**Allgemeine Prüfkriterien**

* Ist die Anforderung klar, eindeutig und verständlich formuliert?
* Gibt es widersprüchliche oder doppeldeutige Begriffe?
* Ist der Zweck/Fachkontext der Anforderung nachvollziehbar?
* Sind nicht-funktionale Anforderungen (z. B. Performance, Sicherheit) enthalten?

**Testbarkeit & Nachvollziehbarkeit**

* Ist die Anforderung **testbar** (z. B. über messbare Kriterien)?
* Gibt es definierte **Akzeptanzkriterien** oder erwartete Ergebnisse?
* Lässt sich die Anforderung in konkrete **Testfälle** umsetzen?
* Ist eine **Nachverfolgbarkeit zur User Story / ID** gegeben (z. B. REQ-123 → TC-001)?

**Aspekte für manuelles Testen**

* Sind **Grenzwerte / Edge Cases** spezifiziert?
* Gibt es Informationen zu **Benutzereingaben, Feldern, UI-Validierungen**?
* Wurden **Fehlermeldungen und Rückmeldungen** für ungültige Aktionen beschrieben?
* Ist das Verhalten bei **Fehlbedienung oder Ausfällen** dokumentiert?

**Aspekte für Testautomatisierung**

* Ist das Verhalten **stabil, deterministisch und wiederholbar**?
* Gibt es dynamische Elemente (z. B. Zufallsdaten, Captchas), die automatisiert schwer testbar sind?
* Ist der **Zugriff auf UI/DOM/API** automatisierbar (IDs, XPath, API-Dokumentation vorhanden)?
* Können Testdaten automatisiert vorbereitet oder bereinigt werden?
* Wurde das gewünschte Verhalten für **sad paths und happy paths** dokumentiert?

**Technische & Integrationsaspekte**

* Sind **System- oder Schnittstellenabhängigkeiten** benannt (z. B. APIs, Datenbanken, externe Module)?
* Gibt es Einschränkungen durch Umgebungen, Rollen oder Rechte?
* Ist der **Testaufwand** durch Komplexität, Abhängigkeiten oder Testumgebungen realistisch planbar?

**Qualität & Rückverfolgbarkeit**

* Wurden frühere Fehlerquellen oder Lessons Learned berücksichtigt?
* Gibt es Versionierung oder Änderungsverfolgung der Anforderung?
* Sind Referenzen zu UI-Mockups, Business Rules, Protokollen vorhanden?

**General Criteria**

* Is the requirement clearly and unambiguously written?
* Are there any vague, conflicting, or undefined terms?
* Is the purpose and business context understandable?
* Are non-functional aspects mentioned (e.g. performance, security, usability)?

**Testability & Traceability**

* Is the requirement **testable** (can it be verified objectively)?
* Are **acceptance criteria or expected outcomes** defined?
* Can the requirement be broken down into concrete **test cases**?
* Is there a **traceable link** to the related User Story, ID or specification?

**Manual Testing Considerations**

* Are **boundary values and edge cases** described?
* Are **input fields, user actions, and validations** clearly defined?
* Are **error messages and feedback behavior** covered?
* Is the behavior in case of **invalid input or system failure** documented?

**Automation Testing Considerations**

* Is the behavior **stable, deterministic, and repeatable**?
* Are there elements that might block automation (e.g. captchas, dynamic IDs)?
* Is there access to identifiable **UI elements / DOM / APIs** for automation?
* Can **test data setup and teardown** be automated?
* Are both **happy paths und sad paths** well defined?

**Technical & Integration Aspects**

* Are **dependencies on other systems, modules or APIs** mentioned?
* Are there **role-based behaviors or permission constraints**?
* Is the **testing effort realistically plannable** given the complexity or integration?

**Quality & Maintainability**

* Have prior defects or lessons learned been considered?
* Is **version control or change tracking** available for this requirement?
* Are supporting materials (e.g. UI mockups, business rules, workflows) referenced?