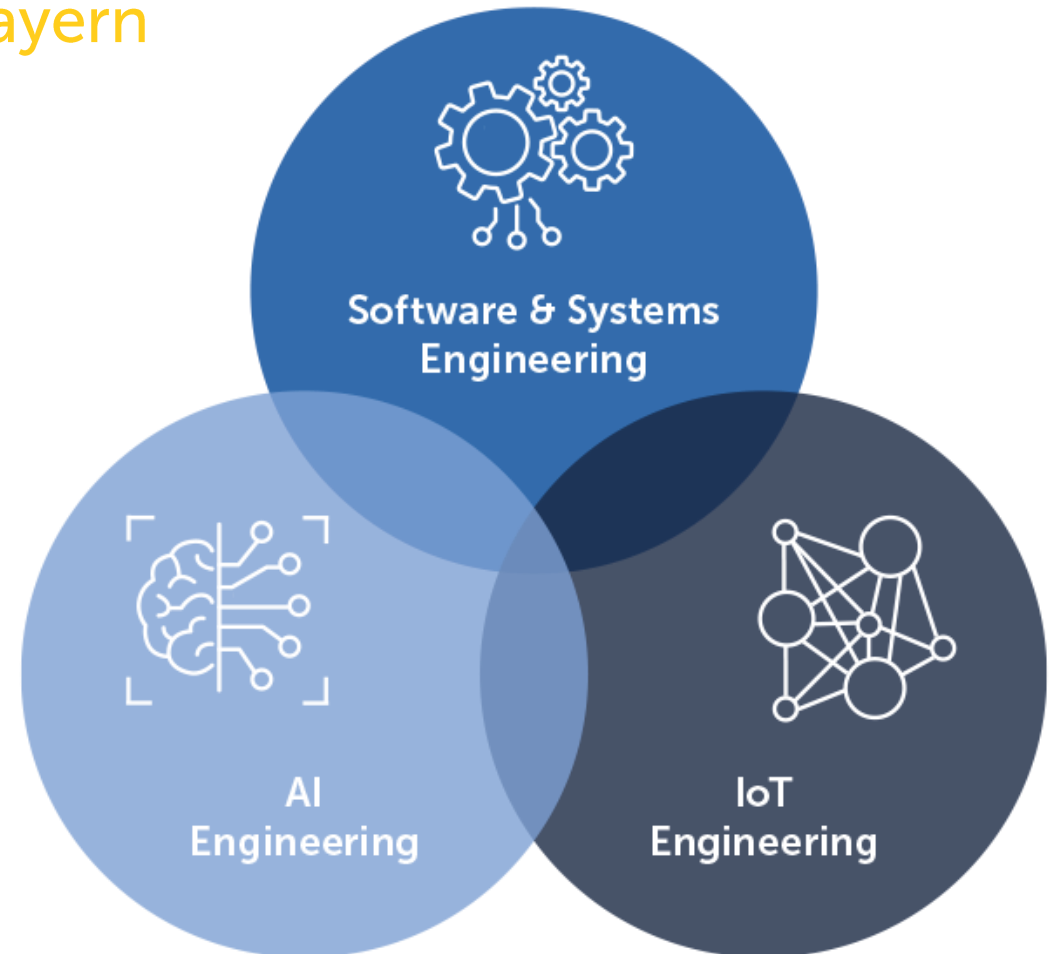


fortiss

Landesforschungsinstitut des Freistaats Bayern für softwareintensive Systeme

- **Rechtsform:** gemeinnützige GmbH
- **Gründung:** 2009
- **Eigentümer:** Freistaat Bayern (2/3) und Fraunhofer-Gesellschaft (1/3)
- fortiss ist ein **An-Institut der TU München**

fortiss bearbeitet auf Spitzenniveau das für den Hightech-Standort Bayern zentrale Thema der Entwicklung softwareintensiver Systeme. Diese basieren zunehmend auch auf den eng verwandten Technologien der KI.



Ihr Ansprechpartner

Dr. Jannik Fischbach

Postdoctoral Researcher

fortiss GmbH

Guerickestr. 25

80805 München

fischbach@fortiss.org



Kompetenzfeld Requirements Engineering

Unsere typischen Aktivitäten für Requirements-Engineering-Ansätze und -Werkzeuge

Forschung



Problemorientierte, empirisch fundierte Forschung und Benchmarks



Veröffentlichung offener Datensätze für evidenzbasierte Forschung

Anwendung



Artefaktbasierte RE-Ansätze als Engineering-Leitlinien



Tool-Unterstützung für automatisierte Compliance-Prüfungen

Wirkung



Wissenstransfer-Workshops und Konferenzen zu ausgewählten Domänen und Themen



Verschiedene Aus- und Weiterbildungsformate (für Fachleute und Hochschulbildung)

Agenda

1. Kritikalität und Grundlagen des Requirements Engineering
2. Dokumentation von Anforderungen
3. Einführung in effektives Prompt Engineering
4. Offene Fragerunde

Relevanz des Requirements Engineering (RE)

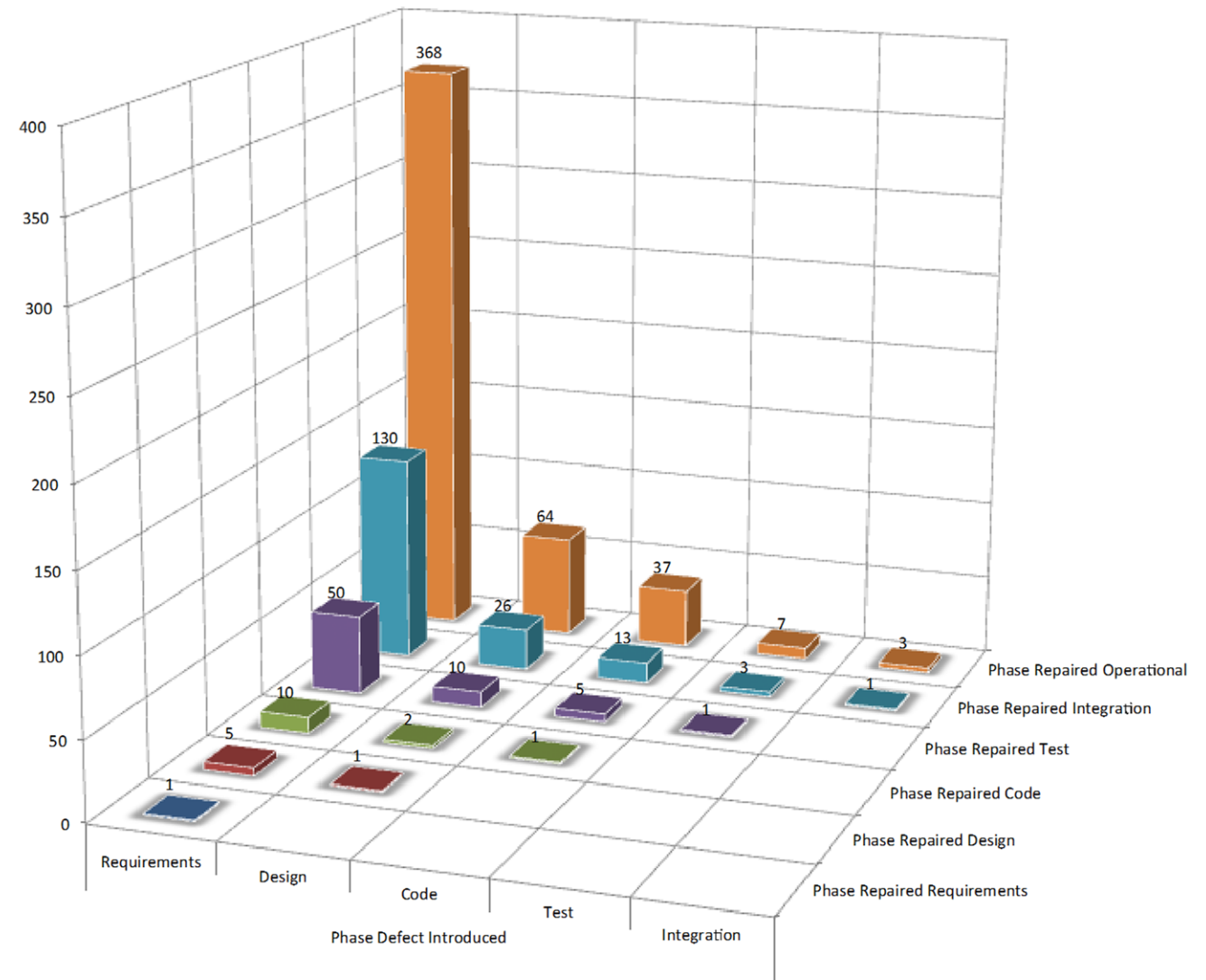
Für ein Softwaresystem beeinflusst das RE (und die Ergebnisse des RE-Prozesses)

- ▶ Die Funktionalität und Qualität des Systems
- ▶ Die Kosten des Systems
- ▶ Der Nutzen des Systems
- ▶ Die Komplexität des Systems

Aber: RE wird häufig unterschätzt und nicht ernst genommen. Oft beherrschen Ingenieure den RE-Prozess nicht einmal ausreichend...

RE als Erfolgsfaktor

- ▶ Je früher ein Fehler eingebracht wird (z. B. im RE) und je später er entdeckt wird, desto teurer ist seine Behebung
- ▶ **Faustregel:**
Faktor 10 pro Stufe



Quelle: Bennet/Wennberg (NASA-Projektanalyse)

Siren police I
a 'debacle'

BBC abandons £100m digital project

James Purnell, BBC's director of strategy and digital: "The BBC has scrapped a £99 million project to launch a new digital service. The director general said it was "not worth the money".

government to downsize
Canada.ca project

London Stock Exchange Suffers 4 Hour Outage

Traders disgusted - again

US Army's Future Combat Systems Program Formally Terminated

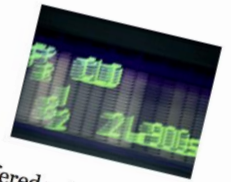
30 Million German Bank Cards Almost Fixed

2010 Date Problem Still Exists For Cards Used Outside of Germany

Robert Charette



As I mentioned a few days ago, some 2010 date problems hit various electronic systems and devices in different parts of the world on New Years Day. One Risk Factor reader noted that Germany likely experienced the greatest date-related problem because software in a security microchip used in 30 million German bank cards was unable to recognize the date 2010.



Exchange suffered yet another technical problem today. This time trading was suspended for some 4 hours. The market then reopened at 0800 London time. The market then reopened at 0800 London time. The market then reopened at 0800 London time. The market then reopened at 0800 London time.

British Airways) London continued into today. For occurring, BA the day began. It also with hand luggage. I

[1] http://www.dw.com/en/germany-seek

[2] Quick and simple Google search

33%

... der Fehler treten im RE auf.

36%

... der Fehler, die im RE auftreten, führen
zum Projektmisserfolg.

Definition: Requirements Engineering (RE)

Requirements Engineering (RE) ist der systematische, iterative und disziplinierte Ansatz zur Entwicklung einer expliziten Anforderungs- und Systemspezifikation, der von allen Stakeholdern akzeptiert wird.

Grundlegende Aufgaben umfassen unter anderem:

- ▶ Anforderungserhebung (Identifikation aller relevanten Anforderungen)
- ▶ Anforderungsanalyse (Anforderungen verstehen und Konsens herstellen)
- ▶ Anforderungsspezifikation (Strukturierung, Modellierung und Dokumentation von Anforderungen)
- ▶ Anforderungvalidierung und -verifikation (Sicherstellung von Gültigkeit und Qualität)

Basis: M. Glinz. A Glossary of Requirements Engineering Terminology, v1.6, IREB, 2014

RE agiert im „Problemraum“

Requirements Engineering zielt auf den **Problemraum**...

- ▶ die Problemstellung für eine Entwicklungsaufgabe möglichst umfassend und präzise zu fassen: Warum ist etwas notwendig?
- ▶ geeignete Anforderungen an die gewünschte Lösung abzuleiten und Konsens unter Stakeholdern zu erzielen: Was (Fähigkeiten, Eigenschaften) ist notwendig?

Design und Implementierung zielen hingegen auf den **Lösungsraum**,
d. h. Entwicklung und Weiterentwicklung möglicher Lösungen: Wie wird die Lösung* realisiert?

Die Unterscheidung von Problemraum und Lösungsraum ist essenziell!

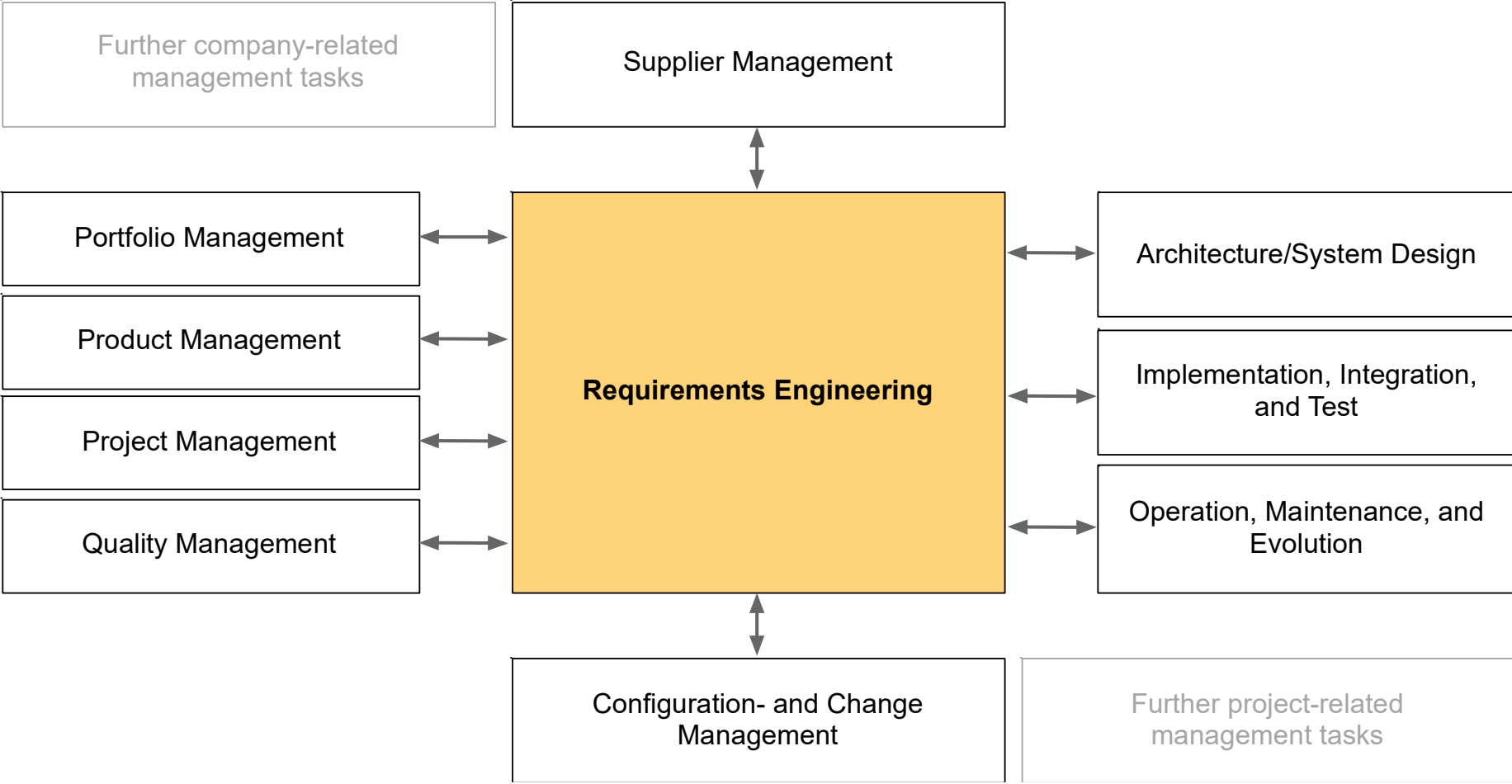
Definition: Requirements Management (RM)

Requirements Management (RM) zielt darauf ab, Anforderungen über den gesamten Systemlebenszyklus hinweg effizient und effektiv zu verwalten und zu nutzen.

Grundlegende Aufgaben umfassen unter anderem:

- ▶ Archivierung und Basislinienbildung von Anforderungen
- ▶ Änderung von Anforderungen aufgrund neuer Erkenntnisse
- ▶ Nachverfolgung und Verifizierung von Anforderungen (Auswirkungsanalyse, Änderungsmanagement)

RE und RM bilden eine Schlüsselschnittstelle zu zahlreichen Aktivitäten im Entwicklungslebenszyklus



Agenda

1. Kritikalität und Grundlagen des Requirements Engineering
2. Dokumentation von Anforderungen
3. Einführung in effektives Prompt Engineering
4. Offene Fragerunde

Eine erste Kategorisierung von Anforderungen

Funktionale Anforderungen

- ▶ Verhalten eines Systems (allgemein)
- ▶ Verhalten aus der Perspektive der Nutzenden

Qualitätsanforderungen

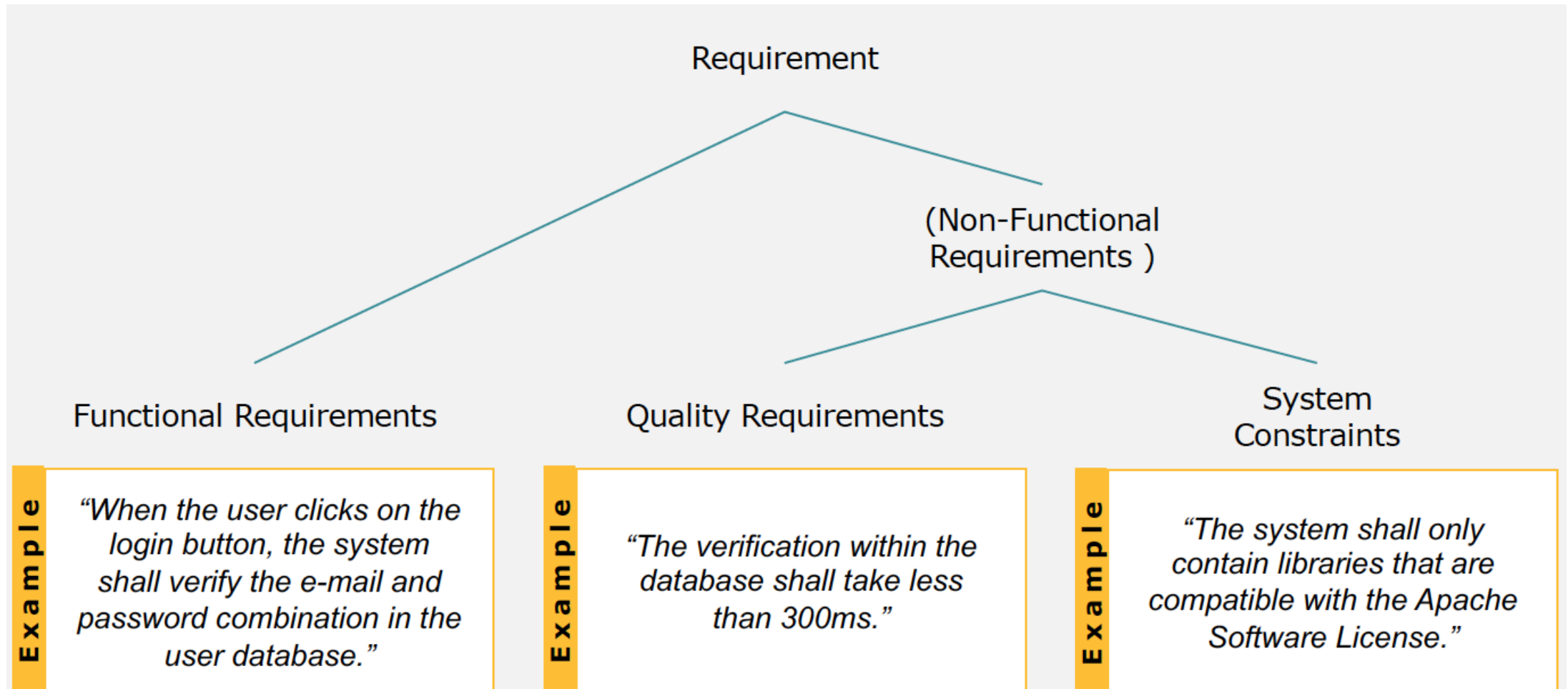
- ▶ Qualitätsmerkmale und Eigenschaften eines Systems
- ▶ (Idealerweise) quantifizierte Eigenschaften des Verhaltens

Oft bezeichnet als
„Nicht-funktionale
Anforderungen“

Prozessanforderungen (projektspezifisch)

- ▶ Anforderungen an den Entwicklungsprozess (Zeitplanung, Meilensteine, Budget usw.)
- ▶ Vorgaben/Regeln für die Umsetzung (Einschränkungen, wiederzuverwendende Komponenten usw.)

Eine erste Kategorisierung von Anforderungen

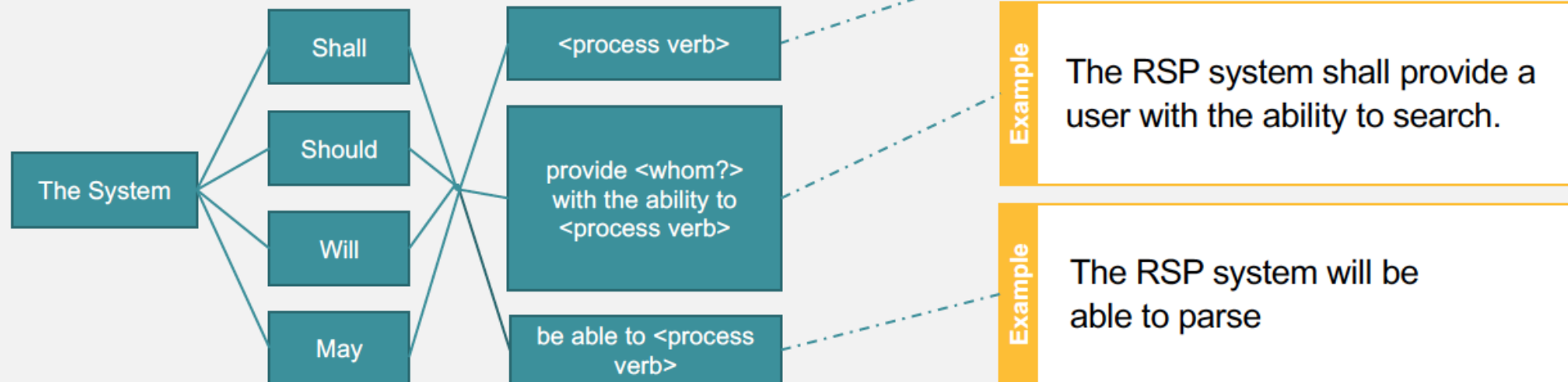


Techniken zur Anforderungsdokumentation

Technik	Vorteile	Nachteile
Natürliche Sprache	<ul style="list-style-type: none">• Kann jede Art von Anforderungsperspektive ausdrücken• Keine Schulung erforderlich	<ul style="list-style-type: none">• Erhöht die Wahrscheinlichkeit mehrdeutiger Anforderungen• Erhöht das Risiko, Anforderungstypen und -perspektiven zu vermischen• Überblick manchmal schwer zu behalten
Modelle	<ul style="list-style-type: none">• Für geschulte Nutzende leicht verständlich• Reduziert den Grad der Mehrdeutigkeit• Bietet verschiedene Abstraktionsebenen je nach Zweck	<ul style="list-style-type: none">• Schulung erforderlich• Ein Modell kann nicht alle Anforderungsperspektiven abdecken• Schwieriger nachzuverfolgen
Kombination aus natürlicher Sprache und Modellen	<ul style="list-style-type: none">• Nachteile beider Methoden werden durch die jeweiligen Vorteile ausgeglichen (Modelle ergänzen natürliche Sprache und umgekehrt)	<ul style="list-style-type: none">• Erhöht das Risiko, Anforderungstypen und -perspektiven zu vermischen• Risiko von Inkonsistenzen

Techniken zur Anforderungsdokumentation

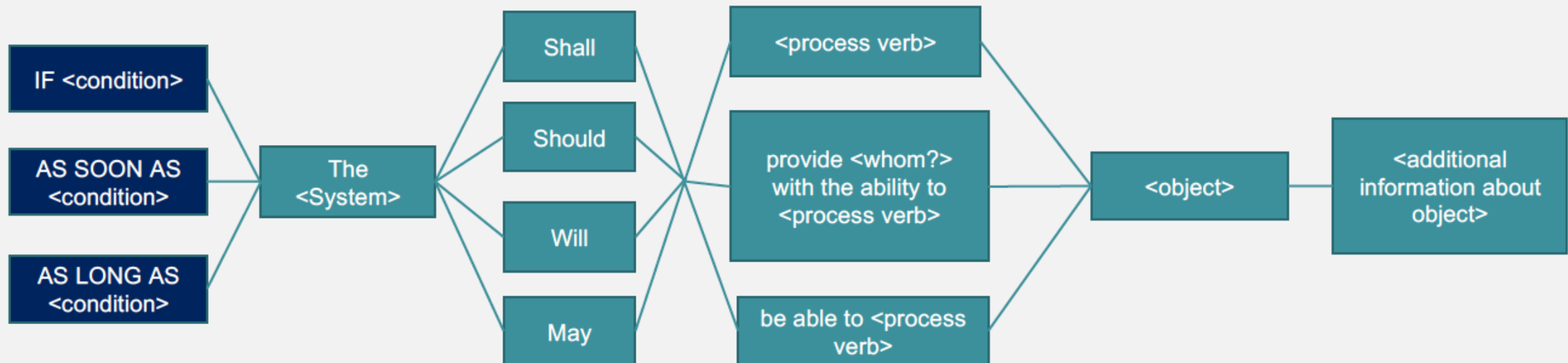
- System Activity Types:
 - Autonomous System Activity
 - User Interaction
 - Interface
- Inform the requirement template type



Techniken zur Anforderungsdokumentation

Constraints under which processes take place

- Logical conditions use logical conjunction 'if'
- Temporal conditions use Temporal conjunction 'As soon as' (Event) or 'As long as' (Time Period)
- Why NOT 'When' ? (-> to prevent ambiguity)



SOFTWARE ENGINEERING TOOLBOX



Möchten Sie mehr über Requirements Engineering und Anforderungen erfahren?

In der **Software-Engineering Toolbox** stehen Ihnen Lernkomponenten und zur Verfügung – kostenlos und ohne Anmeldung.

<https://se-toolbox.info/>

Mittelstand-Digital

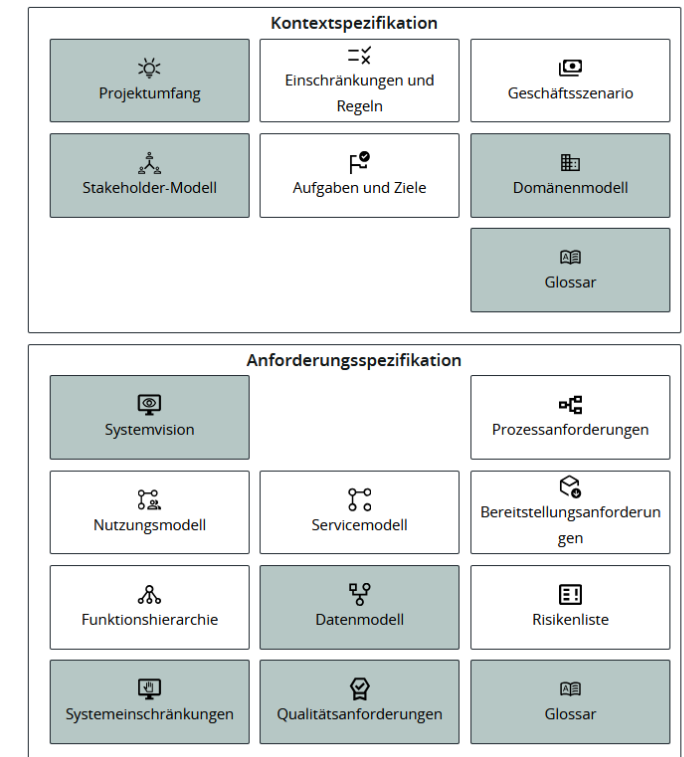


Gefördert durch:



Bundesministerium
für Wirtschaft
und Klimaschutz

aufgrund eines Beschlusses
des Deutschen Bundestages



Agenda

1. Kritikalität und Grundlagen des Requirements Engineering
2. Dokumentation von Anforderungen
3. Einführung in effektives Prompt Engineering
4. Offene Fragerunde

„Prompt Engineering ist die Kunst, mit einem generativen großen Sprachmodell zu kommunizieren.“

Prompt Engineering – Grundlagen (1/2)

Was sind LLMs?

- Transformer-basierte neuronale Netzwerke
- Trainiert auf enormen Textmengen
- Kontext-Verständnis und kohärente Antworten

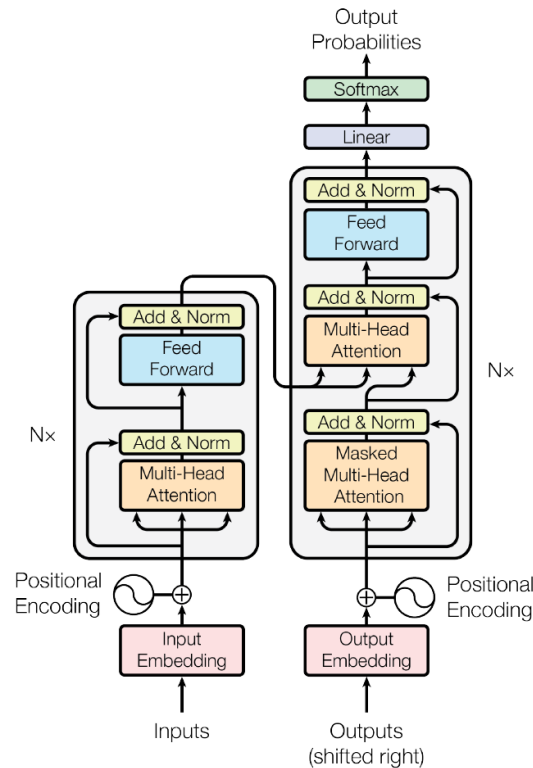
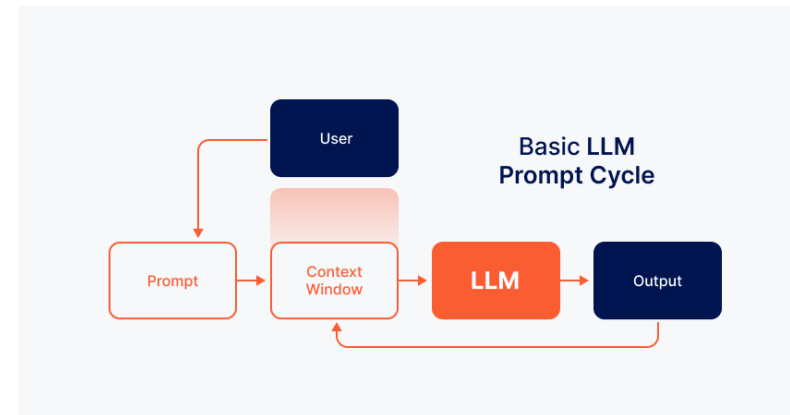


Figure 1: The Transformer - model architecture.

Warum Prompts?

- Brücke zwischen menschlicher Intention und KI-Verständnis
- Textuelle Eingabe zur Kommunikation mit LLMs
- Bestimmt Qualität und Relevanz der Ausgabe



Prompt Engineering – Grundlagen (2/2)

Was ist Prompt-Engineering?

= die Disziplin der Gestaltung und Optimierung von Eingabeaufforderungen (Prompts), um Sprachmodelle (LLMs) effizient für eine Vielzahl von Aufgaben zu nutzen.

Ziel: Verbesserung der Modellleistung bei Aufgaben wie Fragebeantwortung, Textgenerierung und Codegenerierung.

Warum ist es wichtig?

- Verbessert die Genauigkeit und Relevanz der Outputs
- Ermöglicht die Steuerung des Modellverhaltens
- Erschließt das volle Potenzial von LLMs

Grundlagen des Prompts

Zwei Hauptansätze

Zero-Shot Prompting

Das Modell erhält eine Anweisung **ohne jegliche Beispiele**.

Beispiel-Prompt:
Übersetze den folgenden Satz ins Englische: „Ich liebe Prompt Engineering.“

-> **Geeignet für einfache und direkte Aufgaben**

Few-Shot Prompting

Dem Modell werden einige Beispiele (Shots) zur Verfügung gestellt, um den Kontext und das gewünschte Ausgabeformat zu demonstrieren.

Beispiel-Prompt:
"Deutsch: 'Hallo', Englisch: 'Hello'.
Deutsch: 'Wie geht es Ihnen?', Englisch: 'How are you?'.
Deutsch: 'Was ist ein Apfel?', Englisch: „

-> **Verbessert die Leistung bei komplexeren oder spezifischeren Aufgaben**

Chain-of-Thought Prompting

Standard Prompting	Chain-of-Thought Prompting
<p data-bbox="570 482 759 515">Model Input</p> <p data-bbox="529 546 1248 644">Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?</p> <p data-bbox="529 689 810 718">A: The answer is 11.</p> <p data-bbox="529 763 1212 861">Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?</p>	<p data-bbox="1378 482 1567 515">Model Input</p> <p data-bbox="1337 546 2056 644">Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?</p> <p data-bbox="1337 689 2056 758">A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.</p> <p data-bbox="1337 803 2020 901">Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?</p>
<p data-bbox="570 982 759 1015">Model Output</p> <p data-bbox="529 1046 886 1086">A: The answer is 27. ❌</p>	<p data-bbox="1378 982 1567 1015">Model Output</p> <p data-bbox="1337 1032 2056 1172">A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅</p>

Komponenten eines effektiven Prompts

Komponenten für maximale Modelleistung

Anweisung: klare Aufgabe oder Auftrag.

Kontext: Hintergrundinformationen zur Steuerung der Antwort.

Eingabedaten: das konkrete Objekt oder die Frage.

Ausgabeindikator: Format oder Stil der Antwort.

Generelle Tipps:

- **Start Simple:** Iterative Verbesserung durch Experimentieren
- **Klare Instruktionen:** Präzise Befehle mit strukturierter Formatierung
- **Spezifisch sein:** Je detaillierter, desto besser die Ergebnisse
- **Ungenauigkeiten vermeiden:** Direkte und konkrete Kommunikation
- **Positiv formulieren:** Sagen was zu tun ist, nicht was zu vermeiden ist

Optimierung der Interaktion mit KI-Modellen durch effektives Prompting

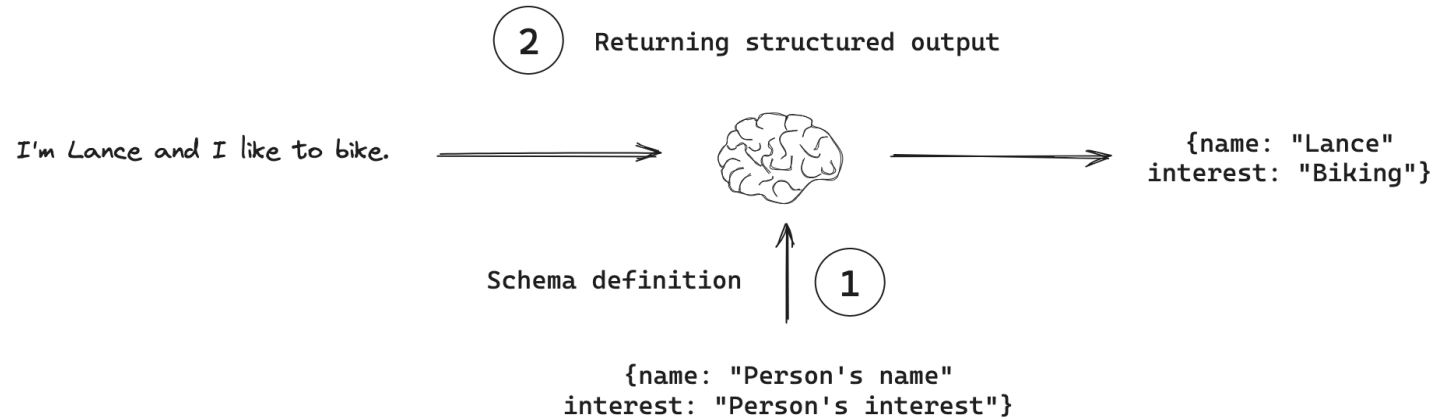


Strukturierte Outputs – Warum sie wichtig sind:

Problem mit unstrukturiertem Text:

- Unzuverlässigkeit
- Schwierige Validierung
- Hoher Entwicklungsaufwand

Formate: JSON, XML, YAML, ...



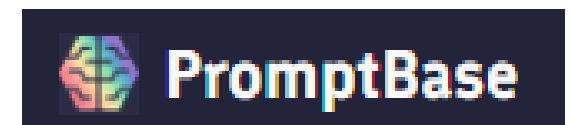
1. Schema definieren (Felder + Datentypen)
2. Schema im API-Aufruf übergeben
3. Sonderfälle / Edge Cases behandeln
4. Typische Daten und Beispiele verwenden

Prompt-Engineering-Tools

Prompt-Verwaltung:



Tools für die Prompt-Entwicklung und -Optimierung:



„Ein Prompt-Pattern ist eine strukturierte Vorlage, die dazu dient, die Interaktion von Nutzenden mit KI-Sprachmodellen zu leiten und zu optimieren.“

Flipped Interaction Pattern

Zweck und Relevanz

- Besonders effektiv in Bereichen mit geringem Vorwissen
- Effiziente Informationsgewinnung
- Kehrt die Nutzer–LLM-Interaktion um
- LLM steuert das Gespräch durch zielorientierte Fragen
- LLM kann Wissenslücken der Nutzenden gezielt nutzen

Struktur und Kernideen

Ich möchte, dass Sie mir Fragen stellen, um X zu erreichen

Sie sollten Fragen stellen, bis diese Bedingung erfüllt ist oder dieses Ziel erreicht wurde (alternativ: ohne Ende).

(Optional) Stellen Sie mir die Fragen einzeln, zu zweit usw.

Example: Outreach Message

You

Act as a professional sales manager with years of experience writing outreach messages via LinkedIn to new clients. I need you to generate an outreach message to a potential client. I want to book a coffee meeting with one of the main stakeholder and talk with her about our GenAI offers as I think it is relevant for her company. When you generate the message keep a professional tone and sound engaging. Keep the message to a maximum length of 280 characters. After you have generated the message, **I would like you to ask me 5 questions that will help you to improve the outreach message making it more tailored to the client.** Then revise the message with my input. We repeat this process until I am satisfied with the result.

(Iteratives) Anweisungsmuster

Zweck und Relevanz

- Nützlich für **wiederholbare Prozesse**
- Beschreibt Prozesse **Schritt für Schritt** an das LLM
- LLM führt Schritte gemäß Anweisung aus
- Jeder Schritt kann auf **Ergebnissen vorheriger Schritte** aufbauen

Struktur und Kernideen

Um [Ziel] zu erreichen, folge diesen Schritten:
Schritt 1: [Aufgabe]. Frage mich nach [Information].
→ Ich liefere [Eingabe].
Schritt 2: [nächste Aufgabe] usw.

(Optional) Frage mich, ob ich das Ergebnis überarbeiten möchte (dann zurück zu Schritt [n]) oder von vorne beginnen will (dann zurück zu Schritt 1).

Beginnen wir jetzt mit Schritt 1.

Example: User Study Design (Extract)

You

I want you to support me in generating a study design for a user research study.

To create the best outcome, I'd like you to do the following:

Step 1: You'll ask me what my goal is with the study. I will provide you the goal.

Step 2: You will ask for the study's context. I will provide you the context. ...

Step 4: Based on my answers, generate 5 research questions, aiming to reach my goal, considering the context. Ask me to select research questions and to provide my own.

...

Step 7: Based on all answers, create a study design document with the following components: ...

Now, let's start with step 1: Ask me about the goal of the study.



You

I want you to support me in generating a study design for a user research study. To create the best outcome, I'd like you to do the following:

1. You'll ask me what my goal is with the study. I will give you an answer, maybe just one sentence.
2. After you got my goal, you will ask for more context and background. I will then provide you more information around my thoughts for the study.
3. You will then ask me, if there is a specific focus of this study that you should be aware of. I can provide a focus to you or just indicate that there is no special focus.
4. Based on my previous answers, you will generate 5 research questions, that could help me to reach my previously defined goal with this study, considering the provided context and, if existing, the special focus. You then ask me to either select one or more of these research questions and/or provide my own research questions.
5. Based on the research questions, goal, background, and possible focus, you provide 3 different user research methods that could be used to answer the research questions effectively. For each proposed method, provide arguments for and against the method which should make it easier for me to decide which method to choose. I will then either choose one of these three methods or provide you my own one.
6. After you received the method, propose 2 to 4 different participant profiles that make sense for this study, considering research questions, goal, background, possible focus, and user research method. I will then select or describe participant profiles to you.
7. Based on all the previous answers, create a study design document with the following components:

Dr. Jannik Fischbach



fischbach@fortiss.org

fortiss Newsletter



Bleiben Sie auf dem
Laufenden!



Potenzialanalyse



Identifizieren Sie
konkrete Chancen!



Q / A Runde

Haben Sie Fragen?