich in der ursprünglichen Analyse nicht erwähnt hatte.

## **Langzeitfolgen**

Mit der Zeit können Standards divergieren und Kosten steigen. Diese zeitliche Dimension fehlte in meiner statischen Erstanalyse.

## **Organisatorische Faktoren**

Unternehmenskultur und Teamkommunikation sind wichtige Faktoren, die ich unterschätzt hatte. Ich konzentrierte mich zu sehr auf technische Aspekte.

## **Regulatorische Anforderungen**

Je nach Datentyp und geltenden Gesetzen variieren die Sicherheitsanforderungen. Diese Differenzierung berücksichtigte ich anfangs nicht ausreichend.

## **Fazit**

Die neue Lösung ist praxisnäher und realitätsbezogener. Anstatt nur zwei Modelle zu vergleichen, berücksichtige ich nun hybride Ansätze, den Anwendungskontext und nicht-technische Faktoren.

Es wird deutlich, dass Sicherheitsmodelle mit Unternehmenskultur, verfügbaren Ressourcen und gesetzlichen Vorgaben verknüpft sind. Der vereinfachte Stil der neuen Lösung macht die Komplexität des Themas besser verständlich. Das Thema erwies sich als vielschichtiger als anfangs angenommen.

# **Aufgabe 2**

## **Fragen der KI**

1. Welche spezifischen Anforderungen haben Finanzunternehmen an Softwareprodukte im Vergleich zu anderen Branchen? (Beispiel 11: Welche zusätzlichen Informationen benötigen wir?)
2. Welche Konsequenzen hätte eine falsche Architekturentscheidung in Bezug auf die Sicherheit von Finanzdaten? (Beispiel 13: Was sind die Konsequenzen?)
3. Welche Annahmen müssen wir über die technische Infrastruktur von Finanzunternehmen machen, um bestimmte Architekturentscheidungen zu treffen? (Beispiel 5: Welche Annahmen müssen wir machen?)

4. Wie würden sich Ihre Prioritäten ändern, wenn Sie wüssten, dass die Finanzunternehmen stark regulierten Compliance-Anforderungen unterliegen? (Beispiel 10: Was würde sich ändern?)
5. Welche Argumente sprechen für und gegen die Priorisierung von Softwarekompatibilität im Finanzsektor? (Beispiel 4: Was sind die Argumente dafür und dagegen?)
6. Welche möglichen unerwünschten Folgen könnte eine Architektur haben, die primär auf eine hohe Anzahl von Benutzern ausgerichtet ist? (Beispiel 16: Gibt es mögliche unerwünschte Folgen?)
7. Auf welche Weise beeinflussen die nicht-funktionalen Produkteigenschaften die Akzeptanz der Software in Finanzunternehmen? (Beispiel 6: Worauf stützen Sie sich bei dieser Aussage?)
8. Welche Schlussfolgerungen können wir über die Wichtigkeit der Produktlebensdauer ziehen, wenn wir die schnelle technologische Entwicklung im Finanzsektor berücksichtigen? (Beispiel 7: Welche Schlussfolgerungen können wir ziehen?)
9. Welche Alternativen zur Priorisierung dieser fünf Aspekte haben wir noch nicht betrachtet? (Beispiel 14: Welche Alternativen haben wir noch nicht erforscht?)
10. Wie würden sich Ihre Architekturentscheidungen auf die zukünftige Wartbarkeit und Erweiterbarkeit des Softwareprodukts auswirken? (Beispiel 12: Welche Auswirkungen hätte es auf unser zukünftiges Handeln?)

### Antworten der KI Fragen

Welche spezifischen Anforderungen haben Finanzunternehmen an Softwareprodukte im Vergleich zu anderen Branchen?

Im Vergleich zu anderen Branchen legen Finanzunternehmen extremen Wert auf Sicherheit, Datenschutz und Compliance. Ich denke, wir müssten uns hier zusätzliche Informationen einholen – zum Beispiel über gesetzliche Vorschriften wie BaFin oder PCI-DSS. Auch wie gut sich die Software in bestehende, oft sehr alte Systeme integrieren lässt, spielt eine große Rolle.

Welche Konsequenzen hätte eine falsche Architekturentscheidung in Bezug auf die Sicherheit von Finanzdaten?

Meiner Meinung nach können solche Fehler echt gravierende Folgen haben. Wenn z.B. Kundendaten oder Transaktionen kompromittiert werden, sind nicht nur hohe Strafen, sondern auch ein großer Vertrauensverlust bei den Kunden wahrscheinlich. Im schlimmsten Fall könnte das sogar die Existenz des Unternehmens gefährden.

Welche Annahmen müssen wir über die technische Infrastruktur von Finanzunternehmen machen, um bestimmte Architekturentscheidungen zu treffen?

Ich gehe davon aus, dass viele Finanzunternehmen noch mit einer Mischung aus lokalen Rechenzentren und Cloud-Lösungen arbeiten. Auch starke Netzsegmentierung und Hardware-Sicherheitsmodule (HSMs) könnten üblich sein. Solche Annahmen sind wichtig, um z.B. zu entscheiden, ob Microservices sinnvoll sind oder eher nicht.

Wie würden sich Ihre Prioritäten ändern, wenn Sie wüssten, dass die Finanzunternehmen stark regulierten Compliance-Anforderungen unterliegen?

In dem Fall würde ich ganz klar Dinge wie Auditierbarkeit, sichere Datenhaltung und gute Zugriffskontrollen priorisieren. Features wie flexible Rollenvergabe und nachvollziehbare Logs wären dann viel wichtiger als etwa Time-to-Market oder Benutzerfreundlichkeit.

Welche Argumente sprechen für und gegen die Priorisierung von Softwarekompatibilität im Finanzsektor?

Ein Argument *für* Kompatibilität ist auf jeden Fall, dass viele Unternehmen noch auf alten Systemen laufen – eine gute Integration spart Zeit und Geld. Andererseits kann es auch ein Nachteil sein, weil dadurch moderne Architekturen ausgebremst werden. Man läuft Gefahr, die Software unnötig kompliziert und schwer wartbar zu machen.

Welche möglichen unerwünschten Folgen könnte eine Architektur haben, die primär auf eine hohe Anzahl von Benutzern ausgerichtet ist?

Wenn man zu sehr auf Skalierbarkeit achtet, kann es passieren, dass Sicherheits- oder Transaktionsanforderungen vernachlässigt werden. Außerdem wird die Lösung vielleicht zu generisch, sodass wichtige branchenspezifische Anforderungen unter den Tisch fallen.

Auf welche Weise beeinflussen die nicht-funktionalen Produkteigenschaften die Akzeptanz der Software in Finanzunternehmen?

Ich glaube, dass Eigenschaften wie Verfügbarkeit, Performance oder auch Nachvollziehbarkeit genauso wichtig wie Funktionalität sind. Wenn eine Software nicht zuverlässig oder schwer integrierbar ist, wird sie selbst dann abgelehnt, wenn sie eigentlich alle Features mitbringt.

Welche Schlussfolgerungen können wir über die Wichtigkeit der Produktlebensdauer ziehen, wenn wir die schnelle technologische Entwicklung im Finanzsektor berücksichtigen?

Ich denke, man sollte Software nicht einfach „für ewig“ bauen, sondern eher modular und flexibel. So kann man auf neue Anforderungen reagieren, ohne gleich alles neu machen zu müssen. Langlebigkeit ist dann mehr ein Nebeneffekt guter Erweiterbarkeit.



Welche Alternativen zur Priorisierung dieser fünf Aspekte haben wir noch nicht betrachtet?

Vielleicht könnten wir mehr in Richtung adaptive Architekturen oder den Einsatz von KI zur Einhaltung von Regeln denken. Auch „Security by Design“ als durchgehendes Konzept statt einzelner Sicherheitsmaßnahmen wäre eine interessante Alternative, die wir noch nicht vollständig durchdacht haben.

Wie würden sich Ihre Architekturentscheidungen auf die zukünftige Wartbarkeit und Erweiterbarkeit des Softwareprodukts auswirken?

Wenn man z.B. auf saubere Modularisierung achtet, wird es später viel einfacher, neue Funktionen zu ergänzen oder Fehler zu beheben. Ich finde, solche Entscheidungen wirken sich massiv auf den Pflegeaufwand und die Weiterentwicklung aus – gerade bei Software, die viele Jahre im Einsatz ist.

### **Feedback der KI zu den Lösungen**

Vielen Dank für die ausführliche Bearbeitung der von mir vorgeschlagenen Fragen. Ich sehe, dass Sie sich intensiv mit den verschiedenen Aspekten der Architekturentscheidungen für Finanzunternehmen auseinandergesetzt haben.

Stärken Ihrer Ausarbeitung:

- Tiefgehende Branchenkenntnis: Ihre Antworten zeigen ein gutes Verständnis der spezifischen Anforderungen im Finanzsektor (Sicherheit, Compliance, Integration mit Altsystemen).
- Ausgewogene Betrachtung: Sie haben bei mehreren Fragen Pro- und Kontra-Argumente angeführt, was ein hohes Maß an kritischem Denken demonstriert.
- Praktische Relevanz: Ihre Überlegungen sind praxisnah und berücksichtigen reale Herausforderungen, wie die Balance zwischen Modernisierung und Kompatibilität.
- Zukunftsorientierung: Besonders bei Fragen zur Produktlebensdauer und Wartbarkeit zeigen Sie vorausschauendes Denken.

Anregungen zur Vertiefung:

- Bei der Betrachtung nicht-funktionaler Eigenschaften könnten Sie noch stärker auf die Priorisierung und mögliche Konflikte zwischen verschiedenen Qualitätsattributen eingehen.
- Die Perspektive der verschiedenen Stakeholder (Entwickler, Endnutzer, Compliance-Beauftragte) könnte noch detaillierter herausgearbeitet werden.

## **Ausarbeitung - Architekturaspekte für Finanzunternehmen**

### Einleitung

In meiner ursprünglichen Lösung hatte ich bereits die Softwarekompatibilität und nichtfunktionalen Produkteigenschaften als wichtigste Architekturaspekte für Finanzunternehmen identifiziert. Nach der Analyse des Zusatzmaterials erkenne ich jetzt, dass meine Betrachtung zu oberflächlich war und wichtige Dimensionen fehlten.

### Vertiefung der Sicherheits- und Compliance-Anforderungen

Ich habe zwar Sicherheit und Compliance erwähnt, aber nicht wirklich verstanden, wie *extrem* wichtig diese Faktoren sind. Die möglichen Konsequenzen von Sicherheitsmängeln können sogar existenzbedrohend sein! Auch die konkreten regulatorischen Anforderungen wie BaFin oder PCI-DSS hatte ich nicht auf dem Schirm.

### Technische Infrastruktur als Entscheidungsbasis

Einen komplett neuen Aspekt habe ich jetzt durch die Berücksichtigung der technischen Infrastruktur von Finanzunternehmen hinzugewonnen. Die typische Mischung aus lokalen Rechenzentren und Cloud-Lösungen, sowie spezielle Sicherheitsmaßnahmen wie Netzsegmentierung und HSMs beeinflussen direkt, ob z.B. Microservices sinnvoll sind oder nicht.

### Das Dilemma zwischen Alt und Neu

Ich hatte nicht das Spannungsfeld zwischen Kompatibilität und Innovation erkannt. Einerseits spart die Integration mit Altsystemen Zeit und Geld, andererseits kann sie moderne Ansätze ausbremsen und unnötige Komplexität schaffen.

### Modularität statt für die Ewigkeit

Neu ist für mich der Gedanke, dass man Software nicht einfach langlebig bauen sollte, sondern modular und flexibel, damit man auf neue Anforderungen reagieren kann. Langlebigkeit ergibt sich dann quasi als Nebeneffekt einer guten Erweiterbarkeit.

### Auditierbarkeit vor Benutzerfreundlichkeit

Bei stark regulierten Anforderungen müssen Dinge wie nachvollziehbare Logs und flexible Rollenvergabe vor Benutzerfreundlichkeit und schneller Markteinführung priorisiert werden – ein Aspekt, den ich unterschätzt hatte.

## Überoptimierung vermeiden

Eine Warnung, die ich mitnehme: Zu starke Fokussierung auf einzelne Aspekte wie Skalierbarkeit kann dazu führen, dass man Sicherheits- oder Transaktionsanforderungen vernachlässigt und zu generische Lösungen entwickelt.

## Fazit

Insgesamt habe ich durch die zusätzlichen Materialien viel dazugelernt. Meine ursprüngliche Einschätzung war zwar grundsätzlich richtig, aber viel zu einfach gedacht. Vor allem hatte ich die Spannungsfelder zwischen verschiedenen Anforderungen nicht ausreichend betrachtet.

Besonders wertvoll finde ich jetzt das Konzept "Security by Design" als durchgängiges Prinzip und nicht nur als isolierte Maßnahme. Auch der Gedanke adaptiver Architekturen und der mögliche Einsatz von KI zur Compliance-Einhaltung sind spannende neue Ansätze für mich.

Für zukünftige Architekturentscheidungen im Finanzsektor werde ich diese differenziertere Betrachtung nutzen und besonders auf die Balance zwischen Innovation und Integration mit Bestandssystemen achten – immer mit dem Fokus auf Sicherheit und Compliance als unverhandelbaren Grundpfeilern.

## Aufgabe 3

Fragen	Stimme zu	Neutral	Stimme NICHT zu
Die Verwendung von KI zur Beantwortung von Fragen verbessert mein Verständnis des Stoffes	X		
Der Einsatz von KI zur Beantwortung von Fragen hilft mir, kritischer über den Lernstoff nachzudenken		X	
Ich vertraue auf die Genauigkeit und Relevanz von KI-generierten Antworten, wenn ich lerne		X	
KI-generierte Antworten sind hilfreicher als die Antworten von Dozenten oder meinen Kommilitonen			X
Der Einsatz von KI zur Beantwortung von Fragen verbessert mein Lernerlebnis insgesamt	X		

## **Einsatz von KI beim Lernen**

Der Einsatz von Künstlicher Intelligenz (KI) beim Lernen bietet viele Chancen, aber auch einige Herausforderungen. Aus meiner Sicht kann KI besonders hilfreich sein, um individuelles Lernen zu unterstützen – etwa durch personalisierte Lernpfade, automatische Rückmeldungen oder adaptive Schwierigkeitsanpassung. Lernplattformen, die KI einsetzen, können erkennen, wo man Schwierigkeiten hat, und gezielt passende Inhalte vorschlagen. Das spart Zeit und sorgt dafür, dass man effizienter lernt.

Außerdem ermöglicht KI einen niederschweligen Zugang zu Wissen, zum Beispiel über Chatbots oder intelligente Tutorsysteme, die jederzeit Fragen beantworten können. Ich finde das besonders nützlich beim Selbststudium, wenn man keine Lehrkraft zur Verfügung hat. Gleichzeitig muss man aber auch kritisch bleiben: Nicht jede Antwort von einer KI ist korrekt, und es ist wichtig, Ergebnisse zu hinterfragen und mit anderen Quellen abzugleichen.

Ein weiteres Problem sehe ich im möglichen Verlust eigener Denkleistung, wenn man sich zu sehr auf KI verlässt. Es besteht die Gefahr, dass man Inhalte nur noch konsumiert, aber nicht mehr aktiv verarbeitet. Deshalb sollte KI aus meiner Sicht als Unterstützung verstanden werden, nicht als Ersatz für eigenes Lernen. Wenn man sie richtig einsetzt, kann sie aber definitiv ein wertvolles Werkzeug im Bildungsbereich sein.