



# Software- Entwicklung

*Sylphen*  
Ihr IT-Systemhaus

## Wie wir arbeiten

Agil, flexibel, nachhaltig –  
maßgeschneiderte Lösungen  
für Ihr Unternehmen

Mit Sylphen an Ihrer Seite ist Ihre Software-Entwicklung nicht nur ein Projekt, sondern eine Partnerschaft. Wir setzen auf Agilität, Flexibilität und höchste Qualität, um Ihre IT-Ziele schnell und effizient zu erreichen.

Ein agiles und flexibles Vorgehen bei der gemeinsamen Entwicklung darf kein Chaos verursachen. Dank ihrer langjährigen Erfahrung wissen unsere Entwickler, wie Lösungen schnell und zielgerichtet wachsen können.

Wenn unser Scrum-Team mit Ihnen als Product Owner nahtlos zusammenarbeitet, entstehen Ergebnisse, die Ihnen schnell Vorteile bringen und sich rechnen.

## Unser Ansatz

Wir setzen auf Scrum, um schnell und flexibel auf Ihre Anforderungen zu reagieren.



### Product Owner

- Anforderungen
- Priorisierung
- Release Termine
- Feedback berücksichtigen
- Stakeholder Management
- Abnahme

Wird vom Kunden gestellt und ist bei den Projekt-Meetings immer dabei, priorisiert die Anforderungen und berichtet seinem Management über die Entwicklung. Er besitzt die Entscheidungsgewalt für Entwicklungsfragen.

## Agiles Projektmanagement mit Scrum

Warum? Weil es einfach funktioniert.

Scrum ist unsere bevorzugte Methode, um Projekte schnell, flexibel und zielgerichtet voranzutreiben. Wir liefern kontinuierlich Ergebnisse, passen uns an neue Anforderungen an und bieten Ihnen stets transparente Einblicke.



### Scrum Master

- Team-Unterstützung
- Prozess-Unterstützung
- Impediment Management
- Kontinuierliche Verbesserung
- Scrum Trainer

Er ist der Moderator und sorgt dafür, dass alle Termine eingehalten werden, dass das Team effizient arbeiten kann und räumt Schwierigkeiten aus dem Weg.



### Entwicklungsteam

- Arbeitsaufgaben definieren
- Qualität sicherstellen
- Aufwände schätzen
- Produkt kollaborativ entwickeln

Es erstellt die Software und schätzt Aufwände.

### Warum Scrum?

- **Flexibilität:**  
Anpassung an Ihre sich ändernden Anforderungen
- **Frühe Lieferungen:**  
Regelmäßige Ergebnisse und schnelle Feedbackschleifen
- **Transparente Kommunikation:**  
Sie sind jederzeit auf dem Laufenden
- **Hohe Qualität:**  
Regelmäßige Tests sichern die Qualität der Software
- **Effizienzsteigerung:**  
Minimierung von Verschwendungen und Optimierung der Ressourcen

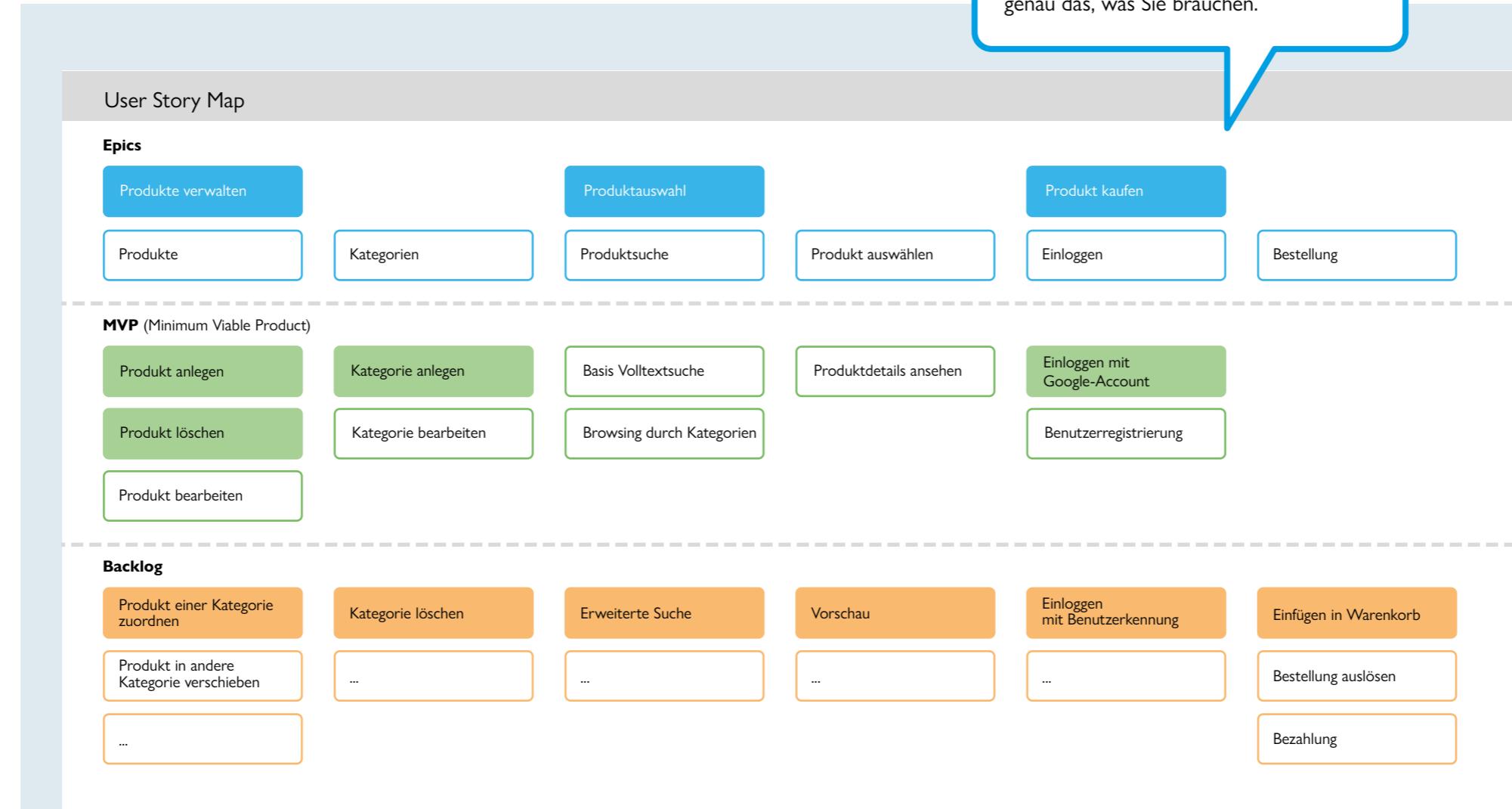
# User Story Map

## Ihre Anforderungen visualisiert

**Die User Story Map** ist unser Schlüssel zu einer strukturierten und effizienten Software-Entwicklung. Sie hilft uns, alle Anforderungen aus Sicht des Benutzers zu ordnen und sicherzustellen, dass wir immer den richtigen Fokus setzen. Durch die Visualisierung Ihrer Bedürfnisse behalten wir gemeinsam den Überblick und der Kundenfokus bleibt stets erhalten.

**In dieser Ansicht** beschreiben wir die Tätigkeiten, die Anwender in den zu entwickelnden Software ausführen sollen. Diese Map bietet jederzeit eine klare Übersicht über den Fortschritt und die Priorisierung der Funktionen. Sie kann auch während der Entwicklungsphase genutzt werden, um den Entwicklungsfortschritt zu visualisieren und Priorisierungen vorzunehmen.

Nach der Erstellung der User Story Map lässt sich zudem ein realistisches **Budget** für das Projekt ermitteln.



## **Volle Einbindung, volle Kontrolle.**

Sie haben jederzeit Einblick in den Entwicklungsstand und können aktiv Feedback geben. So bleiben wir flexibel und liefern genau das, was Sie brauchen.

# Wie funktioniert die User Story Map?

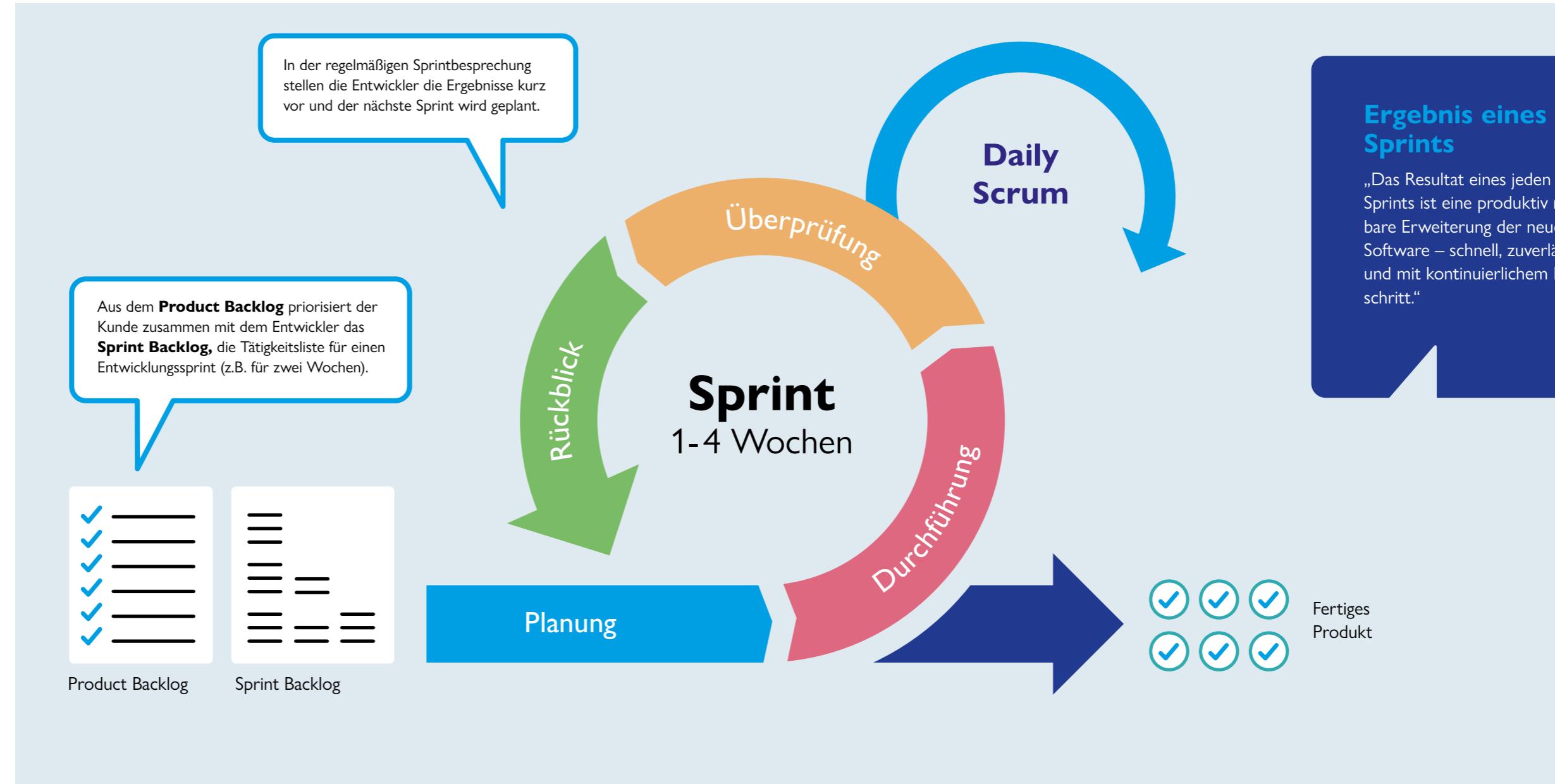
- **Releases** fassen User Stories zusammen und zeigen, wann sie Ihnen ausgeliefert werden. Sie werden hier von oben nach unten in Abschnitten dargestellt. Das **MVP** ist das erste Release.
  - **Epics und User Stories:** Die großen und kleinen Funktionen, die wir priorisieren, um Ihre Anforderungen schnell zu erfüllen.

# Der Entwicklungs-Sprint

# Agil statt dogmatisch – schneller zum Mehrwert

Nachdem die User Story Map erstellt wurde, füllen die Aufgaben aus der Map das Product Backlog. Zusammen mit Ihnen priorisieren wir die Aufgaben und legen sie in das Sprint Backlog für den kommenden Entwicklungs-Sprint (z.B. für zwei Wochen). In diesem Sprint setzen wir die Anforderungen um und lassen diese von Ihnen abnehmen.

- **Sprint Backlog:** Eine Liste der Aufgaben, die im nächsten Sprint bearbeitet werden.
  - **Sprint Dauer:** In der Regel dauert ein Sprint zwei Wochen, in denen wir die priorisierten Anforderungen umsetzen.
  - **Sprintbesprechung:** Am Ende eines jeden Sprints stellen wir Ihnen die Ergebnisse vor und planen gemeinsam den nächsten Sprint. Hier besprechen wir, was funktioniert hat, welche Anpassungen nötig sind und was der nächste Schritt sein wird.

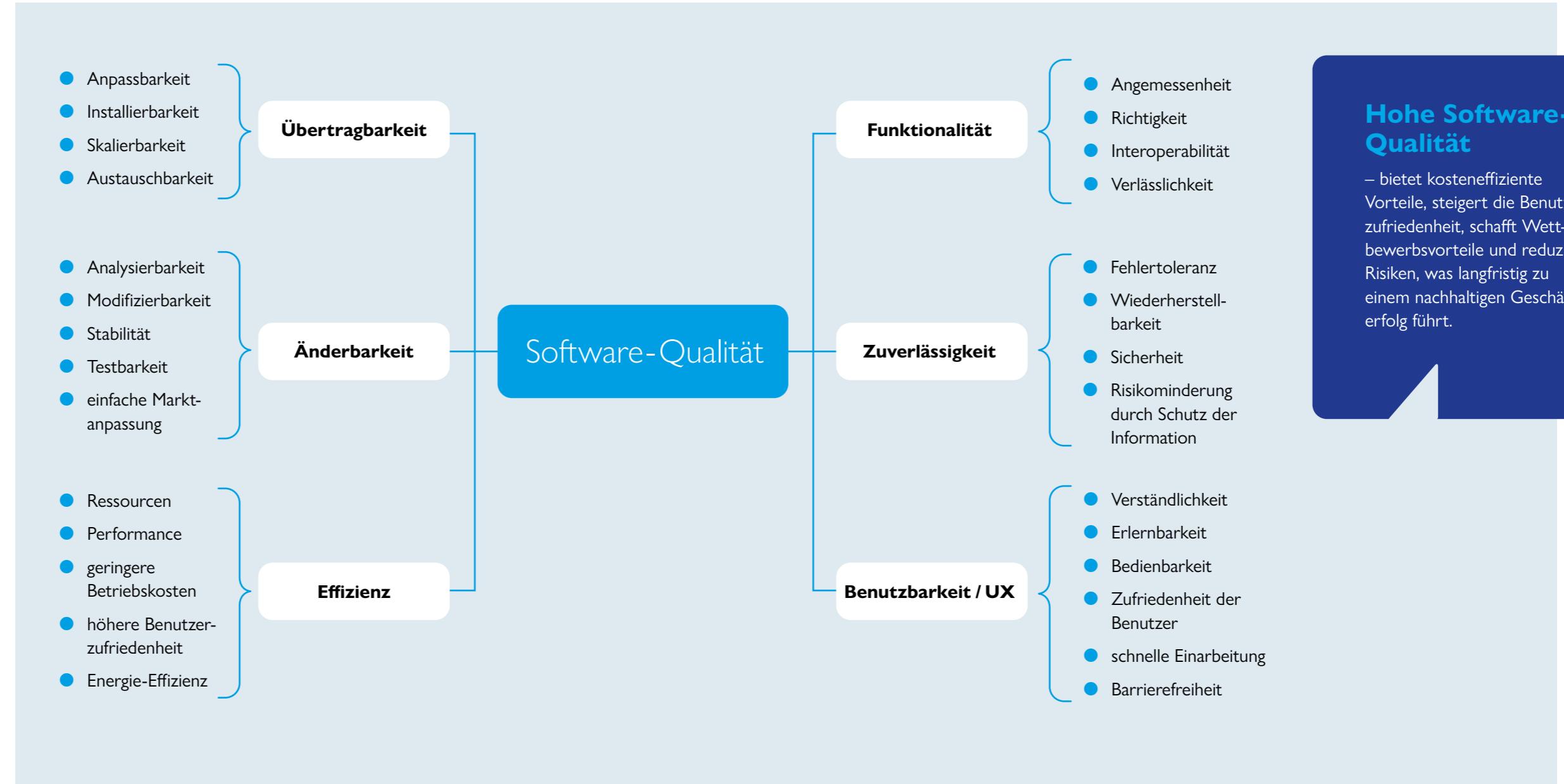


# Merkmale der Software-Qualität

– für einen nachhaltigen Invest

Mit folgenden Maßnahmen erreichen wir für Sie als Kunde eine hohe Software-Qualität:

- **Qualitäts-Richtlinien**, an die sich Entwickler halten müssen, inkl. Abnahmekriterien
- Sinnvolle Entwickler- und **Benutzer-Dokumentation**
- **Tests-driven** development
- **Automatisierte Code-analyse und Überwachung** von Qualitätskriterien
- **Kontinuierlicher Verbesserungsprozess**
- Regelmäßige **Weiterbildung** der eigenen Mitarbeiter
- Frühe Einbindung von **UX-Designern**, besonders bei zentralen Masken
- **ISTQB-zertifizierte Tester**



# Technologien für moderne und nachhaltige Architekturen

## Software-Entwicklung und Frameworks

- **Programmiersprachen:** Java, Kotlin, Python, PHP, PL/SQL, etc.
- **Frameworks:** Spring, Java EE, Symfony, Django, Vaadin, etc.
- **Frontend-Entwicklung:** React, Angular, RCP, React Native, ionic, etc.

## Content-Management und Plattformen

- **CMS:** Typo3, WordPress, Pimcore, etc.
- **Integration:** n8n, Apache Nifi, Talend, etc.

## Qualitätssicherung, CI/CD & Testing

- **Testing-Tools:** JUnit, SonarQube, dependency track, etc.
- **Code-Management und CI/CD:** GitLab, Jenkins, Jira, Redmine, etc.

Weitere Tools und Technologien, die auf Anfrage oder projektspezifisch eingesetzt werden, z. B. Cloud-Dienste (AWS, Azure), Datenbanklösungen (Oracle, MySQL, PostgreSQL), oder Microservices-Architekturen.

[www.sylphen.com/software-entwicklung](http://www.sylphen.com/software-entwicklung)