
Legacy Migration Checklist v3.1

Info

Systemname: __

Team: __

Datum: __

Verwendung dieser Checkliste

Diese Checkliste ist für die Arbeit mit dem Legacy Migration Canvas während eines 4-8-stündigen Workshops konzipiert. Die Fragen sind priorisiert:

- **P1 (Sehr Wichtig):** Sollte im Workshop behandelt werden - wichtig zur Vervollständigung des Canvas.
- **P2 (Wichtig):** Sollte wenn möglich behandelt werden, ggfs. als Aufgaben für ein Follow-Up verteilen.
- **P3 (Ergänzend):** Weitere Fragestellungen, die eventuell wichtig sein könnten (kommt drauf an).

Hinweis: Mit P1-Fragen beginnen, um den Canvas möglichst vollständig abzudecken. P2/P3 behandeln wenn Zeit dafür ist oder um Aufgaben für deren Klärung zu verteilen.

Business Case & Value Proposition

Warum machen wir das & was gewinnen wir dabei?

- ☐ 1.1 **P1** Haben wir die Geschäftsziele für diese Migration klar definiert, die über technische Verbesserungen hinausgehen?
- ☐ 1.2 **P1** Haben wir spezifische "Pain Points" von Kunden/Nutzern identifiziert, die diese Migration lösen wird? Welche sind das?

- ☐ 1.3 **P1** Haben wir eine neutrale Bewertung der Stärken und Schwächen des aktuellen Systems durchgeführt, bevor wir einen Migrationsansatz definiert haben?
- ☐ 1.4 **P1** Verstehen wir, wie eine Migration die langfristige Unternehmensstrategie unterstützen könnte?
- ☐ 1.5 **P1** Haben wir neue Business-Capabilities (Kompetenzen und technisches Know-how) identifiziert, die durch die Migration geschaffen werden?
- ☐ 1.6 **P1** Haben wir uns mit den verschiedenen Stakeholdern über ihre spezifischen Migrationsziele abgestimmt (Umsatzsteigerung, Effizienz im Betrieb, Risikoreduktion)?
- ☐ 1.7 **P2** Haben wir Produktlinien oder Features identifiziert, die unrentabel sind und eingestellt statt migriert werden könnten?
- ☐ 1.8 **P2** Haben wir potenzielle Zeit- und Kosteneinsparungen für das Unternehmen und die Kunden quantifiziert?
- ☐ 1.9 **P2** Haben wir Metriken definiert, um den Nutzen für das Unternehmen vor und nach der Migration zu messen?
- ☐ 1.10 **P2** Haben wir strukturierte Frameworks wie OKRs in Betracht gezogen, um Migrationsziele mit messbaren Geschäftsergebnissen zu verknüpfen?



OKRs (Objectives and Key Results) verbinden abstrakte Geschäftsziele mit spezifischen, messbaren Schlüsselergebnissen, die während der Migration nachverfolgt werden können.

- ☐ 1.11 **P2** Haben wir die Unterstützung durch die Geschäftsleitung (Executive Sponsorship) durch die Ausrichtung an Geschäftszielen sichergestellt?

Hinweis: Fokus auf greifbaren Nutzen für das Unternehmen, nicht nur "technische Schulden reduzieren." Das Team soll die Perspektive von Product Ownern einnehmen und nicht nur an die technische Umsetzung denken.

Current System Analysis

Was ist unser Ausgangspunkt?

- ☐ 2.1 **P1** Haben wir die aktuelle Systemarchitektur und Abhängigkeiten gründlich analysiert und dokumentiert?

- ☐ 2.2 **P1** Verstehen wir die Datenmodelle und ihre fachliche Relevanz?
- ☐ 2.3 **P1** Wissen wir, welche existierenden Features noch genutzt werden und welche obsolet sind?
- ☐ 2.4 **P1** Haben wir Querschnittsbelange (Logging, Security, etc.) identifiziert, die neu gestaltet werden müssen?
- ☐ 2.5 **P2** Verstehen wir die Integrationspunkte mit anderen Systemen und deren Auswirkungen auf die Migration?
- ☐ 2.6 **P2** Haben wir die aktuellen Betriebsprozesse, Monitoring und Support-Prozesse für das Legacy-System analysiert?
- ☐ 2.7 **P2** Haben wir technische Schulden identifiziert, die während der Migration angegangen werden sollten?
- ☐ 2.8 **P2** Haben wir Sicherheitsrisiken sowohl im Legacy- als auch im Zielsystem bewertet?
- ☐ 2.9 **P2** Haben wir Datenflüsse bis zu den ursprünglichen Quellen zurückverfolgt, anstatt anzunehmen, dass das Legacy-System die "Source of Truth" ist?



Oft sind in den ursprünglichen Quellsystemen detailliertere oder qualitativ bessere Daten vorhanden als die, die momentan an das existierende System weitergegeben werden.

- ☐ 2.10 **P2** Haben wir kritische Aggregatoren identifiziert (Reporting-Funktionen, die für den Geschäftsbetrieb entscheidend sind)?



Oft existieren komplexe Reporting- oder Datenaggregationsprozesse, auf die die Geschäftsleitung angewiesen ist um das Unternehmen zu führen. Werden diese erst gegen Ende der Migration berücksichtigt können diese zu Show-Stoppn werden.

- ☐ 2.11 **P3** Haben wir analysiert, wie aktuelle Geschäftsprozesse durch die Einschränkungen des Legacy-Systems geprägt sind?
- ☐ 2.12 **P3** Haben wir potenzielle Event-Interception-Points (Messaging, APIs, Datenbanken) untersucht, um eine schrittweise Migration zu ermöglichen?



Stellen identifizieren, an denen Datenflüsse zwischen Systemen abgefangen werden können, um die Verarbeitung schrittweise auf neue Komponenten umzuleiten.

Hinweis: Nicht zu tief in technische Details im Workshop abtauchen. Fokus auf das große Ganze und darauf, was eine detailliertere Analyse erfordert.

Stakeholder Needs

Wer ist davon betroffen? Welche Anliegen haben die Betroffenen?

- ☐ 3.1 **P1** Haben wir alle relevanten Stakeholder-Gruppen identifiziert (Endnutzer, Fachbereiche, IT, usw.)?
- ☐ 3.2 **P1** Haben wir die Input von Endkunden zu ihren Bedürfnissen und "Pain Points" gesammelt?
- ☐ 3.3 **P1** Haben wir mit Sales- und Customer-Service-Teams gesprochen, um Kundenerwartungen zu verstehen?
- ☐ 3.4 **P1** Haben wir einen Kommunikationsplan, um Stakeholder während der gesamten Migration auf dem Laufenden zu halten?
- ☐ 3.5 **P1** Haben wir Feedback-Kanäle für kontinuierlichen Stakeholder-Input eingerichtet?
- ☐ 3.6 **P2** Haben wir Interviews oder Beobachtungssessions mit den Mitarbeitern durchgeführt, die direkten Kundenkontakt haben?
- ☐ 3.7 **P2** Haben wir Support-Tickets und häufige Nutzerbeschwerden analysiert?
- ☐ 3.8 **P2** Haben wir mit Finance-Teams gesprochen, um die Auswirkungen verschiedener Produkte/Features auf Umsatz und Gewinn zu verstehen?
- ☐ 3.9 **P2** Haben wir die Kommunikationshäufigkeit und wichtige Meilenstein-Updates für verschiedene Stakeholder-Gruppen während der Migration festgelegt?



Verschiedene Stakeholder brauchen unterschiedlich oft Updates - Führungskräfte benötigen eventuell monatliche Updates, während Endnutzer während der Migration häufiger informiert werden müssen.

- ☐ 3.10 **P2** Haben wir Eskalationsprotokolle für die Kommunikation von Migrationsproblemen oder Verzögerungen an Stakeholder festgelegt?



Definieren, wann und wie Probleme kommuniziert werden, wer zuerst informiert werden muss und welche Details verschiedene Gruppen benötigen.

- ☐ 3.11 **P3** Haben wir Workarounds (Spreadsheets, Access-Datenbanken, etc.) identifiziert, die sich um die Einschränkungen des Legacy-Systems herum entwickelt haben?
- ☐ 3.12 **P3** Haben wir Stakeholder dabei einbezogen, die Migration in sinnvolle Teilsegmente/"Slices" aufzuteilen?



Mit Fachexperten zusammenarbeiten, um das System in logische Segmente aufzuteilen, die unabhängig migriert werden können - mit Fokus auf fachlichen Wert statt auf technische Grenzen.

- ☐ 3.13 **P2** Verfügen wir über einen Plan, um Meilensteine der Migration zu feiern und Erfolge zu kommunizieren, um das Engagement der Stakeholder aufrechtzuerhalten?



Migrationsprojekte können langwierig sein – plant daher, Fortschritte anzuerkennen und die Dynamik durch regelmäßige Kommunikation über Erfolge aufrechtzuerhalten.

Hinweis: Technische Teams unterschätzen häufig die Komplexität der Einbindung von Stakeholdern. Sie sollten verstehen, dass „Nutzer“ viele verschiedene Gruppen mit unterschiedlichen Bedürfnissen umfassen.

Organizational Constraints

Reality check

- ☐ 4.1 **P1** Verfügen wir über die richtige Kombination aus technischen Fähigkeiten - sowohl für Legacy- als auch für Ziel-Technologien?
- ☐ 4.2 **P1** Haben wir Ressourcen für das Produktmanagement bereitgestellt, um die Migration zu begleiten und Ziele zu definieren?
- ☐ 4.3 **P1** Haben wir mögliche unvorhergesehene technische Herausforderungen im Budget berücksichtigt?
- ☐ 4.4 **P1** Haben wir Zugang zu Fachexperten für kritische Legacy-Komponenten?
- ☐ 4.5 **P2** Verfügen wir über Notfallpläne für Terminverlängerungen, falls erforderlich?
- ☐ 4.6 **P2** Haben wir die Gesamtbetriebskosten (Total Cost of Ownership) für das neue System korrekt eingeschätzt?

- ☐ 4.7 **P2** Haben wir den Bedarf an externem Fachwissen für spezielle Migrationsaufgaben berücksichtigt?
- ☐ 4.8 **P2** Haben wir die Kosten für den möglichen parallelen Betrieb kritischer Systeme während der Übergangsphase im Budget berücksichtigt?
- ☐ 4.9 **P3** Haben wir Ressourcen für die Implementierung und spätere Entfernung von Komponenten der Übergangsarchitektur bereitgestellt?
- ☐ 4.10 **P3** Haben wir den Zeitaufwand eingeplant, der für die Zusammenarbeit mit den Geschäftsbereichen zur Identifizierung von Migrationsabschnitten (migration slices) erforderlich ist?



Sicherstellen, dass der Zeitplan ausreichend Zeit für Workshops zur Analyse und Definition sinnvoller Migrationsschritte vorsieht.

Hinweis: Die Zeitplanung sollte realistisch sein. Technische Teams neigen dazu, optimistisch zu planen und die Komplexität der organisatorischen Koordination zu unterschätzen.

Quality Goals

Was wirklich zählt

- ☐ 5.1 **P1** Haben wir die 3-5 wichtigsten Qualitätsziele für diese Migration identifiziert?
- ☐ 5.2 **P1** Haben wir die Qualitätsziele auf die wichtigsten Anliegen der Stakeholder und die entscheidenden geschäftlichen Ziele abgestimmt?
- ☐ 5.3 **P1** Haben wir für jedes Qualitätsziel messbare Kriterien festgelegt?
- ☐ 5.4 **P1** Haben wir Qualitätsziele priorisiert, wenn sie miteinander in Konflikt stehen?
- ☐ 5.5 **P2** Haben wir definiert, wie wir den Erfolg der Migration messen werden?
- ☐ 5.6 **P2** Beabsichtigen wir, nach der Implementierung Nutzerfeedback einzuholen?
- ☐ 5.7 **P2** Haben wir einen Prozess für die Bearbeitung von Problemen und Verbesserungen nach dem (Neu-)Start eingerichtet?
- ☐ 5.8 **P2** Können wir überprüfen, ob die versprochenen Vorteile (z. B. schnellere Time-to-Market, geringere Änderungskosten, etc.) tatsächlich verwirklicht werden?
- ☐ 5.9 **P3** Haben wir Prozesse etabliert, um sicherzustellen, dass wir im neuen System keine

technischen Schulden ansammeln?

- ☐ 5.10 **P3** Verfügen wir über einen Plan zur Umsetzung der im Business Case möglicherweise zugesagten Continuous-Delivery-Praktiken?

Hinweis: Qualitätsziele sollten spezifisch und messbar sein, nicht allgemein wie „bessere Performance“. Das Team sollte dabei unterstützt werden, Qualitätsmerkmale mit tatsächlichen Geschäftsergebnissen zu verknüpfen. ---

Technical Constraints

Technische Sachen, die wir nicht/kaum ändern können

- ☐ 6.1 **P1** Haben wir Legacy-Schnittstellen identifiziert, die während und nach der Migration beibehalten werden müssen?
- ☐ 6.2 **P1** Verstehen wir die Anforderungen oder Einschränkungen unseres Technologie-Stacks?
- ☐ 6.3 **P1** Haben wir Integrationspunkte mit externen Systemen erfasst?
- ☐ 6.4 **P1** Kennen wir die Infrastruktur- und Deployment-Constraints, mit denen wir arbeiten müssen?
- ☐ 6.5 **P2** Haben wir geprüft, welche regulatorischen oder Compliance-Anforderungen unsere technischen Entscheidungen einschränken?
- ☐ 6.6 **P2** Verstehen wir die Anforderungen an Data Residency oder Data Sovereignty?



Wo Daten gespeichert werden müssen, wie sie verarbeitet werden können und welche Gesetze am geografischen Standort gelten.

- ☐ 6.7 **P2** Haben wir Vendor-Lock-ins identifiziert, die unsere Optionen einschränken?
- ☐ 6.8 **P2** Wissen wir, welche bestehenden Lizenzen oder Verträge unsere Technologie-Entscheidungen beeinflussen?
- ☐ 6.9 **P3** Haben wir die Auswirkungen der vorhandenen Monitoring- und operationellen Tools auf unsere Entscheidungen bewertet?
- ☐ 6.10 **P3** Kennen wir Netzwerk- und Sicherheitseinschränkungen, die sich auf die neue Architektur auswirken könnten?

Hinweis: Technische Constraints sind oft der Bereich, in dem sich technische Teams am

wohlsten fühlen, aber sie sollten sich nicht darin verlieren. Fokus auf Einschränkungen legen, die tatsächlich den Spielraum einschränken.

Migration Strategy

Wie gehen wir das an?

- ☐ 7.1 **P1** Haben wir einen iterativen/inkrementellen Ansatz entwickelt, der Risiken minimiert?
- ☐ 7.2 **P1** Haben wir geeignete Bereiche/Komponenten für eine frühzeitige Migration identifiziert?



Berücksichtigung sowohl anspruchsvoller/wichtiger Komponenten als auch "low-hanging fruits" mit klaren Kriterien dafür, was ernstzunehmende erste Erfolge sind (z. B. 10% des Umsatzes)

- ☐ 7.3 **P1** Haben wir eine Datenmigrationsstrategie, die die Geschäftskontinuität sicherstellt?
- ☐ 7.4 **P1** Haben wir Big-Bang-Ansätze explizit ausgeschlossen und setzen stattdessen auf schrittweise Migration?
- ☐ 7.5 **P1** Haben wir verschiedene Slicing-Ansätze untersucht?



Optionen wie Migration nach Produktlinie, Nutzergruppe, Geschäftsfähigkeit (business capability) oder Benutzererfahrung betrachten, um den effektivsten Ansatz zu finden.

- ☐ 7.6 **P1** Haben wir gegebenenfalls einen Parallelbetrieb während der Übergangsphase vorgesehen?
- ☐ 7.7 **P1** Haben wir in Betracht gezogen, kritische Aggregatoren (wichtige Reports/Funktionen) frühzeitig anzugehen, anstatt sie bis zum Schluss aufzuschieben? Welche?



Kritische Reports zuerst ersetzen, anstatt sie bis zum Ende aufzuschieben, wo sie die vollständige Migration blockieren können.

- ☐ 7.8 **P1** Haben wir die Feature-Parity-Falle vermieden, indem wir uns auf

Geschäftsanforderungen konzentrieren statt alle bestehenden Funktionen zu replizieren?



Das bestehende System einfach mit neuerer Technologie nachzubauen ist oft keine gute Idee.

- ☐ 7.9 **P1** Haben wir von Beginn der Migration an die Einführung von Continuous Delivery in Betracht gezogen, um künftige CI/CD-Fähigkeiten zu validieren?
- ☐ 7.10 **P2** Haben wir Rollback-Prozeduren für den Fall von Migrationsproblemen festgelegt?
- ☐ 7.11 **P2** Verfügen wir über einen Plan für die Wartung von Bestandssystemen während der Migration?
- ☐ 7.12 **P2** Haben wir eine Teststrategie zur Überprüfung der Funktionalität nach der Migration festgelegt?
- ☐ 7.13 **P2** Haben wir Performance- und Lasttests für jede Migrationsphase geplant, um sicherzustellen, dass das neue System die tatsächlichen Lasten im Produktivbetrieb bewältigen kann?



Tests sowohl für einzelne migrierte Komponenten als auch für die Gesamtsystemperformance einplanen, während sich die Last zwischen altem und neuem System verschiebt.

- ☐ 7.14 **P2** Verfügen wir über eine Strategie zur Prüfung der Datenintegrität und -konsistenz zwischen alten und neuen Systemen während des parallelen Betriebs?



Pläne für automatisierte Datenabgleichprüfungen und zur Behandlung von Abweichungen während der Umstellung berücksichtigen.

- ☐ 7.15 **P3** Haben wir die Anwendung des Strangler-Fig- und des Bridge-to-the-New-Town-Patterns in Erwägung gezogen, um Funktionalität schrittweise zu ersetzen?



Muster, bei denen neue Funktionen nach und nach den alten Code ablösen, indem Aufrufe abgefangen werden (Strangler Fig) oder durch Verbindungskomponenten während der Übergangszeit an das Bestandssystem durchgereicht werden (Bridge to the New Town). Bei der Entscheidung für ein Muster wie Strangler Fig sollte auch berücksichtigt werden, dass die Funktionsmigration zwar unkompliziert sein kann, die Datenmigration jedoch

deutlich komplexer sein kann.

- ☐ 7.16 **P3** Haben wir die erforderlichen Komponenten der Übergangsarchitektur mit klaren Plänen für deren spätere Entfernung entworfen?



Temporäre Komponenten, die während der Migration benötigt werden und entfernt werden sollten, sobald sie nicht mehr gebraucht werden.

Hinweis: Die Migrationsstrategie ist häufig Gegenstand intensiver Diskussionen. Im Workshop sollte der Schwerpunkt auf der Festlegung von Grundsätzen (inkrementell, wertorientiert) liegen und nicht auf detaillierten Ausführungsplänen.

Key Architectural Decisions

Kritische Entscheidungen, die die Migration prägen

- ☐ 8.1 **P1** Haben wir wichtige Entscheidungen zum Technologie-Stack für die Zielarchitektur getroffen?
- ☐ 8.2 **P1** Haben wir uns für ein Migrationsmuster entschieden (Strangler, Big Bang, Event Interception usw.)?
- ☐ 8.3 **P1** Haben wir einen Lösungsansatz für die Datenmigration und -synchronisierung festgelegt?
- ☐ 8.4 **P1** Haben wir entschieden, wie wir mit der Integration von Systemen umgehen, die nicht migriert werden?
- ☐ 8.5 **P1** Haben wir kurze Feedback-Loops für die frühzeitige Validierung von Migrationsansätzen eingerichtet?
- ☐ 8.6 **P1** Schaffen wir regelmäßig geschäftlichen Mehrwert durch inkrementelle Bereitstellung?
- ☐ 8.7 **P1** Verfügen wir über ein klares Ownership-Model für Komponenten während der Übergangsphase und nach der Migration?
- ☐ 8.8 **P2** Haben wir Muster oder Lösungsansätze für die Gewährleistung der Datenkonsistenz während der Migration bestimmt?
- ☐ 8.9 **P2** Haben wir uns für einen Test- und Validierungsansatz für Migrationsinkremente

entschieden?

- ☐ 8.10 **P2** Können wir häufige Releases anbieten, um unsere Fähigkeit zur schnellen Bereitstellung nach der Migration unter Beweis zu stellen?
- ☐ 8.11 **P2** Haben wir Testautomatisierung in Betracht gezogen, die schnelle und sichere Änderungen unterstützt?
- ☐ 8.12 **P2** Haben wir Architekturentscheidungen dokumentiert, um später darauf zurückgreifen zu können?
- ☐ 8.13 **P3** Haben wir Ansätze zum Umgang mit Übergangszuständen und für den gleichzeitigen Betrieb von zwei Systemen entwickelt?
- ☐ 8.14 **P3** Haben wir uns für Monitoring- und Observability-Ansätze für den Migrationsprozess entschieden?
- ☐ 8.15 **P3** Haben wir Ansätze für z. B. Feature-Flags oder das Konfigurationsmanagement während der Übergangsphase angedacht?
- ☐ 8.16 **P3** Entwickeln wir so, wie wir es auch nach der Migration tun wollen, d. h. mit denselben Praktiken und Vorgehensweisen, die wir nach der Migration beibehalten möchten?



Wenn das Ziel darin besteht, nach der Migration alle zwei Wochen ein Release zu veröffentlichen, sollten wir bereits während der Migration damit beginnen, alle zwei Wochen ein Release zu veröffentlichen.

- ☐ 8.17 **P3** Haben wir einen Plan, um eventuell verbleibende Legacy-Komponenten außer Betrieb zu nehmen?
- ☐ 8.18 **P3** Haben wir unseren Testansatz so konzipiert, dass wir überprüfen können, ob die migrierten Funktionen unter realen Bedingungen die gleiche Leistung wie die Legacy-Systeme erbringen?



Zusätzlich zu funktionalen Tests ist sicherzustellen, dass Leistung, Zuverlässigkeit und Benutzererfahrung den Fähigkeiten des Legacy-Systems entsprechen oder diese übertreffen.

- ☐ 8.19 **P3** Haben wir Ansätze für die parallele Leistungsmessung und -validierung (z. B. Tools wie GitHub Scientist) in Betracht gezogen, um das Verhalten des alten und des neuen Systems unter realen Bedingungen zu vergleichen?

Hinweis: Fokus auf Entscheidungen, die später nicht einfach geändert werden können. Nicht in Implementierungsdetails verzetteln, die während der Umsetzung verfeinert werden können.

Risks & Mitigation

Was könnte schiefgehen und wie gehen wir damit um?

- ☐ 9.1 **P1** Haben wir die wichtigsten technischen Risiken identifiziert, die die Migration beeinträchtigen könnten?
- ☐ 9.2 **P1** Haben wir Risiken für die Geschäftskontinuität (business continuity) während des Migrationsprozesses identifiziert?
- ☐ 9.3 **P1** Haben wir Risiken der Datenmigration und mögliche Verifikationsstrategien identifiziert?
- ☐ 9.4 **P1** Haben wir Notfallpläne für unsere kritischsten Abhängigkeiten erstellt?
- ☐ 9.5 **P1** Haben wir mögliche Vorbehalte und Widerstände bei Anwendergruppen identifiziert?
- ☐ 9.6 **P2** Verfügen wir über einen Plan, um Ängste und Bedenken hinsichtlich der Arbeitsplatzsicherheit oder von Rollenänderungen auszuräumen?
- ☐ 9.7 **P2** Haben wir den Schulungsbedarf für verschiedene Nutzergruppen berücksichtigt?
- ☐ 9.8 **P2** Verstehen wir, wie sich tägliche Arbeitsabläufe ändern werden und wie wir diesen Übergang unterstützen können?
- ☐ 9.9 **P2** Haben wir die Benutzer und Stakeholder auf mögliche vorübergehende Unterschiede in der Benutzererfahrung während der schrittweisen Migration vorbereitet? NOTE: Während einer schrittweisen Migration können Benutzer beim Wechsel zwischen alten und neuen Systemkomponenten unterschiedliche Benutzeroberflächen vorfinden.
- ☐ 9.10 **P2** Verfügen wir in jeder Geschäftseinheit über Fürsprecher, die die Veränderungen unterstützen und begleiten?
- ☐ 9.11 **P3** Haben wir die Vorteile des neuen Systems an alle betroffenen Stakeholdergruppen kommuniziert?
- ☐ 9.12 **P3** Haben wir berücksichtigt, wie wir die Arbeitsbelastung während der

Übergangsphase bewältigen können?

- ☐ 9.13 **P2** Haben wir schwergewichtige Veränderungsprozesse vermieden, die unseren zukünftigen Delivery-Zielen widersprechen?
- ☐ 9.14 **P3** Haben wir uns mit den organisatorischen Verhaltensweisen befasst, die überhaupt erst zu der bisherigen Situation geführt haben? NOTE: Welche Muster der Entscheidungsfindung oder der Organisationskultur haben zur existierenden Situation beigetragen und wie könnte man diese ändern?

Hinweis: Die Risikoidentifizierung deckt häufig Lücken in früheren Analysen auf. Wissen zu Risiken nutzen, um vorherige Abschnitte zu validieren und Bereiche zu identifizieren, die einer detaillierteren Planung bedürfen.

System Consolidation (falls zutreffend, nicht in Canvas)

Besonderheiten bei der Zusammenführung mehrerer Legacy-Systeme

- ☐ 10.1 **P1** Haben wir Anforderungen an die Funktionsgleichheit (feature parity) zwischen den zu konsolidierenden Systemen erfasst?
- ☐ 10.2 **P1** Verstehen wir die unterschiedlichen Benutzererfahrungen und -erwartungen für jedes System?
- ☐ 10.3 **P1** Haben wir potenzielle Konflikte in den Geschäftsprozessen zwischen den Systemen identifiziert?
- ☐ 10.4 **P1** Haben wir eine Strategie für den Datenabgleich zwischen unterschiedlichen Systemen?
- ☐ 10.5 **P2** Haben wir Entscheidungskriterien für die Konfliktlösung bei widersprüchlichen Anforderungen festgelegt?
- ☐ 10.6 **P2** Verstehen wir die organisatorischen Auswirkungen der Zusammenführung von Nutzergruppen?
- ☐ 10.7 **P2** Haben wir bestehende Unterschiede in der Verwendung von Fachbegriffen dokumentiert, um ein einheitliches Verständnis zu gewährleisten?
- ☐ 10.8 **P3** Haben wir die Annahme hinterfragt, dass alle Systeme konsolidiert werden müssen (anstatt einige stillzulegen)?

Hinweis: Die Konsolidierung von Systemen erhöht die Komplexität erheblich. Falls zutreffend, dafür sorgen, dass diese Fragen P1-Status erhalten, da sie grundlegende Auswirkungen auf die Migrationsstrategie haben.

Post-Migration Success (falls zutreffend, nicht in Canvas)

Sicherstellen, dass wir den versprochenen Mehrwert auch tatsächlich erbringen

- ☐ 11.1 **P1** Haben wir einen Wartungs- und Support-Plan für das neue System?
- ☐ 11.2 **P1** Haben wir Daten/Metriken, um den Erfolg der Migration zu messen?
- ☐ 11.3 **P1** Haben wir Pläne für den Wissenstransfer an die operativen Teams?
- ☐ 11.4 **P1** Haben wir Architekturentscheidungen dokumentiert?
- ☐ 11.5 **P2** Planen wir, nach der Implementierung Nutzer-Feedback zu sammeln?
- ☐ 11.6 **P2** Haben wir einen Prozess etabliert, um mit Problemen und Verbesserungsvorschläge nach der Einführung umzugehen?
- ☐ 11.7 **P2** Haben wir überprüft, ob die versprochenen Vorteile (z. B. schnellere Markteinführung, geringere Änderungskosten) tatsächlich eingetreten sind?
- ☐ 11.8 **P2** Haben wir die eventuell im Business Case versprochenen Continuous-Delivery-Praktiken umgesetzt?
- ☐ 11.9 **P2** Haben wir eine Retrospektive geplant, um gewonnene Erkenntnisse zu sammeln?
- ☐ 11.10 **P3** Haben wir Prozesse etabliert, um sicherzustellen, dass sich im neuen System keine technischen Schulden anhäufen?
- ☐ 11.11 **P3** Haben wir einen Plan, um eventuell verbleibende Legacy-Komponenten außer Betrieb zu nehmen?

Hinweis: Die Bewertung des Erfolgs nach der Migration kommt oft zu kurz. Die Technische Fertigstellung garantiert noch nicht den wirtschaftlichen Erfolg. Sicherstellen, dass tatsächliche Geschäftsergebnisse gemessen und überprüft werden.

Workshop-Zusammenfassung & Nächste Schritte

- **Wichtige Entscheidungen:**

-
- **Identifizierte kritische Risiken:**
 - **Unmittelbare Folgemaßnahmen:**
 - **Wichtige Folgemaßnahmen:**
 - **Zukünftige Überlegungen: ---**